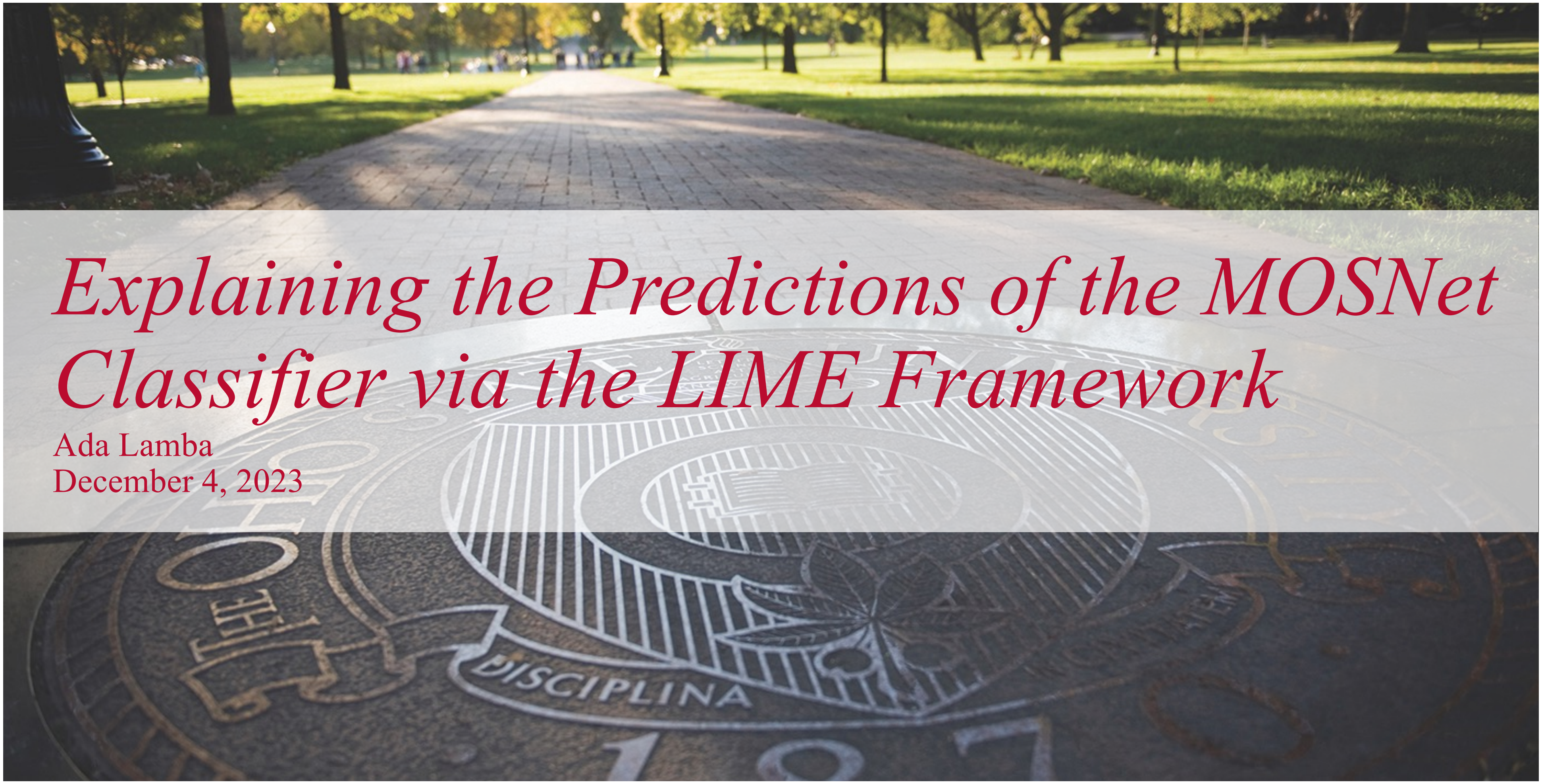


Explaining the Predictions of the MOSNet Classifier via the LIME Framework

Ada Lamba

December 4, 2023

THE OHIO STATE UNIVERSITY

Outline

Introduction and Background

Audio quality assessment, motivation, and recapping LIME, MOSNet

Method

Overview of implementation for running LIME framework on MOSNet

Results

Two examples of LIME output

Conclusion

Limitations and future work

Introduction and Background

Audio quality assessment, motivation, LIME, MOSNet

Speech Assessment

- Two metrics:
  1. *Quality* – how well the signal is perceived.
  2. *Intelligibility* – how well the speech in a signal is understood.

