Demo Abstract: Acoustic Anomaly Detection System

Jae-wook Ahn, Keith Grueneberg, Bong Jun Ko, Wei-Han Lee, Eduardo Morales, Shiqiang Wang, Xiping Wang, David Wood*
{jaewook.ahn,kgruen,bko,wangshiq,xiping,dawood}@us.ibm.com,{wei-han.lee1,eduardo.morales}@ibm.com
IBM T. J. Watson Research Center, Yorktown Heights, NY, USA

ABSTRACT

Acoustic signals contain rich information of the environment. They can be used for detecting anomalous events such as in automated machine monitoring. In this demonstration, we present our acoustic anomaly detection system that captures acoustic signals and classifies them using machine learning techniques. Our system includes a server for sound management and model training, a mobile client for sound capturing and real-time classification, and a workbench that acts as a user interface. We will show the full operational pipeline of our system in this demonstration.

CCS CONCEPTS

• Computing methodologies → Supervised learning by classification; **• Computer systems organization → Client-server architectures**.

KEYWORDS

Acoustic signal analytics, anomaly detection, machine learning

ACM Reference Format:
Jae-wook Ahn, Keith Grueneberg, Bong Jun Ko, Wei-Han Lee, Eduardo Morales, Shiqiang Wang, Xiping Wang, David Wood. 2019. Demo Abstract: Acoustic Anomaly Detection System. In *The 17th ACM Conference on Embedded Networked Sensor Systems (SenSys '19), November 10–13, 2019, New York, NY, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3356250.3361963

1 INTRODUCTION

Acoustic signals carry rich information of our environment. They can be used to detect environmental events through acoustic signal analytics. For example, when a washing machine starts making a rattling sound, it may stop working soon. If a car makes a bad-bearing sound on a wheel, there is something wrong on it. If we use acoustic sensors to automatically monitor the sound of equipment and detect functional anomalies, maintenance services could be in place as soon as the equipment shows an unusual sign, thus preventing the damage or loss of valuable equipment. In another scenario, patients with different medical conditions, such as pneumonia, bronchiolitis, and asthma make different coughing sounds. By analyzing the coughing sounds, we could separate pneumonia from other diseases and hence provide patients with proper treatment in a timely manner.

In recent years, mobile sensing is a very active area of research [1, 2]. With the increase of mobile device's computing power, mobile acoustic sensing, analytics, and sense-making have been proven as a promising approach for automated machine monitoring for various applications. We have designed and built an acoustic monitoring system using machine learning techniques that can operate on mobile devices, aiming at enabling automated machine monitoring for different environments at a large scale. The system is fast enough to process large volumes of acoustic data and can be applied to a broad range of applications, such as home appliance anomaly detection, vehicle fault diagnosis, manufacturing equipment failure detection, building maintenance, health care, etc. In this demonstration, we present the architecture of our system and a live demonstration of ambient sound detection and classification using mobile phones. We also demonstrate how sounds are labeled, how acoustic models are trained using machine learning techniques, and how the trained models are tuned to improve classification accuracy.

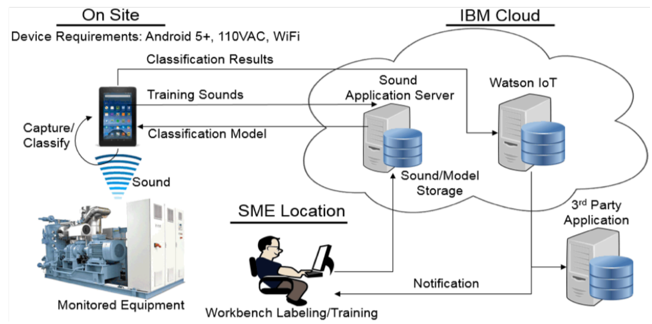

Figure 1: System Architecture.

2 SYSTEM ARCHITECTURE

The system architecture is shown in Fig. 1. A mobile client connects to a cloud-based server that provides functionalities for storing labeled sounds and trained models, as well as a workbench (web-based user interface) for managing the sounds and models. The mobile client is equipped with a software to perform sound capturing, sound labeling, communication with server, on-device feature extraction, and on-device classification. The server performs model training and (server-side) sound classification. The mobile client can perform local on-device classification by retrieving a trained model from the server. After acquiring a model, the client can perform continuous monitoring using either a local model or a server-based model. Optionally, the client can publish the classification results to a Watson IoT instance[1] to enable custom notification and alerting when an anomaly is detected.

*Authors listed alphabetically by last name.

