

Spatiotemporal Analysis of Urban Noise Data

Ashwin Bhaskar Srivatsa, Sasanka Mouli Veleth and Sidharth Veluvolu

1 INTRODUCTION

Noise Pollution is a grievous problem which needs to be tackled, as it's a growing concern for many urban residents. Sounds that are not particularly loud but are nonetheless undesirable and uncontrollable can have serious implications for the listener, particularly if they occur over a long period of time [7]. The noise becomes much more upsetting if the source of noise is an agent or agency that has shown little concern for the individual who is suffering from the noise's effects and, as a result, nothing has been done to reduce the noise. Noise is not only inconvenient and annoying, but it has also been proven to be a health hazard, many have reported that they suffered with behavioral and emotional consequences, such as difficulty in sleeping, relaxing and feeling annoyed, angry or upset [3], and when intrusive noises continue, the body responds physiologically, and there is a risk of irreversible bodily damage - damage to the circulatory, cardiovascular, and gastrointestinal systems - over time [4].

In order to identify the source and mitigate this problem, there is a need to understand sound event detection [2]. Sound event detection is defined as recognition of individual sound events in audio, e.g., "dog barking, engine exhaust noise" requiring estimation of onset and offset for distinct sound as for identification of sound. There are multiple challenges for sound event detection and classifying them into various classes especially when it is to detect in multiple environments or large sets of data, or when there is overlapping of sound events or mixture of sound signals from sensors or identifying the similar sounds compared to the other distinct sound that is harder to classify the source [8]. So, it's important to evaluate the data set with actual recordings from urban noise sensors and to identify the sounds by classifying them into classes or labels. In this article we take the large data set which comprises of 3068 labeled 10 sec recordings from the Sounds of New York City (SONYC) acoustic network [1] (An acoustic network is a method of positioning equipment using sound waves). Using this data, we plan to develop a machine learning model that classifies the audio sounds into particular categories based on the attributes or features and also further analyze it to find out the source. We also plan to address the mismatch of the testing data in this data set.

This can be done by applying machine learning algorithm on the data set and then analyzing it for the development of machine learning systems for real world urban noise monitoring. The timeline and work flow (Fig. 2.) for this is depicted below where The model that is devised from the data can be used on urban neighborhoods to better understand noise in that location and help the authorities mitigate this issue.

2 RELATED WORK

Collecting audio data and annotating the soundscapes are an important part of acoustic research especially in the situations with high variability in different locations.

In the paper [6] they have discussed about soundscapes and different ways to annotate audio data using crowdsourcing. There are basically two ways to achieve this one of them being waveform and the other one being spectrogram visualization. Certain annotation trials were done using the two techniques and interesting results were drawn from the

experiments. People using the spectrogram visualization technique were able to produce high quality and precision annotations than waveform visualization.

According to [2] a supervised learning methodology is applied to real-life high quality recordings of 3-5 minutes with very little noise of 15 different acoustic scenes (lakeside beach, bus, cafe/restaurant, car, city center, forest path, grocery store, home, library, metro station, office, urban park, residential area, train, and tram) and two common environment areas (outdoor - residential areas and indoor - home). A Mel frequency Cepstral coefficient (MFCC) and Gaussian mixture model (GMM) was trained using expectation maximization algorithm. The overall accuracy achieved by the model is 72.5 % ranging from 13.9% for parks to 98.6% for office spaces. With our system we plan to include cross-validation to train our model so that there is no data contamination between train and test sets.

According to the work done in [5], audio annotation is critical for creating machine-listening systems, but there is little research on how to get accurate and timely crowd sourcing audio annotations. They aimed to quantify the reliability/redundancy trade-off in crowd sourced soundscape annotation, look at how visualizations affect accuracy and efficiency, and characterize how performance varies as a function of audio characteristics.

