Acoustic Event Detection Using Machine Learning: Identifying Train Events

Shannon McKenna David McLaren

INTRODUCTION

Light-rail systems are becoming more popular in cities and urban residential areas around the country. One of the main environmental impacts from light-rail systems is noise from the trains as they pass through residential areas. In response to increasing noise complaints, it is becoming more common to perform noise measurements in the residential areas and attempt to identify noise mitigation solutions based on the results. Currently, most of the noise measurements are attended with a technician keeping a log of when trains pass the measurement location so those train events can be extracted from the continuous recording during post-processing. This method limits the amount of data that can be collected due to limited man hours. A machine learning algorithm that identifies train events in a continuous noise recording would increase efficiency during both data collection and post-processing.

ATS Consulting has a significant body of noise recordings adjacent to light-rail tracks. The goal of the project is to develop a supervised learning algorithm that can separate the train events from the background noise with a high success rate. We applied logistic regression and support vector machine (SVM) learning algorithms to attempt to solve this problem. For the purpose of train noise analysis, it is acceptable for the algorithm to fail to identify about one in ten train noise events, especially if they are corrupted by simultaneous environmental noise sources (for example, a lawn mower); however, the algorithm should have a much lower error rate for classifying other noise events (such as a car) as a train event. Misclassifying events as trains could lead to misdiagnosing the cause of loud noise levels and recommending inappropriate mitigation measures, while failing to identify some events as trains is merely weeding out corrupted samples that would most likely not be included in future analysis.

DATA AND LABELING

The training data for this project consists of noise measurements conducted along the Exposition light-rail line in Los Angeles, California. At each of the measurement sites, a 2-3 hour noise measurement was recorded, each including 10 to 22 train passbys. At two of the measurement sites, a twenty-four hour measurement and a second two-hour measurement were recorded.

All of the short-term (2-3 hour) measurements were attended by a technician keeping a log sheet of the time of all train passbys. The labeling of our training data (train or not train) was based off of the notes in these log sheets. The long-term (24 hour) measurements were unattended (no record of train passbys). The data from the unattended measurements has been inspected by hand post-recording to identify train events. The labeling of our long-term data was based on this post-recording inspection, and may include labeling error. For these long term data sets, we expect that some true train events have gone unidentified. Due to the possibility of labeling error in the long-term data, it was not included in the training data set, and was only used for testing the trained classifiers.

The noise measurements were recorded as .WAV files; however, the data is available in text file format with the overall A-weighted noise levels and the 1/3 octave band levels extracted with a RMS time-step of 0.125 seconds (with the exception of one long-term measurement at site 11 that had an RMS time-step of 0.25 seconds). The A-weighted sound level is the overall loudness weighted to approximate the hearing of the human ear. The frequency content (available in 1/3 octave bands) is the pitch of the noise. Figure 1 (left) shows a sample spectra of train events recorded at Site 8. The spectra of background noise is the bolded black line in the figure. The spectra vary from train to train depending on the direction the train is

traveling and the speed of the train; however, all of the spectra have generally the same shape, with noise levels peaking in the 800 Hz and 1000 Hz 1/3 octave bands.

A typical train passby in our data set is approximately 10 to 15 seconds long (as shown in Figure 1 on the right). For a data set with 0.125 second time step, about 80 data points will be included in one train event. Our approach to the problem of identifying train events is to run a machine learning algorithm on each time step of each recording and label each time step as a train or not a train. After classifying each individual time step, we look for clusters of positive examples where a significant percentage of the data points are classified as train events. These clusters of positive examples are then labeled as a single train passby. We chose this approach to guard against our results being sensitive to misclassifications of individual data points.

To label each data point in the training data set, we had to define a precise start and end point to all of the train events. Using only the logsheet notes, data points near the beginning and end of the event were labeled as part of the train event even before the train noise exceeded the background noise. In practice, a train noise event is often defined by its 3dB or 10dB downpoints [US Dept. of Transportation]. We tried two approaches to cull ambiguous data points from our training set: (1) removing the data from the ends of the manually labeled event to the 10 dB downpoints, and (2) removing a constant fraction of the labeled train events from each end of the event.

