AutoAudio: Sound-Based Vehicle Classification

Redford Hudson, Jiaye Liu, Lazar Salvador-Smith, Yuqian Ma, Xingjian Wang
University of California, Irvine
rfhudson@uci.edu, jiayel7@uci.edu, lcsalvad@uci.edu
yuqianm1@uci.edu, xingjiw4@uci.edu

Abstract

Smart city initiatives gives us a reason to explore advanced technologies for effective urban management. Automatic vehicle classification using acoustic signals, a project that our team developed, is a cost-effective alternative to visual methods—which traditionally faced challenges such as high costs and environmental limitations. AutoAudio utilizes deep learning to classify vehicle sounds, and extracts statistical and spectral features from datasets from New Delhi (MVDA) and Munich (IDMT) to ensure our interobserver accuracy across different urban settings. Using models like LSTMs and CNNs, AutoAudio enhances the future of smart urban systems through an accurate classification model that is capable of learning complex audio patterns.

1 Introduction

The fast urbanization and implementation of smart city initiatives have led to a strong demand for cutting-edge technologies that can effectively manage urban infrastructure. Among them, automatic classification of ground vehicles using acoustic signals is one of the most important and interesting problems. The current state of the art of vehicle classification methods mostly considers visual data from cameras or LiDAR sensors. Although we have more effective technologies, they are not without their limitations, for example, high cost, privacy, and sensitivity to environmental conditions such as poor lighting or obstructions (Tran et al., 2021) In contrast, acoustic sensing is a cost-effective, non-intrusive technique that can operate in a wide range of environmental conditions, including areas of total darkness and behind obstacles (Alfonso et al., 2018).

AutoAudio is a major development in the area as it uses machine learning algorithms to find and classify sounds produced by a vehicle. The study uses two extensive datasets from New Delhi, India (MVDA), and Munich, Germany (IDMT) covering various vehicle types and recording conditions. With the dual dataset approach, we could ensure the robustness and generalizability of the models developed for urban flow management across different kinds of urban environments.

The methodology used in AutoAudio consists of sophisticated feature extraction algorithms and top-tier machine learning models, like Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTMs) based models. These models have been proven to perform extremely well in terms of modeling the complex patterns in audio signals and to classify them accurately (Hershey et al., 2017).

AutoAudio is a powerful step beyond standard visual-based approaches; it challenges the status quo and presents a series of cutting-edge sensing methods that could enable future intelligent and context-aware urban systems. The project demonstrates the opportunity offered through the fusion of machine learning with acoustic data to create more intelligent, safer and cleaner urban environments.

2 Data

We are using two acoustic vehicle datasets: IDMT-Traffic[1], and MVD[2].

Both of these datasets contain 2-3 second long audio clips of vehicles passing by, recorded by microphones on the side of the road. Each dataset covers 4 distinct recording sites, making a total of 8 recording sites to analyze. Both datasets also cover multiple times of day and weather conditions. Finally, both datasets record 4 distinct vehicle classes, making a total of 5—car, bus, truck, bike, other (pedestrian). Relevant features are embedded into the file name, and can be incorporated into the machine learning classification module.

Given the similarity between these datasets, we believe it is viable to join these datasets together based on the vehicle class they predict, as well as other salient attributes like weather condition and time of day. However, the datasets will still be somewhat distinct due to the inherent differences influenced by distinct recording sites.

3 Methods

3.1 Feature engineering: from interpretable data to features

For signal processing software (down-sampling, Gaussian filters, normalization, Fourier/reciprocal transforms, wavelet decomposition, etc) we will use SciPy and Librosa. For machine learning models we will use PyTorch and Keras. We will use Pandas for data framing and matplotlib for visualization.

There were three phases in our feature extraction decisions. The first phase includes statistical features; for each audio clip, we averaged the mean, variance, kurtosis, and the IQR of pitches. A further improvement was made to include MFCC (Mel-frequency cepstral coefficients, part of a modern set of standardized audio analysis tools).

Then, the second phase included more spectral analysis, as we added the centroid, roll-off, bandwidth and flux. These are features that could be extracted through Librosa, and are very useful in many applications that involve sequential data analysis with the end goal of classification, or of speech recognition.

Lastly, we realized that many of these features can be extracted sequentially, meaning that we can split an audio clip into many frames, and extract the aforementioned data from each frame, to create a matrix of each column the specified analysis results, with the number of frames equal to the number of columns in our corresponding matrix.

