Detection of Audio-based Emergency Situations using SCALE

Aditya Nair
adityan1@uci.edu
Prerna Singh
prerns1@uci.edu
Shivam Arora
shivama1@uci.edu

June 14, 2019

1 Introduction

Each year, a large number of elder people in the US are injured — some critically or even fatally — by a fall. A study revealed that people who are over 65 years old and up, one out of five falls can cause a severe injury such as broken bones or head injuries. Therefore, we can agree that falls are a pressing matter for older people, and the earlier the emergency help comes for aid, the better. The growth of smart home technology has opened up a lot of options to automate uncomplicated tasks like switching on the lights and turning up the heat, however its potential for healthcare applications has largely remained untapped. It is claimed that around 18.7 million homes in the US have a smart speaker which opens up a lot of opportunities to tap the healthcare potential in a smart home. The problem we plan to address is the detection of emergency situations such as a fall in a household using the microphones in a smart home device with various sensors. Once an emergency situation occurs, i.e., it detects a person screaming and asking for help, it informs their emergency contacts and also calls the emergency services.
SCALE [4] is an event-driven middleware platform where devices upload sensed events to a cloud data exchange and the analytics service retrieves these events, scan for possible emergencies and send residents alerts to confirm or reject the emergency.
We built a tool that constantly records audio using the mic on the Raspberry Pi and when it detects a shout or a loud noise, it starts recording. This recorded audio is sent to a trained machine learning model which detects if it is a human scream or not. If the model identifies that as a human scream, it captures an image through a camera connected to the Raspberry Pi. This is primarily done to minimize the false positives. This image is published to SCALE to confirm or reject the emergency and also for informing the emergency services. We have used an event based escalation approach to make sure that the privacy of users is not encroached upon. An interesting future direction that can be explored will be to have a variety of sensors to confirm about emergency situations. Also, we can have a more complex machine learning model which identifies stress keywords like 'help', 'ouch' and others.
The rest of the report is organized as follows: section 2 gives an overview of the related works, section 3 describes the architecture of the tool we developed, section 4 comprehensively describes our project setup, section 5 states the results of the developed tool. Finally, in section 6, we discuss how we can extend our work.

2 Related Works

This section reviews various literary works that have made commendable contribution in this area. We have taken inspiration from these works for building our tool.

2.1 Using smartphone for ADL classification

Activities of daily living (ADLs) are routine activities that can be finished by a person without any outside help. ADLs are categorized into six groups, eating, bathing, dressing, toileting, walking, and continence. Health status and long-term care need of a person is evaluated through the degree to which a person can persistently perform ADLs.

User's daily living activities can be detected using multiple sensors installed in many places in a smart home, including the user's body. In this paper, Feng, K. Chang, and Chang [5] approach data collection and classification of activities through a smartphone alone. Using a smartphone reduces the energy consumption and setup cost of a smart home. Combination of sound, orientation and Wi-Fi signals data is analyzed to identify ADLs (activities of daily living).

This paper uses GPS, Wi-Fi-based Received Signal Strength Indicator (RSSI) information, and combined strength characteristics of Wi-Fi access points (APs) for precise localization. Location is derived from where the audio events take place. The existing Wi-Fi infrastructure installed in the homes is used, and RSSI data is collected through the smartphone. For this project, only the potential area or room was identified, rather than the exact precise location to reduce the calculation overhead. Support vector machine is used as a classifier for completing localization prediction.

To detect the orientation, the rotation matrix in the Android system is used. It computes rotation around x, y, and z axes. It transforms from the previous matrix to the current matrix. getRotationMatrix() method is used to compute the inclination matrix and the rotation matrix transforming from the device coordinate system to the world's coordinate system.

Timbre is the primary perceptual attribute connected with pitch, loudness, intensity, and duration of sound. Different sounds can be classified by analyzing timbres. Location and orientation information is used to conclude user's activity. Smartphone records the audio files in device storage. Fast Fourier transform is applied to each audio file to get fingerprints, which is used to classify audio to each category of activity.