[1] https://developer.ibm.com/tutorials/cl-mqtt-bluemix-iot-node-red-app/

SenSys '19, November 10–13, 2019, New York, NY, USA

ACM ISBN 978-1-4503-6950-3/19/11…$15.00
https://doi.org/10.1145/3356250.3361963

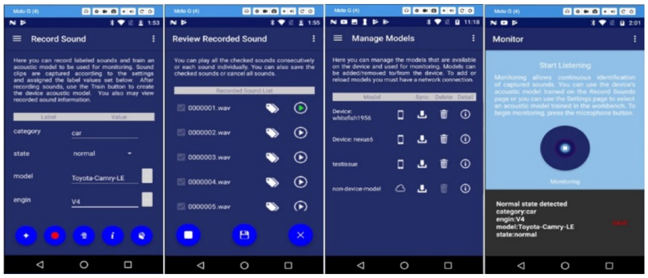

Figure 2: Mobile client user interface.

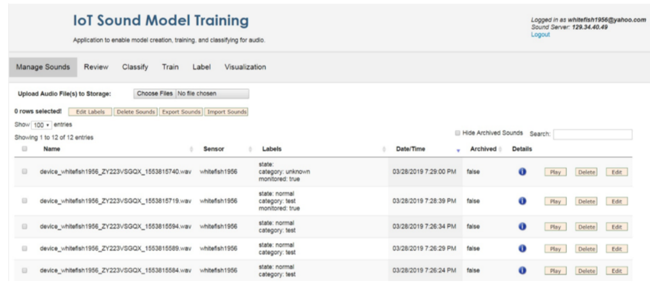

Figure 3: Workbench view.

3 SYSTEM OPERATION

3.1 Client

Fig. 2 shows the client interfaces. In order to train an accurate model, the training sounds must be rich, balanced, and correctly labeled. The client effectively manages the labeling of training sounds by providing mandatory and custom label types. A sound data sample may contain multiple types of labels. After sounds are recorded, the user can verify if they are properly labeled through a sound review page (screenshot 2 in Fig. 2). To support the system working without Internet connection, the client is designed to support offline recording and classification so that the user can record and label sounds anywhere and perform on-device classification after a model is retrieved from the server. The local models are mirrored from the server and managed by the client. These features have been proven particularly useful for trials in fields or remote facilities. Furthermore, to improve the model accuracy, the client can be configured to optionally collect the monitored sounds (screenshot 4 in Fig. 2) at a specified interval. These sounds are sent to the server, so that the server can keep updating the existing model.

3.2 Server

The server builds a model per request of a client or workbench after it receives all the training sounds. To improve the classification accuracy, the server is designed to support an ensemble of classifiers. Various models are combined with various feature extractors [3] in the ensemble training phase. The model and feature extractor combination that gives the highest accuracy on a held-out test dataset (which is randomly extracted from the training data) is used for classification. The models include Gaussian mixture model (GMM), k nearest neighbor (k-NN), etc., and the feature extractors include fast Fourier transform (FFT), mel-frequency cepstral coefficients (MFCC), etc. Potentially anomalous sounds detected by the system are captured in “work orders” and can later be confirmed by a human whether the sound was indeed anomalous. After confirmation, the model is updated to include the new (labeled and confirmed) training sounds. The trained models are stored at the server and can be downloaded by clients for on-device classification.

3.3 Workbench

Fig. 3 shows a workbench view. Since the server hosts a large amount of sounds and models, it is necessary for the users to manage them in an effective way. The web-based workbench is designed for this purpose, allowing the user to interact through the user interface that communicates with the server through a REST API. The workbench is able to import, review, export sounds, edit labels, train models, classify sounds based on a trained model, and visualize the data and labels. In our system, each device has its own models, and users can build and train models from the workbench. All the users’ models, either from the device or workbench, can be shared across his/her devices. The monitored sounds stored by the mobile device can be reviewed and relabeled using the workbench, so that an existing model can be updated through the workbench by adding new sounds to the existing training data set. In addition, the workbench has advanced features that helps efficient data management, including automatic labeling of sounds, automatic splitting or clustering long recordings into smaller clips to assist manual labeling, and visualization of data with respect to ontology relationships of sound labels.

4 DEMONSTRATION

We will demonstrate our system’s capability to detect anomaly from sounds, using a mobile device and a server. We will show how a user captures the training data using the mobile device, labels and manages the data, trains classifier models, and monitors new incoming sound so that the system classifies them (e.g., into normal or abnormal) in real time. The data management features including audio clustering and visualization in the web-based workbench as well as on the mobile device will also be shown.

ACKNOWLEDGMENTS

This research was partly sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-16-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] Chandrasekaran et al. 2016. SEUS: A Wearable Multi-Channel Acoustic Headset Platform to Improve Pedestrian Safety. In *Conference on Embedded Network Sensor Systems CD-ROM*. ACM.

[2] Lane et al. 2015. DeepEar: robust smartphone audio sensing in unconstrained acoustic environments using deep learning. In *International Joint Conference on Pervasive and Ubiquitous Computing*. ACM.

[3] Urmila Shrawankar and Vilas M. Thakare. 2013. Techniques for Feature Extraction In Speech Recognition System : A Comparative Study. *CoRR* abs/1305.1145 (2013). arXiv:1305.1145 http://arxiv.org/abs/1305.1145