Activities Of Daily Living Detection Using Smart Microphone

Athulya Shaji
MSc Computer Science (FT)
School of Computing, Engineering, and
Built Environment
Ulster University
Belfast, Northern Ireland
Shaji-A@ulster.ac.uk

***Abstract*— The paper discusses and demonstrates the use of an Artificial Intelligence (AI) model for classifying audio signals to identify Activities of Daily Living (ADL). The model detects ADL related sounds such as running water, door, washing machine, microwaves, etc. Preexisting ADL audio data were collected for the purpose of building an AI model. The data was preprocessed, and Mel-Frequency Cepstral Coefficient (MFCC) technique was used to extract useful features from the audio signal for classification. The model was trained with different Machine Learning (ML) algorithms such as Artificial Neural Network (ANN), Support Vector Machine (SVM), Decision Tree, k-Nearest Neighbors (kNN), Naïve Bayes (NB), and Perceptron. The performance of all the classifiers was evaluated and compared to select the best suited audio classifier. SVM obtained a high and consistent performance result of 78% accuracy. For capturing real time audio signals for ADL detection, a MATRIX Voice and Raspberry Pi 3 have been used. The ADL detection done using the selected audio classifier is logged along with the date and time for monitoring ADLs of the user. The model, when encountering unfamiliar audio signals, has been set up to annotate and store them for future reference which is a semi-automated process. Thereby continuously expanding the training dataset for the model to keep improving its functionality over the time.**

Keywords*— *Artificial Intelligence (AI), Activities of Daily Living (ADLs), Mel-Frequency Cepstral Coefficient (MFCC), Machine Learning Classifiers (MLCs), ANN, SVM, KNN, NB, MATRIX Voice, USB mic, Raspberry Pi, Audio classifier, Continuous learning.

I. INTRODUCTION

The term Activities of Daily Living (ADLs) was coined by Sidney Katz in 1950 [1][2]. The term refers to the basic activities one should be able to perform in order to have a good quality of living without dependence on another person or instruments [1]. The basic ADLs include dressing, cooking, toileting, personal hygiene, and transferring [1], as shown in Figure 1. The incapability to do those elementary tasks will lead to unsafe living conditions. Monitoring the ADLs of an individual is very significant in terms of health care as the measurements of the ability to perform ADLs can be an indicator of the need for assistance and other healthcare facilities.

A study published in the National Library of Medicine in 2019, which was conducted on 5,540 participants (2,739 women and 2,801 men; median age, 53.7 years), shows that out of the 5,540 participants, 1,097 (19.8%) developed ADL impairment between 50 and 64 years [3]. The association of functional impairment in middle age with hospitalization, nursing home admission, and death concludes that individuals with ADL impairment had an increased risk of adverse outcomes, including hospitalization, nursing home admission, and death compared with those without impairment [3].

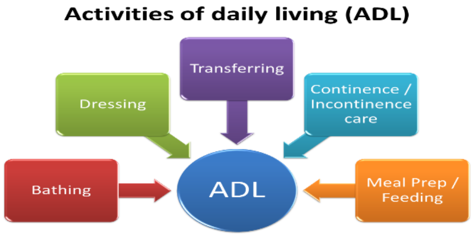

Activities of daily living (ADL)
Transferring
Dressing
Continence / Incontinence care
Bathing
ADL
Meal Prep / Feeding

Fig. 1. Basic ADLs. [4]

The above study indicates the importance of close monitoring of ADLs which gives rise to the relevance of this project. As the project is aimed at identifying the basic ADL sounds from a household using an AI model thereby helping to track an individual's living condition with minimal human interference or assistance. This has been brought into practice with the help of audio classifiers. Audio classifiers are the kind of machine learning algorithms that can identify different sounds from audio signals.

Figure 2 demonstrates the general process of building an audio classifier:

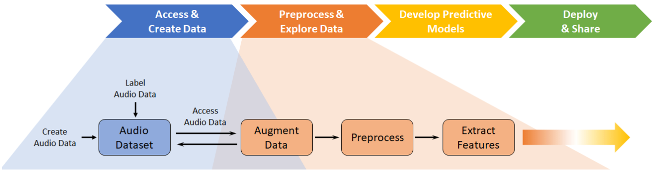


Fig. 2. Audio classifier building process. [5]

(1) Data collection: the audio signals that the model should identify have to be collected in the form of datasets that will be used by the model to learn. Several sounds from each class that the model needs to predict must be grouped together and labelled. (2) Pre-processing: the audio dataset is analysed using different data analysis techniques to understand the structure of the dataset and clean the dataset, which includes balancing the data, excluding outliers, etc. (3) Feature extraction: the process is to find relevant features in an audio signal that can help a model to classify it. Some of the techniques used to extract features from audio signals are Mel-Frequency Cepstral Coefficients (MFCCs), Fourier Transforms, etc. (4) Model development: implementing different machine learning algorithms such as Decision Tree, Naive Bayes, Support Vector Machines (SVM), Neural Networks, K-Nearest Neighbours (kNN), etc., followed by feeding different models with the pre-processed data. This

step also involves collecting the results of accuracy, precision and recall to understand the performance of each model and to select the best suited one with respect to the particular situation. (5) Model Deploy: using the best result model for predicting new audio signals collected in real time with the help of sensors such as a microphone integrated.