This ADL recognition system was tested in four apartment setups. The three sources, Wi-Fi fingerprinting, orientation detection and sound classification system assist in deciding activity results. The results from the experimental setup provided satisfactory results in the average recognition rate of ADL.

In the paper by Feng, K. Chang, and Ming [6], they present a similar ADL recognition system that is affordable and is a single point smartphone-based mobile application. This paper involves almost the same authors as in the previous paper. This paper focuses on the business side of the same concept of recording ADLs using a smartphone. The system involves a front-end Recorder App that records activities and a cloud backend that does data fusion and recognition analyses.

This system collects additional information from sensors providing behavioral as well as environmental context. These include embedded sensors, microphone, Wi-Fi scan module, orientation of device, light proximity, step detector, accelerometer, gyroscope, magnetometer, and timestamp. There is a negligient gap between sensors and user which improves the quality of data collected. This data is preprocessed, analyzed, and goes through a fusion process at the backend of the ADL system.

A few research questions persist regarding the accuracy of the ADL details of the user. These include GPS turn off, a user moving only indoors, no walking movement but arm or body posture changes, small actions like typing on a keyboard or turning on microwave, can these kinds of motion be detected and analyzed. All these details are essential to obtain accurate ADL details.

Location and movement problems are solved using robust positioning characteristics, GPS, localization classifier aided Wi-Fi fingerprinting technology and light-based indoor positioning algorithm for precision from room-level down to furniture-level. Data fusion operation is performed on the backend to ascertain users' movement activities, including running, walking, standing, sitting, lying, taking an elevator, etc. Small actions are recognized using acoustic feature extraction and classification into different categories from environmental sound recognition method integrated in the backend. Sound is recorded in the app as audio files and sent to the server. There are other highly accurate ADL indentification algorithms implemented to detect audible events, actions like cutting vegetables, and environmental sounds. Details of the different algorithms have not been discussed in the paper.

The ADL recognition system was also tested in four apartment setups. The accuracy was higher since it involved more data from additional sensors used in the system.

2.2 Making ADL detection system energy efficient

In the paper by Alhassoun, Uddin and Venkatasubramanian [3], a perpetual heterogeneous IoT system, SAFER is presented. It is deployed in homes to recognize critical events that require urgent action and response. To make the system energy efficient, battery-operated and wall-powered IoT devices are used. A semantic approach is followed that extract ADLs from device data for energy-optimized sensor activation.

Different IoT devices including wearables, ambient, and vision, have different capabilities of power, connection, reliability, and accuracy. There is a need to minimize the energy consumption of the battery and wall-powered devices for long term operations and also maintaining accuracy. SAFER is a three-phase system framework. In the learning phase, the deployment setting is captured which includes monitoring and instrumentation of floor space segmentation, IoT device profiles, and status and configurations of devices. In the activity recognition phase, knowledge from the learning phase is utilized to recognize activities whenever new occupancy sensor events are recorded. In the configuration phase, depending on the current activity location and type, the status of the IoT devices is adjusted at runtime. A dynamic configuration algorithm executes on the local controller to compute and realize optimal energy configurations of devices.

To regulate the energy efficiency of all the IoT devices in the network, different techniques are employed including priority algorithms based on location and power supply, and greedy algorithms based on balanced remaining battery lifetime and cost function gradient. The SAFER system for elderly fall detection includes a smart pressure pad, wearable sensor, mobile accelerometer sensor, and camera. The interconnection of different sensors and networks was supported by Raspberry Pi B. It publishes the data to SCALE [4] server using MQTT protocol.

Efficiency and accuracy of different sensors were measured, and it was recorded through experimental studies that the algorithm achieved success in energy consumption reduced to a great extent.

2.3 Audio-based emergency situation detection

Audio signals provide important pieces of evidence of the situation. Audio signal information complements the information from video signals. Elderly care, home care, and home security are essential. Screaming is one of the events that is important in such environments for family members, caregivers, and security guard.

In the paper by Huang et al. [10], an approach to scream detection is presented, using both analytic and statistical features for the classification. Log energy is used to detect energy continuity of audio.