- Ways to measure quality or intelligibility
  - *Subjective* vs. *objective* measures: whether human listeners provide the score.
  - *Intrusive* vs. *non-intrusive* measures: whether a clean reference signal is present.

- *Mean Opinion Score (MOS)*: a subjective, non-intrusive measure of average human quality ratings on a scale of 1-5 .

Speech Assessment: Explainability

- Objective metrics are commonly used but correlate poorly with human perceptual scores – *why?*
- Hard to reason about the poor correlation if we don't understand what aspects of a signal the networks focus on.
- Networks are not black-boxes, but difficult to reason about on a large scale.

LIME [1]

- Recall from class discussion.
- Learns an already-interpretable model which locally approximates a target network on a specific input.

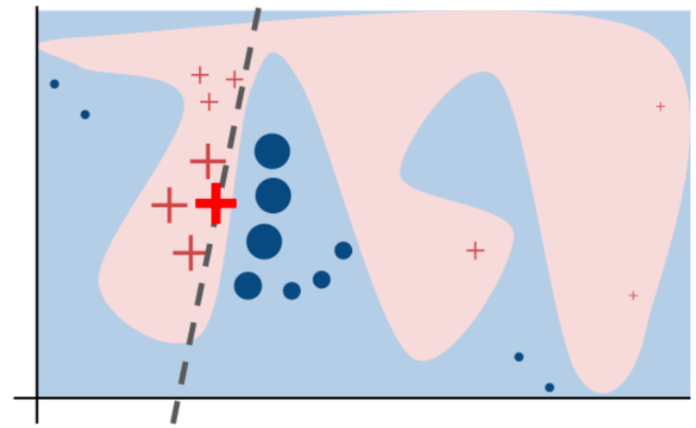


Figure 3: Toy example to present intuition for LIME. The black-box model's complex decision function f (unknown to LIME) is represented by the blue/pink background, which cannot be approximated well by a linear model. The bold red cross is the instance being explained. LIME samples instances, gets predictions using f, and weighs them by the proximity to the instance being explained (represented here by size). The dashed line is the learned explanation that is locally (but not globally) faithful.

MOSNet [2]

- Deep neural network which predicts a signal's MOS (audio quality) score from a magnitude spectrogram.

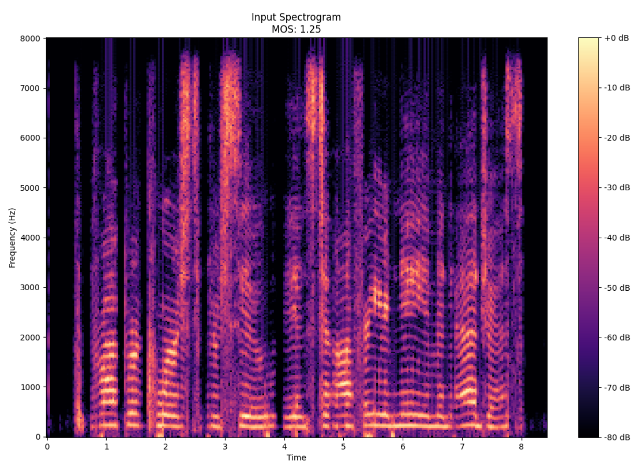


Table 2: *Configuration of different model architectures. The convolutional layer parameters are denoted as "conv{receptive field size}-{number of channels}/{stride}." The ReLU [23] activation function after each convolutional layer is not shown for brevity. N is for the number of frames.*

model	**BLSTM**	**CNN**	**CNN-BLSTM**
input layer	input (***N X 257 mag spectrogram***)		
conv. layer		$\left\{\begin{array}{l} conv3-(channels)/1 \\ conv3-(channels)/1 \\ conv3-(channels)/3 \end{array}\right\} X4$ $channels$ = [16, 32, 64, 128]	
recurrent layer	BLSTM-128		BLSTM-128
FC layer	FC-64, ReLU, dropout	FC-64, ReLU, dropout	FC-128, ReLU, dropout
	FC-1 (***frame-wise scores***)		
output layer	average pool (***utterance score***)		