The process and technique implemented in this project to achieve the results have been explained in detail further in this paper. The remainder of this paper is as follows. Section 2 presents the related works in the area of ADL detection, HAR and SVM classifiers are discussed. Section 3 comprises of methodology elaborating the process of the development of the audio classifier. Followed by the 4th Section, which is the results of the trainings conducted to select the best audio classifier and outputs from the audio classifier. The last Section of the paper is the conclusion and discussion based on the outputs acquired from this project and the possible future scopes of this project.

II. RELATED WORKS

To understand more on the state of the art on audio classifiers, a close look was taken at some of the works done in the domain of audio recognition.

In [6], the authors provide details about the advantages and limitations of extracting features from audio signals using different techniques such as Zero-Crossing Rate (ZCR), Spectral Flux, Band-Energy Ratio, Spectral Contrast, and Gamma Tone Frequency Cepstral Coefficients. The paper offers evidence showing how these techniques works well with certain type of dataset and fails with others, for example MFCC gives good results in speech recognition but poor results when comes to classifying environmental sounds as they are highly variable with respect to the acoustic conditions. The paper discussed the implementation of a Convolutional Neural Network and its ability for feature extraction for audio-based datasets. Where a case study has been conducted on the developed model under real-life circumstances where the 1D CNN model was evaluated, unseen data gave detailed insight into the performance of the model.

Non-Markovian Ensemble Voting (NEV) is an audio-based recognition algorithm that can identify sounds of different duration that can be captured from a continuous stream of the audio signal that has been explored in detail in [7]. The audio stream was divided into several short frames which overlap with each other for a high level of data correlation. The MFCC of each frame is computed. Each MFCC feature is classified using a learned Random Forest (RF) classifier. Non-Markovian Ensemble Voting (NEV) is built upon the classification output from the RF classifier. NEV eliminates the need for silence detection and audio segmentation which is an advantage that makes it suitable for variable-length human activities detection.

An elaborate study of the features of Human Activities Sounds, especially indoor human activity sounds such as brewing coffee, teeth brushing, cooking, washing dishes, hand washing, showering, etc. that are significant in classifying the sound is presented in [8]. The samples for each sound were collected from different household environments with a 10 second duration each. The 10 second audio data was converted into an integer array where each integer represents the magnitude of the sound at that instance of time. Many statistical features such as mean, median, standard deviation, variance, kurtosis of the probability distribution, etc. were derived from the integer array. The MFCC of the sound signals was also extracted. Using a feature selection algorithm called Galgo, the features were ranked according to their potential to classify audio. Forward selection and backward elimination were conducted based on the rank of the features. The features selected using the aforementioned method were used to build two Human Activity Recognition (HAR) models based on Random Forest (RF) and Neural Network (NN) algorithms [8]. The result of the study concludes that Mel-Frequency Cepstral Coefficients (MFCC) is the most appropriate parameter to identify a sound with respect to a HAR model. The work in [8] also showcases a Random Forest model being the best suited under the experimental circumstances in the study.

Samik Sadhu and Hynek Hermansky [9] discuss the possibility of a continuous learning model for Automatic Speech Recognition (ASR) where they moved from a database specific to a natural learning approach that can improve the model over time by continuously learning from the new data collected. The approach presented in [9] details a dynamically expanding Hidden Markov Mode (HMM) – Deep Neural Network (DNN) ASR model for continuous learning in ASR. The study showcases the results of knowledge transferring both forward and backward while implementation of Continuous Learning (CL) taking the catastrophic forgetting problem into account.

The authors of [10] presented two approaches for semi-automated online data labeling. The purpose of the approach is to overcome the demerits of self-annotation. The data annotation can be classified based on four criteria: (i) temporal (which depends on when the labeling is done), (ii) annotator (which is based on who is labeling), (iii) scenario (which answers where the activity is conducted), and (iv) annotation mechanism (which is how the annotation is performed).The work in [10] has a detailed explanation of the above four labeling approaches. The challenges of labelling and annotating unknown sounds have been explored under two circumstances. The first is data labelling using gesture recognition which obtained a precision of 81.5% on an SVM model, and the second is labelling home activities sounds using a smart microphone which results in 96.20% accuracy on an HMM model [10].