This continuity represents screaming that usually lasts longer than many other sounds. High pitch detection based on autocorrelation is used to extract the highest pitch of each frame.

Scream, a high-frequency sound, is usually presented as a sound segment with continuous and relatively high energy. The pitch is often higher than normal speech. With these key features in scream sound, autocorrelation technique is used to extract high pitch in a sound with high energy. Now after detecting long segments of high energy and high pitch, to further improve the detection new features need to be introduced. MFCC was used for speech recognition and comparing scream and non-scream audio.

The analysis of the extracted features that characterize scream could be done in real time with a delay of 40ms. In the training phase, scream and non-scream segments were labeled in a data set, and MFCC features were extracted in these segments for SVM training. The results suggested that this scream detection approach can work well if trained properly.

In the paper by Nguyen, Yun, and Choi [9], a perception sensor network (PSN) is presented for detecting audio-based emergencies in room environments such as human scream, cry, and alarm. PSN has multiple units consisting of a Kinect for capturing audio signals and a pan-tilt-zoom camera. Audio signals obtained from Kinect are processed. Audio processing is done in two parts, sound source classification (SSC) and sound source localization (SSL). SSC identifies if the sound source is an emergency or normal speech. SSL computes the location of the sound source. For instance, in the case of screaming, SSL tries to identify source person from multiple people in the room. After the event is detected, commands are sent to the robot for taking necessary action.

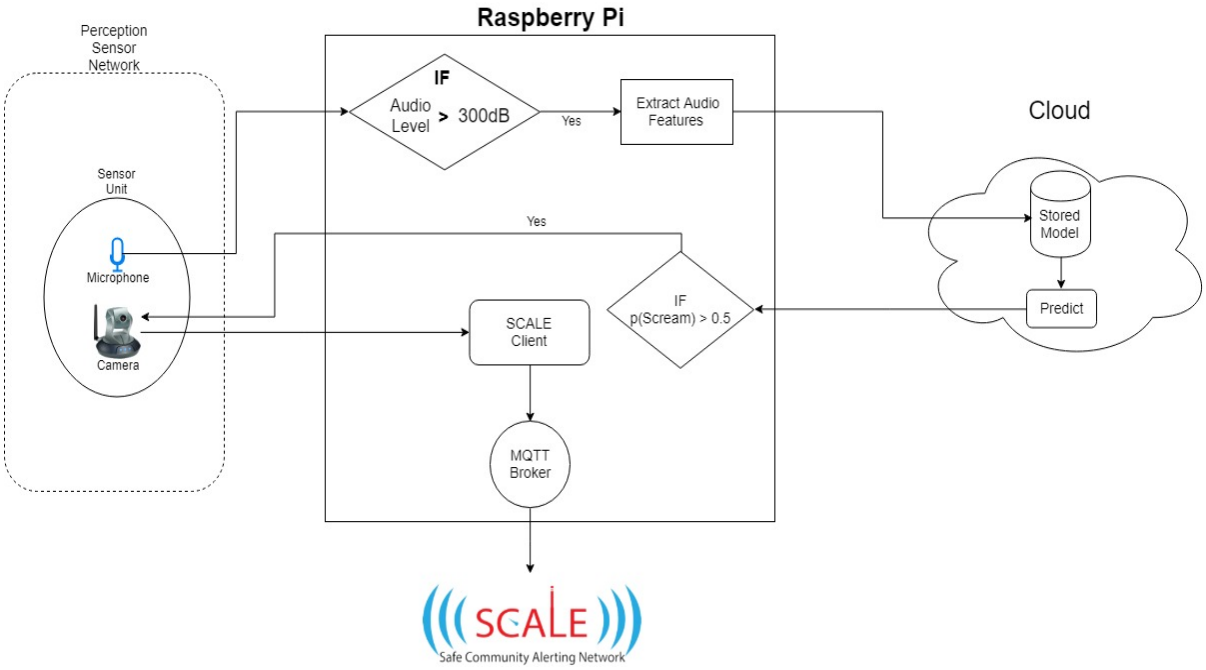
SSC module is also used to find what happened in the room using the audio signals. Matching pursuit algorithm is used to extract features from audio signals by decomposing signals into a set of atoms called a book. The number of atoms is linked to the duration of the acoustic event. To maintain the temporal and spectral information of atoms, the book is mapped into time-frequency histogram of fixed size. Random forest classifier is used to classify the spectro-temporal features, which trains decision tree classifiers on sub-samples of the dataset. For prediction, a simple vote is used for aggregation of the predictions.