Methods

Overview

Data Preprocessing

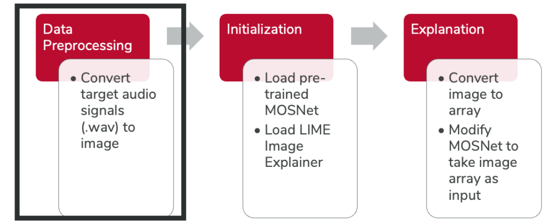
- Convert target audio signals (.wav) to image

Initialization

- Load pre-trained MOSNet
- Load LIME Image Explainer

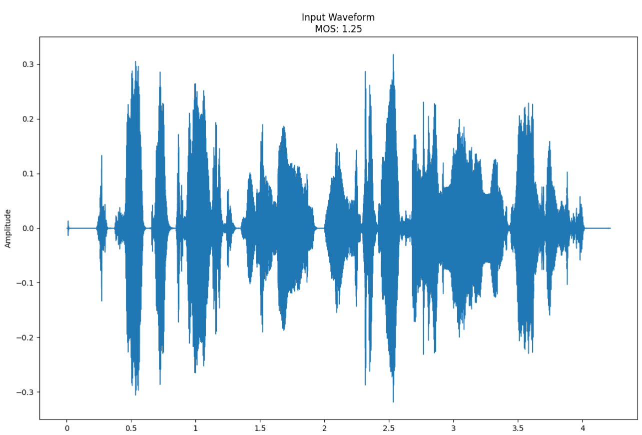
Explanation

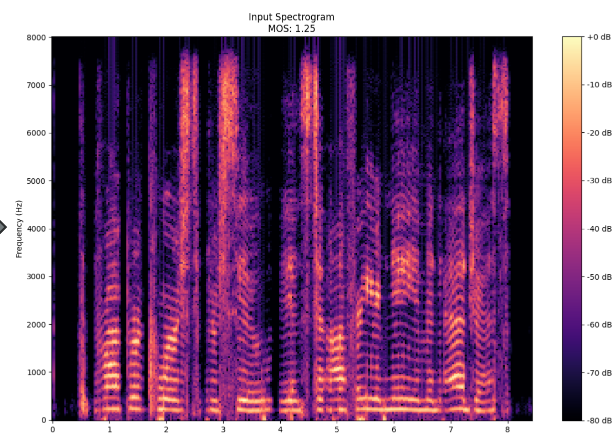
- Convert image to array
- Modify MOSNet to take image array as input



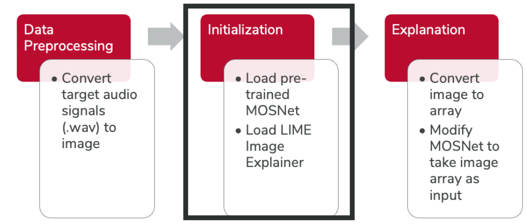
Data Preprocessing

- Dataset format: .wav
- MOSNet input format: magnitude spectrogram
- LIME input format: image array



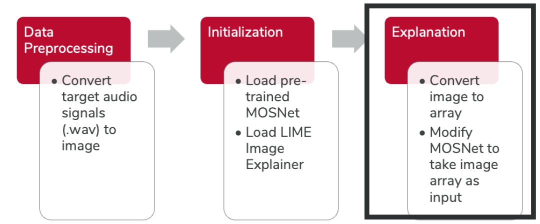


	0	1	2	3	4	5	6	7
0	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
1	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
2	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
3	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
4	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
5	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
6	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
7	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
8	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
9	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
10	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
11	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
12	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
13	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
14	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
15	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
16	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
17	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
18	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...
19	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...	[0.0, 0.0, 4...



Initialization

- Lime Image Explainer
  - Pros: Easy conversion from MOSNet input format
  - Cons: Requires a classifier, and MOSNet is a regressor
  - Example:
    - MOS: 1.25
    - Truncated: 1
    - Class probabilities [1, 0, 0, 0, 0]