The benefits of this would show up later in the LSTM training. Using convolutions (audio-wise) is one good example of finding patterns that correlate between these different statistical and spectral features, yet feeding sequential data into LSTM is a standard practice that we have, in this project, adapted using our frame decomposition method.

3.2 Deep learning: from features to label

We predicted vehicle class by feeding these features into a classification network. We explored the performance of multiple machine learning models including CNN's, LSTM and linear networks.

4 Model and Results

4.1 LSTM

LSTM is in one model we used. Audio data are essentially time series, and recurrent networks perform well on time series data by capturing both long-term and short-term patterns, so LSTM can work well on our project.

To identify whether the model for one dataframe can make a good prediction on another dataframe, we trained one model for the IDMT dataset, one for the MVD dataset, and one for the combined dataset (class “B” is dropped in the combined model, since it contained less than 0.5% (106/21735) of the total data).

Data Preprocessing included the following steps. First, handling none values in IDMT Dataset. Presently 8144 out of 17506 audios in IDMT dataset are audio for no vehicle, yet do not have label. Then, extracting features from audio: MFCCs, Chroma-stft, Root Mean Square. We used a OneHotEncoder, with categories ’C’, ’T’, ’M’, ’N’. Train-test-split. Lastly, we reshaped the dataframe to the shape required by LSTM.

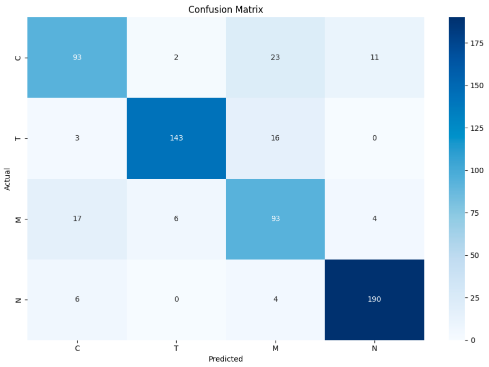
The initial Architecture we built includes two LSTM layers, each with 128 units, relu activation function, and 0.25 dropout rate, and two dense layers, one with 64 units and relu activation function, last one with dimension of output’s units and softmax activation function. We also used ‘adam’ optimizer and EarlyStopping with patience=20 to restore the best weights.

We improved our architecture by grid search function we defined ourselves and random search in Kerastuner. Random search is more efficient than grid search, which finds values that are very close to the best values in a much shorter time. Parameters and hyperparameters used include the number of neurons, batch_size, optimizer, activation function, dropout_rate, and the number of LSTM layers.

The last model for the combined dataset is three LSTM layers, each with 416 units, relu activation function, and 0.25 dropout rate, and two dense layers, one with 208 units and relu activation function, last one with the dimension of output’s units and softmax activation function, ‘adam’ optimizer and EarlyStopping.

4.2 Result for LSTM Models

As for accuracy, we have the four classes C, T, N, M as car, truck, none, motorcycle as such. For C: 93/129 correctly, 72%. For T: 143/162 correctly, 88%. For N: 190/200 correctly, 95%. For M: 93/120 correctly, 78%. Overall, the accuracy yields 85%.

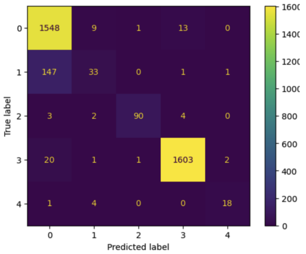


4.3 CNN + LSTM

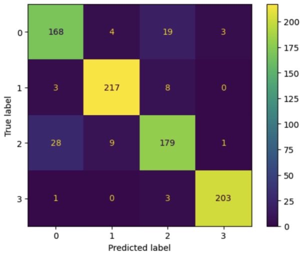
An augmented approach is based on convolutional neural networks. And since we wanted to identify patterns across every frame, using convolutions is a great idea, and we apply three layers of convolutions from 8 to 16 to 32, with Max pooling in between. Further, we fed the data into an LSTM model, which is based on convolutions for every frame.

Following an 80/20 split on the Indian dataset, we gathered a uniformly distributed set of training data, in terms of vehicle types. This is to ensure that false negatives are not applied onto categories with fewer frequency. Using cross entropy and ADAM with a learning rate of 0.0001, the size of these features extracted required one epoch to take 2 seconds. It was trained over 500 epochs to ensure loss stability.