The system was tested on a scenario comprising of four people, and one person suddenly screams for help. It was able to detect the screaming person correctly and send the robot to check the condition of that person.

3 Architecture

The system has a perception sensor network which currently has one sensor unit consisting of a microphone and a camera. This can be scaled up to have multiple sensor units as per requirements. Fig. 1 illustrates the general architecture of the hardware and software components of the system. The microphone in a sensor unit sends audio data to the central processing unit which is a Raspberry Pi. If the incoming audio crosses a configurable threshold value (set to 300dB currently), the audio is extracted as a file and sent to the cloud for classification. The cloud is a remote server containing the pre-trained ML model whose predict function is exposed as a web-service. This model classifies audio inputs into two categories - Scream (Emergency) or Random by giving the percentage of similarity to the audio in the training dataset. This percentage is sent back to the Pi and if it is greater than 50%, it triggers the video camera of the sensor unit for further scrutiny of the event. This event-based escalation technique has multiple advantages such as lesser energy consumption compared to a perpetual IOT system, decrease in false positives and also minimal intrusion of privacy. In case the event is classified as an emergency the activity result is passed on as a message to the MQTT component of the Safe Community Alerting

Figure 1: Architecture of the distributed system



Network (SCALE). The sound source classification is explained in further detail in the following section.

4 Project Setup

4.1 Machine Learning model

To classify if the recorded audio is actually a human scream, there was a need of a trained model which can differentiate between human scream and other noise. After doing a literature survey, it was observed that Support Vector Machines(SVM), Random Forest and various configurations of Neural Networks were used for the task at hand. Out of the available choice of algorithms, we began with building a simple model using SVM with the non-linear kernel as it has proved to be successful in two experiments discussed in two of the papers.[5][10]Further on, we implemented a 2D Convolutional Neural Network(NN) in Python using the Keras library which proved to be successful. This neural network is not too complex to make it scalable for the Raspberry Pi but scalable enough to give accurate predictions. It uses exponential linear unit(ELU) as the activation function and also uses max pooling for better training the model. The neural network took around 6 hours to train and the model was saved as a h5py file. This saved model was deployed to make predictions about the recorded audios. Our approach draws from experience by successful experiments in the paper by Huang et al [10] which show that the training based method works well even when there is substantial background noise.
For training the model, we had various options for the datasets, the aptest and easier to train is the "ESC-50: Dataset for environmental sound classification". The ESC-50 dataset is a labeled collection of 2000 environmental audio recordings suited for benchmarking methods of environmental sound classification. It has 5-second-long recordings arranged into 5 major categories. The other most viable option is the scream audio set provided by Google research. Any of the dataset we use for training the classifier, preprocessing still needs to be done. One such observation after examining the data is that the human scream sounds similar to crying and laughing because of various segments with amplified noise. Such data may produce false negatives, and it is hence required to apply the proper preprocessing techniques on the audio set. We ended up training our machine learning model using two datasets, namely, Google

AudioSet[8] and Urban Sound Dataset [7]. The Google AudioSet provides the screaming noises for training the machine learning model. The Urban Sound Dataset has various sounds classified as air conditioner, car horn, children playing, dog bark, drilling, gun shot and others. These two datasets formed the best combination for the binary classification task at hand and produced noteworthy results.