In [11] a detailed analysis of the annotation of unknown audio using smart microphones. The paper details an Intelligent System for Sound Annotation (ISSA) which continuously listens and asks for input from the performer to label the unknown audible human activity. ISSA consists of a MATRIX voice board, and an HMM classifier model which used MEL–MBSES features of audio. The voice assistant is built using an open-source tool Snips [11]. ISSA has been implemented in a form to identify pre-trained activities and to understand and confirm new or unknown sounds. The model resulted in a 92.65% accuracy.

Audio classification in an unstructured environmental sound which contains noise is discussed in [12], where an MFCC–SVM model has been implemented. The study was conducted in a live office environment where audio data such as pen drop, cough, keyboard, alerts, phone, keys etc. were collected. The classifier gave a result of 62% precision and 58% recall.

The authors in [13] have used an MFCC – SVM model to identify the emotions in speech such as anger, happiness, sadness, and neutral. The audio data required for the same was captured from Indonesian movies, which was cropped, and WAV formatted during data preprocessing. The model was implemented using linear kernel and polynomial kernel which resulted in an accuracy of 66% and 45% respectively.

In [14] a grid search approach is studied to understand in depth the significance of optimization of kernel parameters in a Support Vector Machine (SVM) classifier that can improve the results of a classifier on audio signals. The study was conducted on linear, radical basis, and sigmoid kernels. Mel-Frequency Cepstral Coefficients were the feature used for classification. The study shows that the optimization of kernel parameters results in a significant improvement in accuracy. Especially in sigmoid where the default parameters give an accuracy of 40.95 %, whereas an optimized model gives an accuracy of 97.32%.

Yuh and Kang [15] implemented a Deep Neural Network (DNN) to classify human ADLs from real-time sound captured from the domestic environment in their work [15]. Sounds from the kitchen, bedroom, and bathroom were captured using a microcontroller sensor device. An open-source platform called Audacity was used to annotate the recorded WAV signals. Short-Time Fourier Transform (STFT) has been used to extract features from the collected audio dataset. A 2D Convolutional Neural Network (CNN) was implemented as the classifier where the data was split into 80:20 train: test ratio with 20 epochs. The model achieved an accuracy of 95.55%.

The work in [16] by Prasanna and Tripathi details the performance of Machine Learning Classifiers (MLC) on audio signals. The primary aim of their work is to detect speech. The audio dataset consists of 10 classes of 30 sounds each, the audios are in WAV format. The dataset was preprocessed, and features were extracted which were then fed to multiple Machine Learning Classifiers (MLC). Multiple parameters of performance analyses such as accuracy, precision, F1 score, specificity, and sensitivity from each classifier were recorded and compared. The highest accuracy was obtained for Stochastic Gradient Descent (SGD) followed by Support Vector Machine (SVM) at 93% and 91% respectively. Random Forest (RF) and Naïve Bayes (NB) have also given adequate results with an accuracy of 87% and 84% respectively.

Audio signals recorded using the mobile phone were used to identify human activities by Naronglerdrit et al. in [17]. Audio data was collected from a home environment with the help of a mobile microphone at a distance of 1 meter. Short time analysis is used to extract features from the audio signals. An unsupervised clustering technique was imposed on the decomposed subspaces of the audio dataset. The parameterization methods used were the Harmonics to Noise Ratio (HNR), the 12 first Mel-Frequency Cepstral Coefficients (MFCC), Zero-Crossing Rate (ZCR), 20 Linear Prediction Coding (LPC) coefficients [17]. The ReliefF algorithm is used to rank the features and Expectation-Maximization (EM) for clustering them. Decision Tree, Naive Bayes (NB), Support Vector Machines (SVM), k-Nearest Neighbor classifier (KNN), and perceptron Neural Network (NN) were the classifiers used. The accuracy results of the models were: 92.46% for NN, 89.5% for KNN, 89.03% for SVM, and 83.30% for NB.

III. METHODOLOGY

The process of developing an audio classifier to detect Activities of Daily Living (ADLs) and using the model in real time has been elaborated in this section. Figure 3 shows the architecture of the model. The principle building blocks of the model and the cycle of the process can be understood from the architecture diagram. The Python language and the PyCharm IDE [24] were used for all the processes.