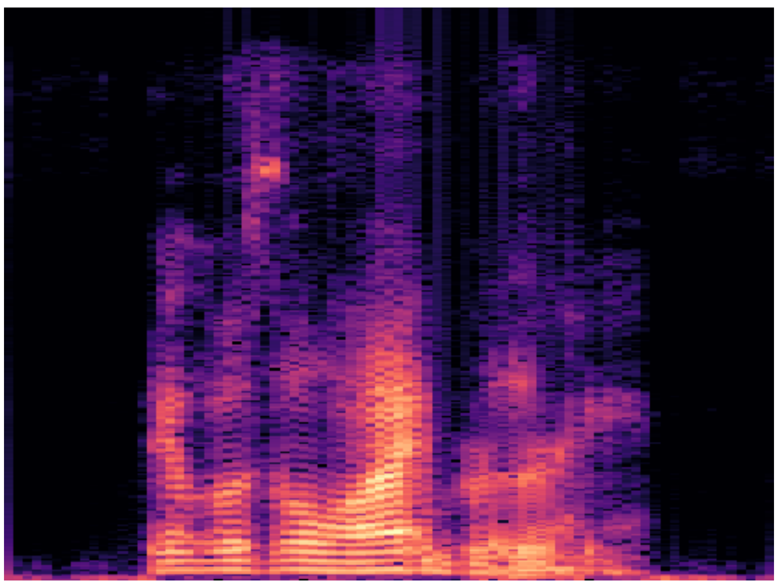
Explanation

- LIME calls the target model's prediction function
  - Issue: LIME has an image array, MOSNet takes magnitude spectrogram
  - Solution: Wrapper prediction function that loads wav file and calculates magnitude spectrogram for associated image
- Limitation: to reduce computation time, currently using 100 samples