Our application will be based on the research paper's data [5], which presents us the process to collect acoustic data. SONYC has developed an acoustic sensor with high quality and low production cost to monitor the noise pollution levels across the city in neighborhoods like Manhattan, Brooklyn and Queens. The collected data was then annotated using a campaign on Zooniverse. The sensors follow DCASE (Detection and classification of acoustic scenes and events) to eliminate discrepancy. There are various other datasets like UrbanSound, UrbanSound8k that address this particular problem but have limited spacial and temporal data points. A VGGish model has been developed and trained using stochastic gradient descent to minimize cross-entropy loss. To eliminate over-fitting early stopping on validation set has been implemented. Two models were trained on coarse-level and fine-level tags. The overall AUPRC achieved by this model is 0.62 and 0.76 on different level classes, which performed poorly on music and non-machinery impact sounds. We plan to use this model and analyze the mismatches caused by the prediction on actual test data and find out the causes which led to these mismatches.

Over the years there has been a lot of research on annotation of audio data in different scenarios and predicting the source of noise in big cities. We will be extending this to analyze the mismatch of such predictions by leveraging machine learning metrics and coming up with our own model to predict the results with higher accuracy. A big part of our research will also be visualizing the locations of these mismatches. However most of our work focuses on analyzing existing models for the (SONYC-UST) dataset [1].

3 DATA DESCRIPTION

In order to build a system which would visualize the various sound points in and around a particular geographical location and also allow us to analyse the mismatches between the test and machine data, we need to have a dataset which has a diverse distribution of labeled sounds with spatial and temporal attributes. For which we have taken a dataset containing a training subset (13538 recordings from 35 sensors), validation subset (4308 recordings from 9 sensors), and a test subset (669 recordings from 48 sensors). Each recording has been annotated using a set of 23 "sound tags" like "engine presence, machinery presence,

-
- Ashwin Bhaskar Srivatsa, E-mail: asriva36@uic.edu
 - Sasanka Mouli Veleth, E-mail: sveleth2@uic.edu
 - Sidharth Veluvolu, Email: sveluv2@uic.edu

non-machinery-impact presence, dog-barking-whining presence, music presence etc.” [5].

The training, validation, and test subsets of the annotation data are contained in annotations.csv (see Figure 1). Each row in the file represents one multi-label annotation of a recording—it might be a single citizen science volunteer’s annotation, a single SONYC team member’s annotation, or the SONYC team’s agreed-upon ground truth (for more information, see the annotator id in column description). The audio files used were recorded using the SONYC acoustic sensor network for monitoring urban noise pollution. In New York City, over 60 distinct sensors have been placed accumulating the equivalent of more than 50 years of audio data, of which a small fraction is used. The data is sampled by picking the closest neighbors based on VGGish qualities of recordings with recognized classes of interest. All of the recordings are 10 seconds long and were made with the same microphones and gain settings.

The sensors in the test set will not disjoint from the training and validation subsets, but the test recordings are displaced in time, occurring after any of the recordings in the training and validation subset. We plan to use the test data to find out the aggregate of mismatch by using a Multi Label classification Machine Learning Model like support vector machines and artificial neural networks.

In our work we will utilize the latitude and longitude columns to spatially locate the sound origin, the year, week, day, hour columns to precisely point the time and the category column which contain engine, machinery-impact, non-machinery-impact, powered-saw etc.to get the sound type to analyze and visualize the data.

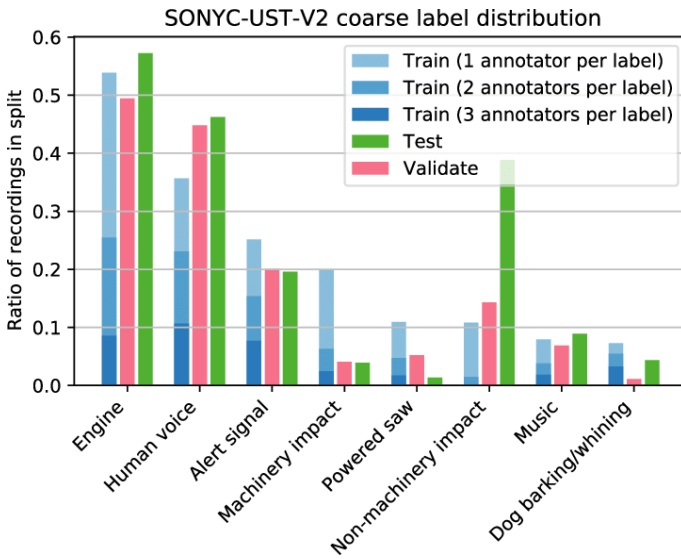


Fig. 1. Distribution of various sound tags in test, train, validate sets of dataset