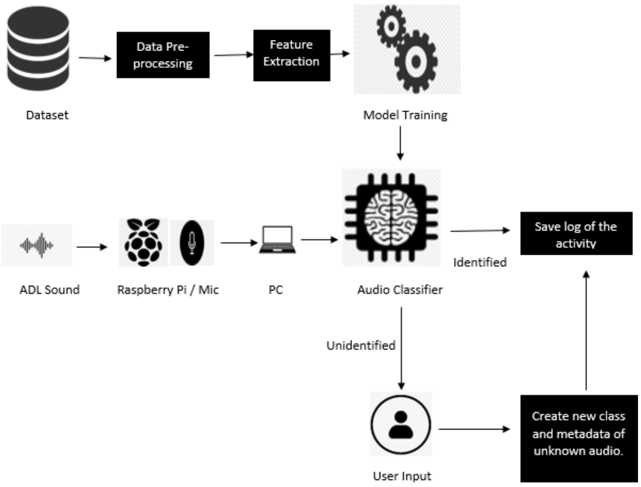
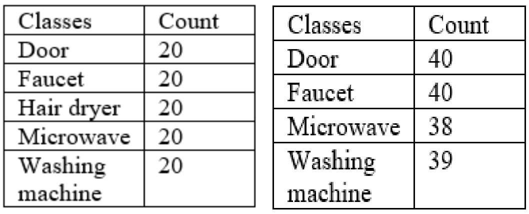


Fig. 3. Architecture of the Model.

Data Collection: the data used for the training and testing of the model was collected from the open-source website Pixabay [18]. Sound effects for different household activities were gathered from the website. The five classes of audio collected are door opening and closing sounds, running water from a faucet, hair dryer, microwave, and washing machine. The audio signals were of different duration and in MP3 format. The two sets of datasets used for training the Machine Learning Classifiers are shown in Table 1.

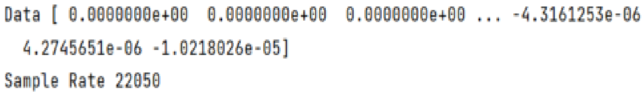
TABLE I
Datasets used for training the model.



Classes	Count
Door	20
Faucet	20
Hair dryer	20
Microwave	20
Washing machine	20

Classes	Count
Door	40
Faucet	40
Microwave	38
Washing machine	39

Data preprocessing: the collected MP3 audio datasets were converted into WAV format and the metadata of the audio sets were stored in a CSV file. The Librosa library from Python was used for data preprocessing of audio signals. Audio signals are continuous waveforms, these analog signals have to be converted into digital signals for further feature extraction and model training. The audio signals were converted into a Numpy array of Data and Sample Rate using the Librosa library as shown in Figure 4. The data is an array of integer values of discrete samples derived from the continuous audio signal. The sample rate indicates the number of times the analog signal has been divided per second [25]. The sample rate taken here is 22,050. This can also be visualized as a waveform as shown in Figure 5.



```
Data [ 0.0000000e+00  0.0000000e+00  0.0000000e+00 ... -4.3161253e-06
  4.2745651e-06 -1.0218026e-05]
Sample Rate 22050
```

Fig. 4. Data array and Sample rate of a 1.12-minute-long microwave audio signal.

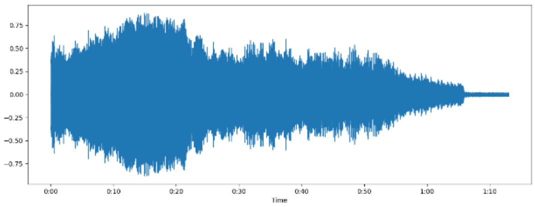


Fig. 5. Waveform representation of a 1.12-minute-long microwave audio signal.

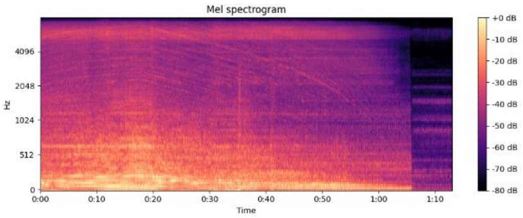
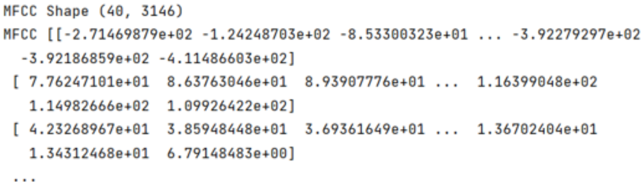


Fig. 6. Spectrogram of a 1.12-minute-long microwave audio signal



```
MFCC Shape (40, 3146)
MFCC [[-2.71469879e+02 -1.24248703e+02 -8.53300323e+01 ... -3.92279297e+02
  -3.92186859e+02 -4.11486603e+02]
 [ 7.76247101e+01  8.63763046e+01  8.93907776e+01 ...  1.16399048e+02
   1.14982666e+02  1.09926422e+02]
 [ 4.23268967e+01  3.85948448e+01  3.69361649e+01 ...  1.36702404e+01
   1.34312468e+01  6.79148483e+00]
 ...
```

Fig. 7. The first 40 MFCCs of a 1.12-minute-long microwave audio signal.