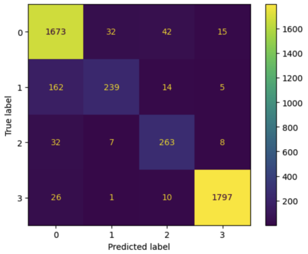
4.4 Results for CNN-LSTM Models



IDMT(before dropping ‘B’)



MVD



Joint

Accuracy Scores

	Tested on			
Trained on		IDMT	MVD	Joint
	IDMT	93.6%	70.0%	94.3%
	MVD	48.3%	89.8%	90.7%
	Joint	84.1%	84.1%	92.5%

5 Future Directions

Upon the positive results of the AutoAudio project, there are several approaches to increase the quality of the developed system and make it more widely applicable. One way to improve the robustness and generalizability of the models is to add more data sources, for example environment noise or to include different recording sites etc. Including various urban and rural settings in the datasets might reinforce the generalizability of the algorithm.

The first one can be addressed by a step of combining more complex ensemble models, and thus to assemble the best of different machine learning techniques which can ultimately yield higher classification accuracies. More advanced ensemble techniques like stacking or boosting could exploit these CNNs, LSTMs or similar model architectures more effectively to bring out some of their unique capabilities.

Another promising direction would be the prediction of other vehicle attributes, like velocity, orientation and acceleration. Such first-order features are important for better traffic dynamics understanding and enforce a more complete traffic management system as well.

Finally, metrics such as weather conditions, time of day, and amount of traffic may enable an inherent context for acoustic data interpretation, making the model better able to predict certain civilian activities.

6 Conclusion/Discussion

At the beginning of the semester, while our team was just forming we came together with an interest in utilizing audio data for our neural network. We all knew this wasn't something we had the most explicit experience in handling, but knew it was an important concept and a vital learning experience for all of us.

While we had attempted to find more audio data sets (even reaching out to the City of LA) , we were able to find two that met similar specifications. The IDMT (Germany) and MVD (India) both included short clips of various audio from multiple different types of vehicles in their areas. Once we had transformed all of our data we were able to all explore various avenues of neural networks. We focused much of our attention on Linear Networks, CNN's, and LSTM's (with the resulting values above). Each of these models posed different challenges in building and tuning.

Once we had trained the resulting models we were able to see how they were able to withstand various data sets and see the resulting predictive value. While they were not perfect in their predictive measure across the board, we were made aware of some key differences in our data set that proved invaluable to its predictive ability (sound quality, background noise, and distance from mic).

7 Individual Contributions

1. **Redford** developed the baseline code infrastructure and helper functions to manipulate the datasets, SQL database, and the training, grid-search and evaluation of PyTorch models. He also developed and obtained results for linear models. Finally, he worked with Lazar to refine the presentation.
2. **Jiaye** developed one of the three feature extraction functions, as well as the LSTM model. Jiaye also obtained results for the combination of these two components, which significantly bolstered the overall project.
3. **Lazar** helped to create some of the questions that were addressed throughout the project. This included addressing some questions we had surround location and motorcycles specifically. Additionally, Lazar helped with LSTM creation and debugging process. Throughout the process of coding we ran into some issues with certain indexing and iteration issues with our built functions. Finally, Lazar helped put together the structure of the presentation and the general design of the presentation.
4. **Yuqian** helped identify and think through the themes of our group's project and helped find and identify appropriate datasets. I was also responsible for the data cleanup of both datasets, including integration, feature engineering, and creating feature matrices to extract audio information. Finally, I completed the introduction and future directions.
5. **Xingjian** helped develop the convolutional approach upon reading a few spectral analysis standards (including Centroid, Roll-off, Bandwidth, Flux) in audio processing. Xingjian suggested breaking audio clips into around 180 frames—which effectively aligned with the sequential nature of our data, and wrote the CNN+LSTM architecture with these features.

References

[1] Fraunhofer. "IDMT-Traffic Dataset." https://www.idmt.fraunhofer.de/en/publications/datasets/traffic.html. Accessed 14 June 2024.

[2] "MVD Dataset." GitHub, https://github.com/Ashhad785/MVD. Accessed 14 June 2024.

[3] Alfonso, J., Naranjo, J. E., Menéndez, J. M., & Alonso, A. (2018, January). Vehicular communications. In Intelligent Vehicles (pp. 103-139). Butterworth-Heinemann.

[4] Hershey, S., Chaudhuri, S., Ellis, D. P., Gemmeke, J. F., Jansen, A., Moore, R. C., ... & Wilson, K. (2017). CNN architectures for large-scale audio classification. In 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 131-135). IEEE.

[5] Tran, V. T., & Tsai, W. H. (2021). Audio-vision emergency vehicle detection. IEEE sensors Journal, 21(24), 27905-27917.