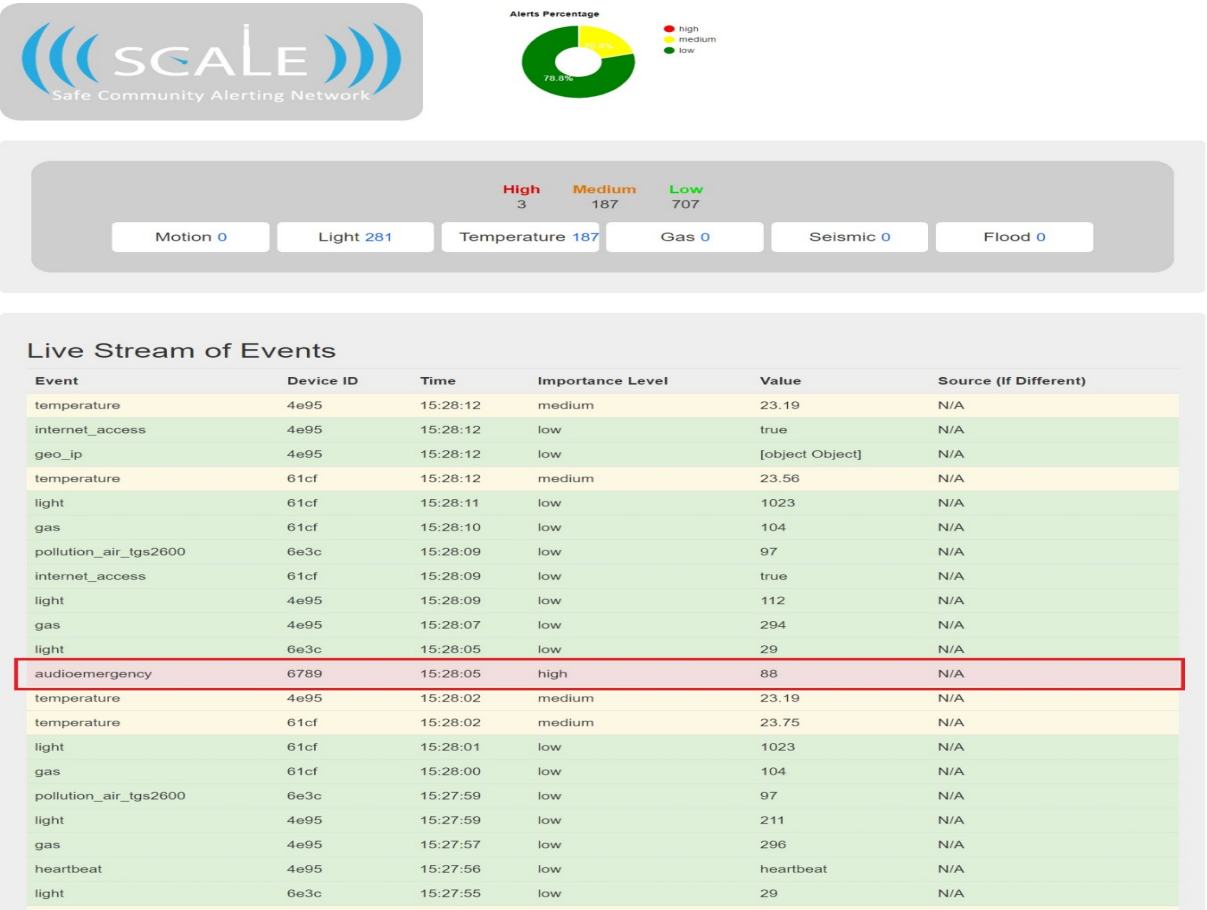
4.2 Raspberry Pi Setup

The Raspberry Pi we used for the experimentation has the following configuration: Quad Core 1.2GHz Broadcom BCM 2837 64bit CPU with 1GB RAM. Even though this is limited in terms of computational ability but we made sure, we were able to perform the requisite task using this. With the Raspberry Pi, we also used a USB microphone and a camera attached to the Raspberry Pi. The microphone is in a constantly listening state and it records the audio and sends it to the machine learning model only when its strength crosses a certain threshold. When the machine learning model predicts the audio as a scream, the camera starts and clicks a picture.

4.3 Sending data to SCALE

The image captured in the previous step along with the percentage of the audio being a scream is sent to the SCALE dashboard. This can be seen in Figure 2 where the red event is 'audioemergency' sent through the Raspberry Pi. This is published using the MQTT protocol. When the message is received by SCALE, the emergency services can be contacted. This is the last process in our escalation-based system.