Feature extraction: the feature extraction technique used in this methodology is Mel-Frequency Cepstral Coefficients (MFCCs). MFCC is a very commonly used feature extraction method for audio signals in activity recognition, speech recognition, etc. ([6], [7], [12], and [13]. The audio signal is converted into a spectrogram (see Figure 6) showing the energy distribution in an audio signal which can be divided into several Mel-frequency bins. A logarithmic transformation is applied on the spectrum resulting in a set of cepstral coefficients (see Figure 7).

Model Training: the preprocessed audio data was used to train different Machine Learning Classifiers (MLCs), the Tensorflow and Keras library was used for this purpose. The classifying algorithms used were Artificial Neural Network (ANN), Support Vector Machine (SVM), Decision Tree, k-Nearest Neighbors (kNN), Naïve Bayes (NB), and Perceptron. The models were trained using two sets of datasets (see Table 1) i.e. a dataset consisting of five classes of 20 samples each and another dataset with four classes of approximately 40 samples each. The models were also trained on the first 40 MFCCs as well as on 20 MFCCs. Different performance parameters were computed for each of the models to select the best suited model for the audio classifier. The consolidated results of all the training conducted are given further in this paper.

Audio classifier: the previous process of training different Machine Learning Classifiers (MLCs) and evaluating their performance led to the conclusion of selecting the Support Vector Machine (SVM) model as the best suited classifier for the classification of Activities of Daily Living (ADLs) considered in this project with a cross-validation accuracy score of 78%.

Raspberry Pi and MATRIX/USB microphone: the Raspberry Pi 3 model b+ (shown in Figure 8) is a single-board computer (BCM2835 SoC) with 1GB RAM which is incorporated into the model [22]. The MATRIX Voice (see Figure 8) is a smart array of microphones aligned in all directions and has an Analog-to-Digital Converter (ADC) [23]. Ideally, the proposed system was to capture real time ADLs using the MATRIX Voice system and Raspberry Pi. However, the configuration of the MATRIX Voice along with the Raspberry Pi were giving some bugs, therefore a USB mic (see Figure 8) was used to capture ADL sound in real-time and to store it in the Raspberry Pi home folder.

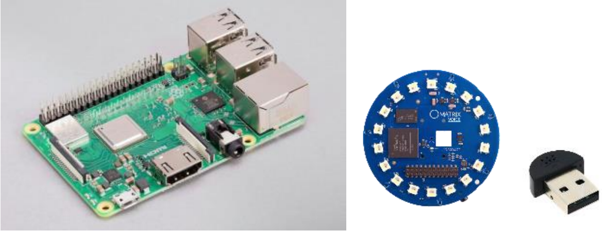


Fig. 8. Raspberry Pi [19], MATRIX voice [20], USB mic [21] (left to right).

Audio signal transfer and detection: the ADL sound captured by the Raspberry Pi and the USB microphone was sent to the computer machine using the SSH client Putty [26]. The audio signal was then fed to the pre-trained audio classifier. Depending upon the output of the audio classifier different sets of functions are performed. If the audio classifier identifies the given audio signal with an accuracy greater than or equal to 70%, the log of the activity, which includes the current date, time, and the activity is stored in a CSV file. On the other hand, if the audio signal is identified with an accuracy of less than 70%, user input is requested for identifying the audio signal.

Annotation of unknown audio signals: the user-interacting interface for data identification and confirmation of new data labelling is the computer screen and the input is taken in text form. If the received input from the user is an existing class of the current dataset, the new audio signal is added to the respective dataset class. Otherwise, a new class will be created, and the audio signal will be stored in the newly created class for future training of the audio classifier. The metadata of the newly identified audio signal will be created and stored. As well as the ADL log will also be created for the newly identified ADL.

IV. Results

The results of the training of different Machine Learning Classifiers (MLCs) were captured for the selection of the best classifier with respect to the audio dataset used in this project. The parameters considered for the evaluation of performance were accuracy, precision, recall, and 10-fold cross-validation mean score. Accuracy is the percentage of correct predictions done by the model with respect to the total number of predictions done by the model. Precision is the percentage of true positives with respect to the total of true positives and false positives. Recall is the percentage of true positives with respect to the total of true positives and false negatives. Cross-validation is a method of assessing the model's performance on different subsets of the dataset. The six classifiers considered, namely ANN, SVM, Decision Tree, KNN, NB, and Perceptron were trained multiple times under different conditions and parameters. Table 2 shows the results of the training conducted on all the above-mentioned classifiers on a dataset of five classes with 20 samples each (see Table 1) with an MFCC value of 40 and 20. The train test ratio in this test was 80:20.