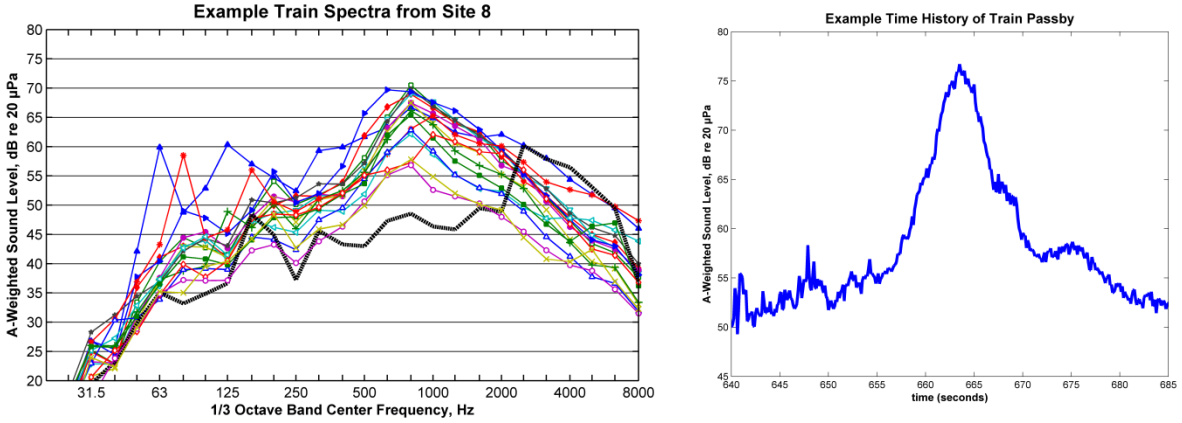

Figure 1: Example Spectra of Train Passby (left) and Time History of a Train Passby (right)

LOGISTIC REGRESSION

We first implemented a Naïve Bayes classifier with discretized 1/3 octave band features, but soon moved on from this approach in favor of logistic regression so we could work directly with the continuously valued features (the 1/3 octave band noise levels). In our first regression implementation, we used the 1/3 octave band noise levels in the 200 Hz to 4000 Hz 1/3 octave bands. The noise from trains in bands outside of this range is generally not above the background noise level, as seen in Figure 1. The performance of this first iteration of our classifier was mediocre, identifying only about half of the train events. Performance of our classifier was significantly increased when we normalized the noise levels by subtracting out the mean noise level of each 1/3 octave band and dividing by the standard deviation. The normalization significantly decreased the false positive identification rate of measurement locations with high background noise level (a low signal-to-noise ratio).

Because the 1/3 octave band noise level can vary significantly depending on the speed of a train or the distance from the microphone to the train, we were also interested in including features that did not depend solely on the noise level. The train noise spectra typically have a very consistent shape, with peak noise levels showing up in the 800 or 1000 Hz 1/3 octave band levels. Including features that capture the shape of the spectrum, rather than just the magnitude of the 1/3 octave band levels may improve the performance of our classifier [Clavel, Ehrette, and Richard]. Possible features include the first and second

spectral moments as features. The first and second spectral moments are the mean frequency and the frequency variance. We experimented with adding these features. In addition, we tried adding a feature that was the product of the normalized levels in what we expected to be the peak 1/3 octave bands for train events (800 Hz, 1000 Hz, and 1200 Hz) and a feature of the peak 1/3 octave band.

The preliminary results from our model are presented in Table 1. Results are presented for our analysis with our original features, which included only the normalized 1/3 octave band levels and A-weighted noise level, and for our analysis that added extra features in an attempt to capture the shape of the spectrum (including features such as the mean frequency and variance of the frequency).

Following are some observations from the results presented in Table 1:

- As expected, the logistic regression classifier performs best on sites that have the highest signal-to-noise (SNR) ratio. Sites with high SNR are 1, 2, 6, 7, 9, and 10. Sites with a low SNR are 2, 3, 4, 8, and 11.
- In general, the extra features (mean frequency and the product of sound levels in the 800 Hz, 1000 Hz, and 1250 Hz 1/3 octave bands) did not improve the performance of the algorithm. There was little to no change in the true-positive rate. There was a more consistent decrease in the false-positive rate at sites with a low SNR; however the false positive rate was low to begin with.
- The true positive rate is unacceptably low for sites with low SNR. However, there is not a significant increase in the false positive rate for sites with low SNR.