4 RESEARCH

We propose to build a tool which will visualize the spatiotemporal values of the dataset, visualize the output of Machine Learning model and also visualize the points where the mismatches of testing and machine data occur based on the results of the Machine Learning model. We will be using Multi Label Classification Machine Learning Algorithms like support vector machines and existing neural networks to classify the audio files into various categorical sounds. Based on the results of ML model, we will dwell into understanding the mismatches which occur between the annotated and machine predicted data and analyze the causes behind it.

The technologies which we plan to use in order to build this tool would be ReactJS on the front-end and Django on the back-end. We are using ReactJS because of its component based architecture and its rich support with D3.js, Open Layers. Similarly, Django allows

us to run various python libraries to process the data and creation of REST API’s (representational state transfer application programming interface) to communicate with the front-end.

We have divided the process of building this tool into four stages (see Figure 2) namely data preprocessing stage where we process the audio files, Modeling stage where we apply the classification models on the dataset and create REST API’s to interact with the front-end, Visualization stage where we visualize the results of machine learning model on the front-end, component building stage where we build components which allows the user to pick spatial and temporal values of various sound samples and display the results on a map.

This tool would be useful to authorities who monitor sounds around the city which will allow them to pinpoint the location of sound origin.

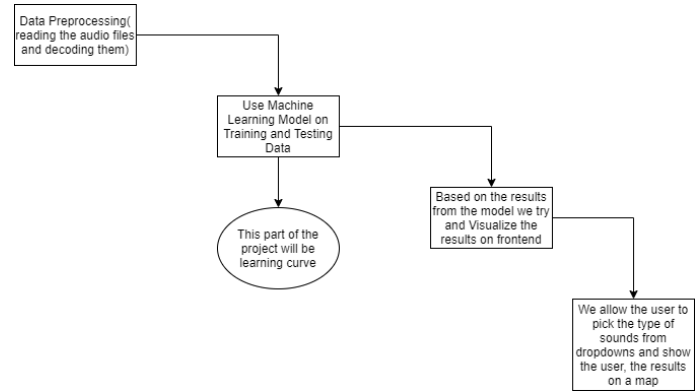


Fig. 2. Work flow diagram

5 RESULTS

6 CONCLUSION

REFERENCES

- [1] Sonyc urban sound tagging (sonyc-ust): a multilabel dataset from an urban acoustic sensor network.
- [2] T. V. Annamaria Mesaros, Toni Heittola. Tut database for acoustic scene classification and sound event detection.
- [3] A. Bronzaft. Neighborhood noise and its consequences.
- [4] T. K. S. M. S. Hammer and R. L. Neitzel. Environmental noise pollution in the united states: developing an effective public health response, 2013.
- [5] J. C. V. L. G. D. H.-H. W. J. S. O. N. Mark Cartwright, Ana Elisa Mendez Mendez1 and J. P. Bello. Sonyc urban sound tagging (sonyc-ust): A multilabel dataset from an urban acoustic sensor network. Detection and Classification of Acoustic Scenes and Events 201, October 2019.
- [6] J. S. A. W. S. M. D. M. E. L. J. P. B. Mark Cartwright, Ayanna Seals and O. Nov. Seeing sound: Investigating the effects of visualizations and complexity on crowdsourced audio annotations.
- [7] W. H. Organization. Burden of disease from environmental noise: Quantification of healthy life years lost in europe, 2001.
- [8] A. E. T. V. Toni Heittola, Annamaria Mesaros. Context-dependent sound event detection. EURASIP Journal on Audio, Speech, and Music Processing, January 2013.