TABLE II
Accuracy (%) of six MLCs.

MFCC	ANN	SVM	Decision tree	KNN	NB	Perceptron
40	50	78	60	70	68	61
20	55	78	80	70	65	53

Table 3 consolidates the result of the classifiers on a dataset of four classes with approximately 40 samples each (see Table 1) with both the MFCC value of 40 and 20 respectively. The train test ratio in this test is 80:20.

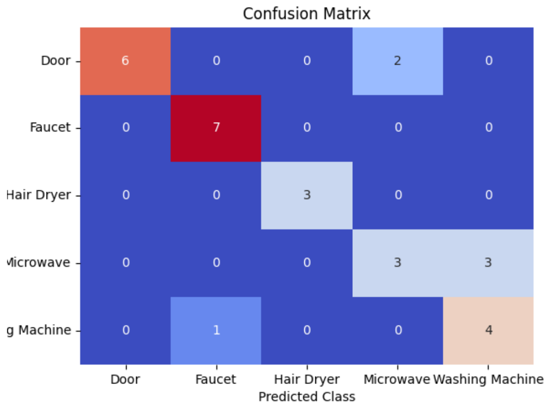
TABLE III
Accuracy (%) of six MLCs.

MFCC	ANN	SVM	Decision tree	KNN	NB	Perceptron
40	43	75	66	68	72	65
20	50	72	62	68	71	62

The interpretation from the above training results concludes that Support Vector Machine (SVM) is the classifier to give a consistently high accuracy rate. For a further understanding of the stability of SVM, the model was also trained on a 70:30 train: test ratio. Results of which are presented in Table 4.

TABLE IV
Training results of SVM on 70:30 train: test ratio.

Dataset	4 classes 40 samples		5 classes 20 samples	
MFCC	Mean CV (%)	Accuracy (%)	Mean CV (%)	Accuracy (%)
40	74	62	71	79
20	66	60	78	79

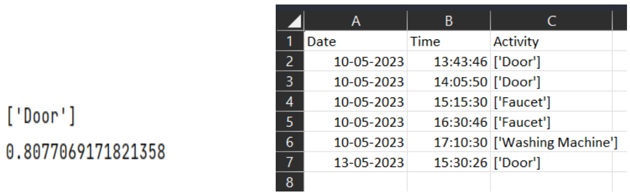


	Door	Faucet	Hair Dryer	Microwave	Washing Machine
Door	6	0	0	2	0
Faucet	0	7	0	0	0
Hair Dryer	0	0	3	0	0
icrowave	0	0	0	3	3
g Machine	0	1	0	0	4

Fig. 9. Confusion Matrix of SVM.

The model selected as the audio classifier in this project is the SVM model trained on the dataset of five classes with 20 samples (Table 1). The first 40 MFCC features were considered for training with a train test ratio of 70:30. Figure 9 shows the confusion matrix of the selected Support Vector Machine (SVM) audio classifier.

Figure 10 shows a sample output from the selected SVM audio classifier detecting ADL captured in real time with an accuracy of 80% and the ADL log been created for the same.

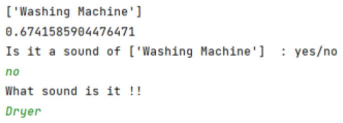


```
['Door']
0.8077069171821358
```

	A	B	C
1	Date	Time	Activity
2	10-05-2023	13:43:46	['Door']
3	10-05-2023	14:05:50	['Door']
4	10-05-2023	15:15:30	['Faucet']
5	10-05-2023	16:30:46	['Faucet']
6	10-05-2023	17:10:30	['Washing Machine']
7	13-05-2023	15:30:26	['Door']
8			

Fig. 10. Output from the audio classifier and the ADL log created.

Figure 11 shows a sample output from the audio classifier detecting ADL captured in real time with an accuracy of 67%, therefore user input has been taken for labelling the unknown audio signal. In this case, the audio signal is saved in a newly created class called "Dryer" in the dataset directory (Table 5).



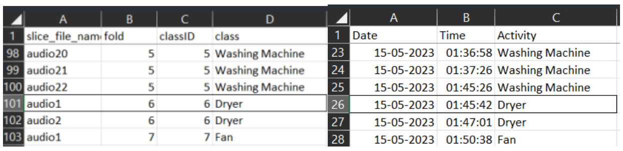
```
['Washing Machine']
0.6741585904476471
Is it a sound of ['Washing Machine']  : yes/no
no
What sound is it !!
Dryer
```

Fig. 11. Output from the audio classifier.