An encouraging result is the number of false positives is relatively low for most of our sites, which was one of our design goals. One option for improving the number of true positive events identified is including a ridge parameter to reduce our sensitivity to outliers. Optimizing a ridge parameter may allow us to increase our true positive rate, without increasing the false positive rate to the point where we are false positives occur in a large enough cluster to be classified as an event. However, this was not implemented during the quarter. Another avenue for improving the performance may be to do more research on features that are less dependent on the SNR—for example, features that take into account how the spectral values change with time. Currently, we are not exploiting the fact that the noise levels rise and decrease with a very regular pattern as the trains approach and pass the microphone location.

Table 1: Summary of Logistic Regression Results

Site	Location	Total number of events	Number of correct positive events		Number of false positive events		True Positive Rate Data points		False Positive Rate- Data points	
			Orig. Features	W/ Extra Features	Orig. Features	W/ Extra Features	Orig. Features	W/ Extra Features	Orig. Features	W/ Extra Features
1	Caroline	12	12	12	1	0	0.45	0.54	0.06	0.11
2	Catalina	20	16	11	0	0	0.32	0.27	0.02	0.01
3[(a)]	Catalina2	12	10	7	1	0	0.33	0.25	0.03	0.01
4	Cimarron	10	6	**7**	1	0	0.18	0.25	0.13	0.01
5	Cloverdale	22	22	22	1	0	0.80	0.79	0.04	0.04
6[(a)]	Cloverdale2	12	12	12	8	6	0.80	0.79	0.69	0.68
7	Fay	13	13	13	2	2	0.66	0.70	0.09	0.11
8	Hillcrest	17	17	17	3	1	0.70	0.66	0.44	0.42
9	Redondo	12	12	12	0	0	0.93	0.95	0.14	0.17
10[(a)]	Cloverdale Long-Term	140	140	140	18	18	0.83	0.80	0.28	0.25

11[(a)]	Catalina Long-Term	143	7	7	0	0	0.05	0.25	0.01	0.01
Notes:[(a)]The Catalina2, Cloverdale2, Catalina Long-Term, and Cloverdale Long-term data sets were not included in our training data.										

SVM

We used the libsvm implementation of Support Vector Machines [Chang & Lin] to search for train events in a similar manner as we did with logistic regression, by classifying individual timepoints and then discerning events from clusters of points. An SVM may improve performance by using a kernel to take advantage of higher dimensional features. Intuitively, we believe train events are most easily identifiable at their peak volume levels, when the train is closest to the recorder. The volume during the crossing will rise to a peak over some number of seconds, and then fall off, and the peak decibel level of a train passby is usually much louder than background noise levels across many features. We expect that these peak noise levels will be separable from the levels experienced during non-train events.

To produce our training data, we reserved five recordings as test data which were not used in training, including both 24-hour recordings, and trained using all data from the middle 70% of each train passing (to exclude datapoints which would be harder to distinguish from background noise). To avoid biasing the classifier by providing too many negative examples [Ben-Hur & Weston], we only included 15% of all datapoints from outside train crossing in our training data, as train crossings were relatively rare. To normalize data, we zeroed out the mean of the decibel levels for each octave band, and divided by standard deviation, as in the logistic regression implementation. (We also attempted to normalize values by scaling the 1st and 99th percentile values of each feature to the range 0-1, but this produced poor performance).

Early in development, we performed a forward search to try to trim the set of features to those which gave the best performance, but did not find this to be a good time investment. Forward search did seem to confirm that frequencies between 800 and 2000 Hz were the most useful for distinguishing events, as we suspected; but after normalizing our input data, we did not find a decrease in accuracy when classifying using data from all octave bands. We also attempted classification with the extra features added for logistic regression (variance, and products of the 800, 1000, and 1250 Hz bands) but did not notice a significant improvement in performance.