Results

Two examples of LIME output

Image Explanation: Example 1



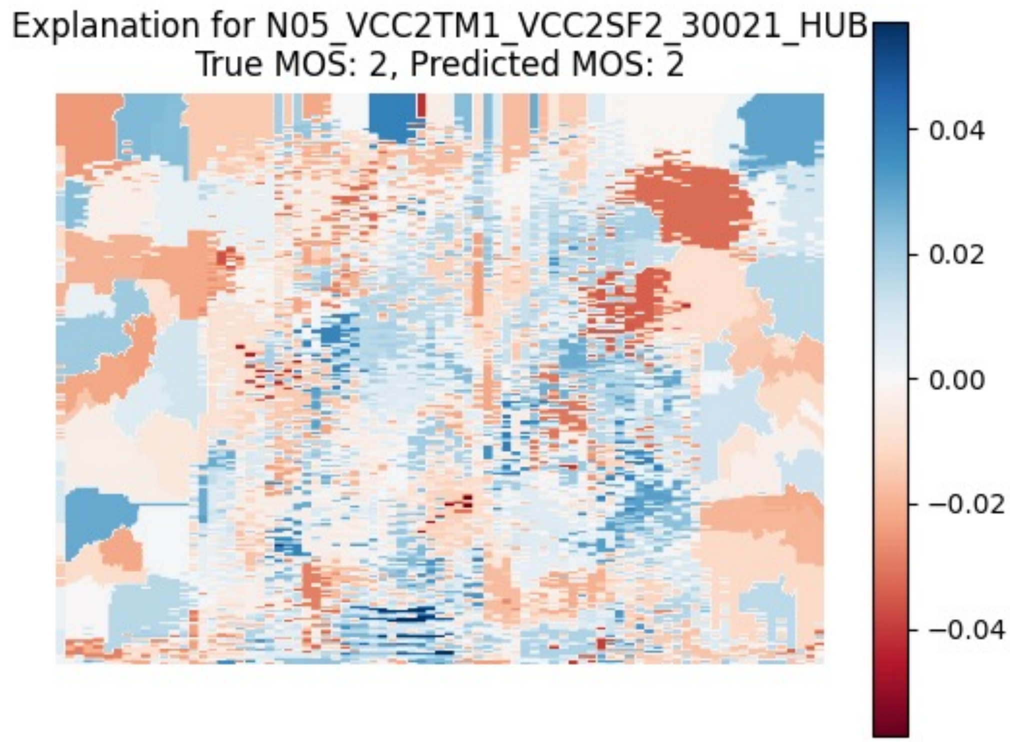
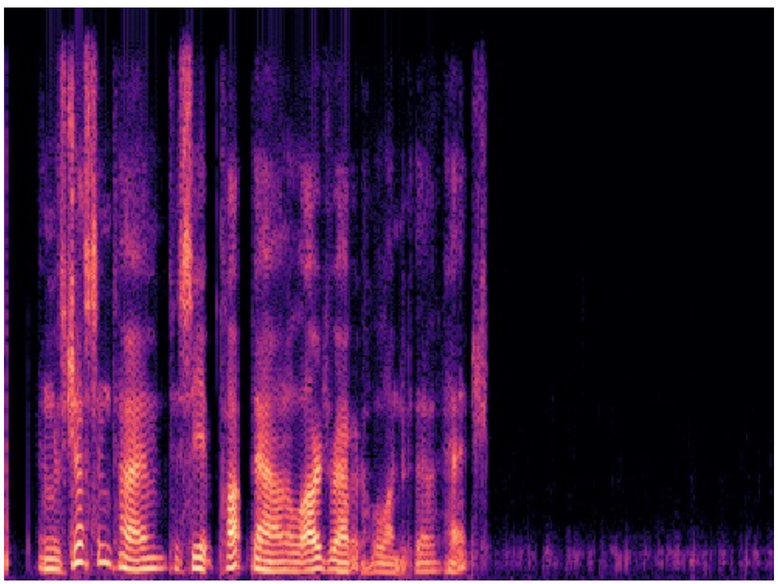
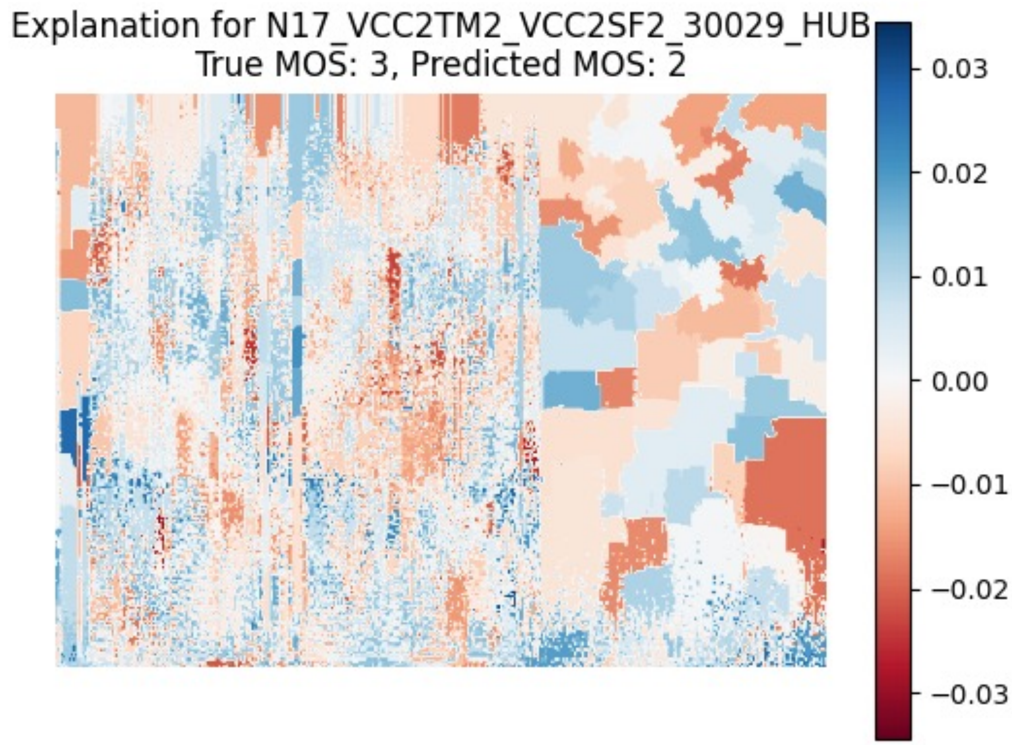


Image Explainer: Example 2





Conclusion

Challenges, Limitations, and Future Work

- Slight abuse of data format: converting audio file to image and using that.
  - Lossy?
- Audio signals are not natively supported by LIME.
  - Related work: audioLIME [3] is a preprint publication which implements LIME for audio signals.
- Future work:
  - Use a regression explanation model from LIME instead of categorization.
  - Use audioLIME implementation for MOSNet.

References

[1] M. T. Ribeiro, S. Singh, and C. Guestrin, "“Why Should I Trust You?” Explaining the Predictions of Any Classifier," in *KDD ’16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144 .

[2] C.-C. Lo, S.-W. Fu, W.-C. Huang, X. Wang, J. Yamagishi, Y. Tsao, and H.-M. Wang, “MOSNet: Deep learning based objective assessment for voice conversion,” in *Proc. Interspeech 2019*, 2019.

[3] Haunschmid, V., Manilow, E., and Widmer, G. “audioLIME: Listenable Explanations Using Source Separation.” 13th International Workshop on Machine Learning and Music, 2020