The log of the newly identified activity is stored in the ADL log CSV file. The metadata for the new audio signal is also added to the corresponding CSV file (Figure 12)

TABLE V
Updated Dataset

Classes	Count
Door	20
Faucet	20
Hair dryer	20
Microwave	20
Washing machine	20
Dryer	1



	A	B	C	D
1	slice_file_nam	fold	classID	class
98	audio20	5	5	Washing Machine
99	audio21	5	5	Washing Machine
100	audio22	5	5	Washing Machine
101	audio1	6	6	Dryer
102	audio2	6	6	Dryer
103	audio1	7	7	Fan

	A	B	C
1	Date	Time	Activity
23	15-05-2023	01:36:58	Washing Machine
24	15-05-2023	01:37:26	Washing Machine
25	15-05-2023	01:45:26	Washing Machine
26	15-05-2023	01:45:42	Dryer
27	15-05-2023	01:47:01	Dryer
28	15-05-2023	01:50:38	Fan

Fig. 12. Updated metadata and ADL log files

V. CONCLUSION AND DISCUSSION

As a result of several training conducted on the six MLCs under different parametric circumstances, it can be concluded that Support Vector Machine (SVM) is the best-fitted audio classifier for this project. SVM is one of the common audio classifiers used in human activity recognition, speech recognition, and music classification [12]- [16]. The results from this project also prove SVM to be the most reliable audio classifier.

This project has tried to implement an audio classifying model to detect the Activities of Daily Living (ADLs) initially trained to identify five activities namely door opening and closing sounds, running water from a faucet, hair dryer, microwave, and washing machine. The model is expected to improve its functionality over time by retraining the model with the expanding dataset. Using the semi-automated labelling technique implemented in this model, the system can identify new audio signals with the help of user input.

The accuracy, as well as the mean CV score of 10-fold cross-validation, was considered to understand the performance of the selected audio classifier. Accuracy alone can be misleading depending on whether the data is balanced or unbalanced. The expanding dataset in this project can lead to an unbalanced dataset with few samples per class. Cross-validation is a good technique to understand the robustness of the model. The chosen SVM model gives consistent results mostly above 70% under different training conditions conducted in this project.

The Activities of Daily Living (ADLs) detection using a smart microphone system implemented in this project has many scopes of future improvement. A few of which are:

Integrating the MATRIX Voice: the current approach is using a USB microphone along with a Raspberry Pi to capture ADL sounds in real time for audio detection. A successful integration of MATRIX Voice can help not only to capture sound but also to identify the direction from which the sound is produced. This can be of added advantage in a domestic environment to identify sounds like running water from a faucet whether coming from the kitchen or bathroom.

Automate audio recording and transfer: in the current system the audio recording is done using the record command on Raspberry Pi and the transfer of the captured audio signal to the computer for detection is done with the file transfer command on the computer after connecting the Raspberry Pi with the computer using the SSH client Putty. This process can be automated into a system that can continuously capture ADLs and transfer them to the system where the audio classifier is built.

Improved User Interface (UI): the current user interface for receiving input while annotation of unidentified audio signal is the IDE console. This can be changed to a webpage for a better user experience. As well as the current format of input received is in text format which can be upgraded to voice input through the same hardware components used to capture the ADL sounds.

ADL log analysis: the log created by the model after the detection of every ADL sound captured in real time over a period of time can be analyzed for identifying patterns for health monitoring purpose as well as can be used for giving suggestions to the user while annotating an unidentified audio signal.

Automate Audio classifier retraining: the proposed system has an ever-expanding dataset that will be used to re-train the model to improve its functionality. An automatic system to retrain the model after every 30 days and select the best model for future use by comparing the performance metrics.

VI. REFERENCE

[1] Edemekong, P. F. et al. (2022) Activities of daily living. StatPearls Publishing.

[2] Katz S. Assessing self-maintenance: activities of daily living, mobility, and instrumental activities of daily living. J Am Geriatr Soc. 1983 Dec;31(12):721-7. [PubMed]

[3] Brown, R. T. et al. (2019) ‘Association of functional impairment in middle age with hospitalization, nursing home admission, and death’, JAMA internal medicine, 179(5), pp. 668–675.doi:10.1001/jamainternmed.2019.0008.

[4] Redirect notice (no date) Google.com. Available at: https://www.google.com/url?sa=i&url=https%3A%2F%2Fltcga.com%2Flong-term-care-guide-activities-of-daily-living%2F&psig=AOvVaw37QBacEzWByPx9YRG5qYfS&ust=1683637561856000&source=images&cd=vfe&ved=0CBAQjRxqFwoTCJjBqezk5f4CFQAAAAAdAAAAABAE (Accessed: 8 May 2023).