We trained and tested an SVM using a Gaussian kernel for different values of δ with the cost for all classes fixed to its default value (c = 1). In our experiments we found that a Gaussian kernel with δ = 0.01 gave the best performance for this problem, and that linear kernels also performed well. In Table 2, we present results from testing for two cases: (1) using all frequencies at a timepoint as input features, and (2) using only the eleven audio bands between 315 and 3150 Hz. Finally, to find train crossings in the SVM output, we returned intervals where 80% of all data points were positively classified within a 6-second timespan.

Table 2: Summary of SVM Results

Site	Location	Total number of events	Number of correct positive events		Number of false positive events		True Positive Rate Data points		False Positive Rate- Data points	
			All Features	Subset Features	All Features	Subset Features	All Features	Subset Features	All Features	Subset Features
1	Caroline	12	12	12	0	0	0.66	0.61	0.01	0.01
2	Catalina	20	19	20	1	3	0.69	0.67	0.01	0.02
3[(a)]	Catalina2	12	10	11	0	0	0.60	0.58	0.01	0.02
4	Cimarron	10	8	9	0	0	0.56	0.47	0.03	0.04

5	Cloverdale	22	22	22	0	0	0.75	0.74	0.01	0.01
6[(a)]	Cloverdale2	12	11	11	5	6	0.84	0.85	0.02	0.03
7	Fay	13	12	12	1	2	0.71	0.66	0.01	0.01
8	Hillcrest	17	17	17	0	4	0.82	0.75	0.02	0.03
9[(a)]	Redondo	12	12	12	0	0	0.89	0.91	0.01	0.02
10[(a)]	Cloverdale Long-Term	140	138	138	25	23	0.69	0.70	0.02	0.02
11[(a)]	Catalina Long-Term	143	97	74	29	6	0.32	0.25	0.01	0.01
Notes:[(a)] The Redondo, Catalina2, Cloverdale2, Catalina Long-Term, and Cloverdale Long-term data sets were not included in the SVM training data										

We see that as with logistic regression, it is very difficult to find train crossings in the Catalina long-term recording without identifying many false positives. The Cloverdale2 test case also stands out again as being a difficult case. We could adjust the costs of misidentifying the two classes to tweak results, but there is not much leeway for improvement over all cases—increasing the penalty for false positives or negatives improves results for some cases at the expense of others. The overall performance in finding events is similar to logistic regression, except that the SVM was able to find many more train events during the long-term Catalina recording.

CONCLUSIONS

The logistic regression and SVM classification methods both show some promise in correctly classifying train events and show very similar results. For both methods, the number of correct positive events identified decreases with decreasing SNR. Somewhat surprisingly, the number of false positive train events is quite high for the data Site 6 and 10 which had high SNR, but were not included in the training data.

We did not meet our original goal of positively identifying nine out of ten train events for sites with low SNR. However, based on the success with identifying at least nine out of ten events for sites with high SNR, implementing more sophisticated features into either of the two classification methods may yield satisfactory results. These features may include methods of extracting more information about the shape of the spectrum, or including features that have to do with how noise levels change over time for a train event. Time series classification also seems promising for identifying events and incorporating information about the shape of the audio signal. [Wei & Keogh] claim one-nearest neighbor time-series classification with a Euclidean distance metric to be very competitive; we experimented with this algorithm but did not manage to improve performance to the same level as the SVM within the quarter.

REFERENCES

A. Ben-Hur and J. Weston. *A User's guide to Support Vector Machines*. In Biological Data Mining. Oliviero Carugo and Frank Eisenhaber (eds.) Springer Protocols, 2009.

C-C. Chang and C-J. Lin. *LIBSVM: a library for support vector machines*. 2001. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

C. Clavel, T. Ehrette, and G. Richard. *Events Detection for an Audio-Based Surveillance System*, ICME 2005 IEEE International Conference on Multimedia & Expo. pp.1306-1309. 6-6 July 2005.

US Dept of Transportation. *Transit Noise and Vibration Impact Assessment*. Federal Transit Administration. Document FTA-VA-90-1003-06. May 2006.

L. Wei and E. Keogh. *Semi-Supervised Time Series Classification.* Proc. of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. Pp. 748 – 753. 23 August 2006.