Figure 2: Message Received on the SCALE Dashboard



5 Results and Evaluation

The machine learning model was evaluated on a test set and resulted in an accuracy of around 81%. This is significant for a non-trivial problem like this. We felt this model is complex enough and the accuracy achieved is good enough for the task at hand. A major roadblock which we tried handling throughout the project is to handle if computation is feasible on the Raspberry Pi. We feel better results can be achieved if higher computation ability is available. Also, libraries like TensorFlow Lite which are specifically suited to light devices is also a good alternative. We also found a limitation with the SCALE platform, that is, we cannot publish an image to SCALE if it is greater than 10 kilobytes. Lastly, as demonstrated in the demo video, we are able to successfully fulfill the task and we would love to scale our system for receiving data from multiple microphones and incorporate other different types of sensors as well.

6 Conclusion and Future Work

This project showcases the application of an ML model to detect emergency situations using audio and publishing data to the SCALE platform which acts as a dashboard for keeping track of all such events. Cameras which are connected to the central system will be triggered for image capturing to minimize false positives generated by the scream detection algorithm. This sensor network can serve as a good addition to the existing ecosystem of sensors present in SCALE which in turn can move one step closer to a robust smart-home or smart-assisted living solution.
Future work for this project includes shifting to more edge computing and further away from cloud computing. This involves the processing of audio signals on the raspberry pi itself to improve the performance of the system. With the analytics performed on the data obtained, we will classify events as emergencies and send residents alerts to confirm or reject the emergency. The current model which employs a convoluted neural network to classify audio can be made more accurate by further training with more data. The model can also be trained to identify stress keywords other than screaming. Link to demo for the project [1] [2].

References

[1] Demo video part 1. `https://drive.google.com/file/d/1dhpfxkpqHrqfA9NgaTwpGsn5Ecy3Mk9b/view?usp=sharing`.

[2] Demo video part 2. `https://drive.google.com/file/d/1dawSrgsem6_f4AZevZ4LkIMa8RRnlmUv/view?usp=sharing`.

[3] ALHASSOUN, N. S., UDDIN, M. Y. S., AND VENKATASUBRAMANIAN, N. Safer: An iot-based perpetual safe community awareness and alerting network. In *2017 Eighth International Green and Sustainable Computing Conference (IGSC)* (Oct 2017), pp. 1–8.

[4] BENSON, K., FRACCHIA, C., WANG, G., ZHU, Q., ALMOMEN, S., COHN, J., D'ARCY, L., HOFFMAN, D., MAKAI, M., STAMATAKIS, J., AND VENKATASUBRAMANIAN, N. Scale: Safe community awareness and alerting leveraging the internet of things. *IEEE Communications Magazine 53*, 12 (Dec 2015), 27–34.

[5] FENG, Y., CHANG, C. K., AND CHANG, H. An adl recognition system on smart phone. In *Inclusive Smart Cities and Digital Health* (Cham, 2016), C. K. Chang, L. Chiari, Y. Cao, H. Jin, M. Mokhtari, and H. Aloulou, Eds., Springer International Publishing, pp. 148–158.

[6] FENG, Y., CHANG, C. K., AND MING, H. Recognizing activities of daily living to improve well-being. *IT Professional 19*, 3 (2017), 31–37.

[7] FREESOUND.ORG. Urbansound 8k dataset. https://urbansounddataset.weebly.com/urbansound8k.html.

[8] GOOGLE. Google screaming audioset. https://research.google.com/audioset/dataset/screaming.html.

[9] NGUYEN, Q., YUN, S., AND CHOI, J. Detection of audio-based emergency situations using perception sensor network. In *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* (Aug 2016), pp. 763–766.

[10] WEIMIN HUANG, TUAN KIANG CHIEW, HAIZHOU LI, TIAN SHIANG KOK, AND JIT BISWAS. Scream detection for home applications. In *2010 5th IEEE Conference on Industrial Electronics and Applications* (June 2010), pp. 2115–2120.