[5] Deep learning for audio applications - MATLAB & Simulink - MathWorks Switzerland (no date) Mathworks.com. Available at: https://ch.mathworks.com/help/audio/gs/intro-to-deep-learning-for-audio-applications.html (Accessed: 8 May 2023).

[6] Cruciani, F. et al. (2020) ‘Feature learning for Human Activity Recognition using Convolutional Neural Networks: A case study for Inertial Measurement Unit and audio data’, CCF Transactions on Pervasive Computing and Interaction, 2(1), pp. 18–32. doi: 10.1007/s42486-020-00026-2.

[7] Stork, J. A. et al. (2012) ‘Audio-based human activity recognition using Non-Markovian Ensemble Voting’, in 2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication. IEEE, pp. 509–514.

[8] Galván-Tejada, C. E. et al. (2016) ‘An analysis of audio features to develop a human activity recognition model using genetic algorithms, random forests, and neural networks’, Mobile Information Systems, 2016, pp. 1–10. doi: 10.1155/2016/1784101.

[9] Sadhu, S. and Hermansky, H. (2020) ‘Continual learning in automatic speech recognition’, in Interspeech 2020. ISCA: ISCA.

[10] Cruz-Sandoval, D. et al. (2019) ‘Semi-automated data labeling for activity recognition in pervasive healthcare’, Sensors (Basel, Switzerland), 19(14), p. 3035. doi: 10.3390/s19143035.

[11] Garcia-Constantino, M. et al. (2019) ‘Semi-automated annotation of audible home activities’, in 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops). IEEE, pp. 40–45.

[12] Kucukbay, S. E. and Sert, M. (2015) ‘Audio-based event detection in office live environments using optimized MFCC-SVM approach’, in Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015). IEEE, pp. 475–480.

[13] Muljono et al. (2019) ‘Speech emotion recognition of Indonesian movie audio tracks based on MFCC and SVM’, in 2019 International Conference on contemporary Computing and Informatics (IC3I). IEEE, pp. 22–25.

[14] Grama, L., Tuns, L. and Rusu, C. (2017) ‘On the optimization of SVM kernel parameters for improving audio classification accuracy’, in 2017 14th International Conference on Engineering of Modern Electric Systems (EMES). IEEE, pp. 224–227.

[15] Yuh, A. H. and Kang, S. J. (2021) ‘Real-time sound event classification for human activity of daily living using deep neural network’, in 2021 IEEE International Conferences on Internet of Things (iThings) and IEEE Green Computing & Communications (GreenCom) and IEEE Cyber, Physical & Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics). IEEE, pp. 83–88.

[16] Prasanna, D. L. and Tripathi, S. L. (2022) ‘Machine learning classifiers for speech detection’, in 2022 IEEE VLSI Device Circuit and System (VLSI DCS). IEEE, pp. 143–147.

[17] Naronglerdrit, P., Mporas, I. and Sotudeh, R. (2017) ‘Monitoring of indoors human activities using mobile phone audio recordings’, in 2017 IEEE 13th International Colloquium on Signal Processing & its Applications (CSPA). IEEE, pp. 23–28.

[18] (No date) Pixabay.com. Available at: https://pixabay.com/sound-effects/search/flush/?theme=household (Accessed: 11 May 2023).

[19] (No date b) Prismic.io. Available at: https://images.prismic.io/rpf-products/bef8cda3-64ea-4098-bf188e731a6e9a0d_3b%2B%20Angle%202.jpg?ixlib=gatsbyFP&auto=compress%2Cformat&fit=max&w=799&h=533 (Accessed: 11 May 2023).

[20] (No date c) Matrix.one. Available at: https://www.matrix.one/assets/imgs/products/voice/device.png (Accessed: 11 May 2023).

[21] (No date d) Shopify.com. Available at: https://cdn.shopify.com/s/files/1/0176/3274/products/mini-usb-microphone-the-pi-hut-10286533678856618179_grande.jpg?v=1646920627 (Accessed: 11 May 2023).

[22] Raspberry Pi Ltd (no date) Raspberry pi OS, Raspberry Pi. Available at: https://www.raspberrypi.com/software/ (Accessed: 11 May 2023).

[23] MATRIX voice (no date) Matrix.one. Available at: https://www.matrix.one/products/voice (Accessed: 11 May 2023).

[24] Download PyCharm: Python IDE for professional developers by (no date) JetBrains. Available at: https://www.jetbrains.com/pycharm/download/ (Accessed: 11 May 2023).

[25] McFee, B. (2019) Why resample on load?, librosa blog. Available at: https://librosa.org/blog/2019/07/17/resample-on-load/ (Accessed: 12 May 2023).

[26] Download PuTTY - a free SSH and telnet client for Windows (no date) Putty.org. Available at: https://www.putty.org/ (Accessed: 12 May 2023).