Audio Separation and Isolation: A Deep Neural Network Approach

Ahmed Hamdy, Pratap Kiran Vedula, Muni Venkata Jasantha Konduru
{ahamdy, pvedula, jkonduru}@stanford.edu

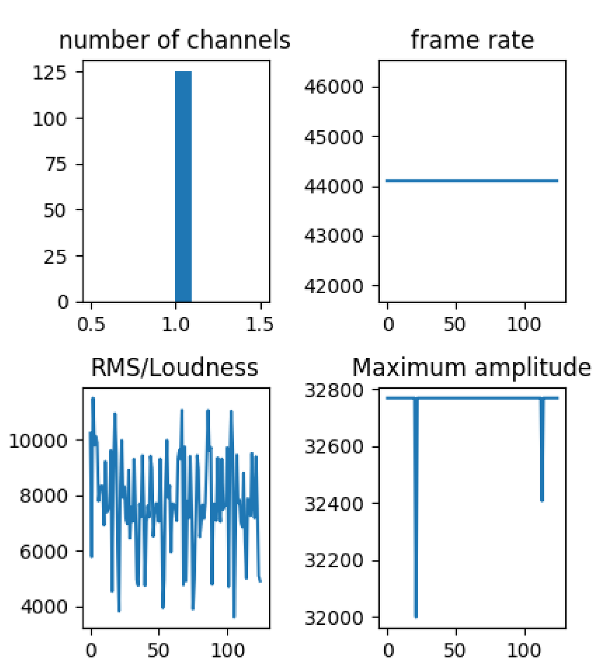
https://youtu.be/Gmc1WaOpw6U

INTRODUCTION

Sound event classification and isolation requires a trained system, when presented with an unknown sound, to correctly **identify** and **isolate** it. We propose a solution to what is commonly referred to as the "cocktail party problem". The **autoencoder CNN model** implemented would learn and **apply different filters to an input** in order to obtain a **set of audio sources from a mixed audio input**. Few example scenarios of what we have attempted to enable are self-driving cars identifying a police siren, isolating a broadcaster's voice from others in a loud crowd, recognizing an infant crying in a noisy environment, or all at once!

DATA

Individual audio input were obtained from MSSC 2018 [1], SWC [2], as well as several Kaggle [3] and GitHub [4][5] submissions pertaining to the model classes (1) **Baby Cry**, (2) **Dog Bark**, (3) **Emergency Siren**, (4) **Human Speech**, and (5) **Others**.



To ensure that model will be trained and evaluated on a robust data set with minimal duplications, each class audio, except Speech, was **augmented** by **randomly shifting pitch** by 2 or 4 steps, **stretching** by 1.2 times, and **shifting loudness**. We this we then leveraged **93 percent of the data for the training set** and remainder **7 percent as dev set**.

Number of Audio Files per Class that were used for dataset generation

	Crying Baby	Siren	Dog	Speaker	Other
Reduced MSoS + SWC	31	16	16	2540	300
MSoS + SWC + Baby + Siren + Dog + Augmented (Baby, Siren, Dog)	111	336	440	2540	1442

Percentage of Duplicate across 100000 Dataset per Class

	Crying Baby	Siren	Dog	Speaker	Other
Reduced MSoS + SWC	1.935483871	3.75	3.75	0.023622047	0.2
MSoS + SWC + Baby + Siren + Dog + Augmented (Baby, Siren, Dog)	0.540540541	0.178571429	0.136363636	0.023622047	0.041608877

REPRESENTATION

Prepossessing is performed by converting input and output audio files to a **sample rate of 22050 Hz**, reducing the duration of audio files to **3 seconds**, **normalizing using min-max** parameters, and obtaining an **STFT representation** of the audio input and target outputs with a **window size of 23 ms**, and a **hop length that's fourth the size of window** used.

MODEL

Vertical convolution layer allows us to obtain the **frequency feature** while the horizontal convolution layer provides us with the **time dependent feature** and thus outputting a **time-frequency encoding**.

The model attempts to learn a set of **five different filters that are applied to the input** to obtain **five outputs representing the isolated sources**.

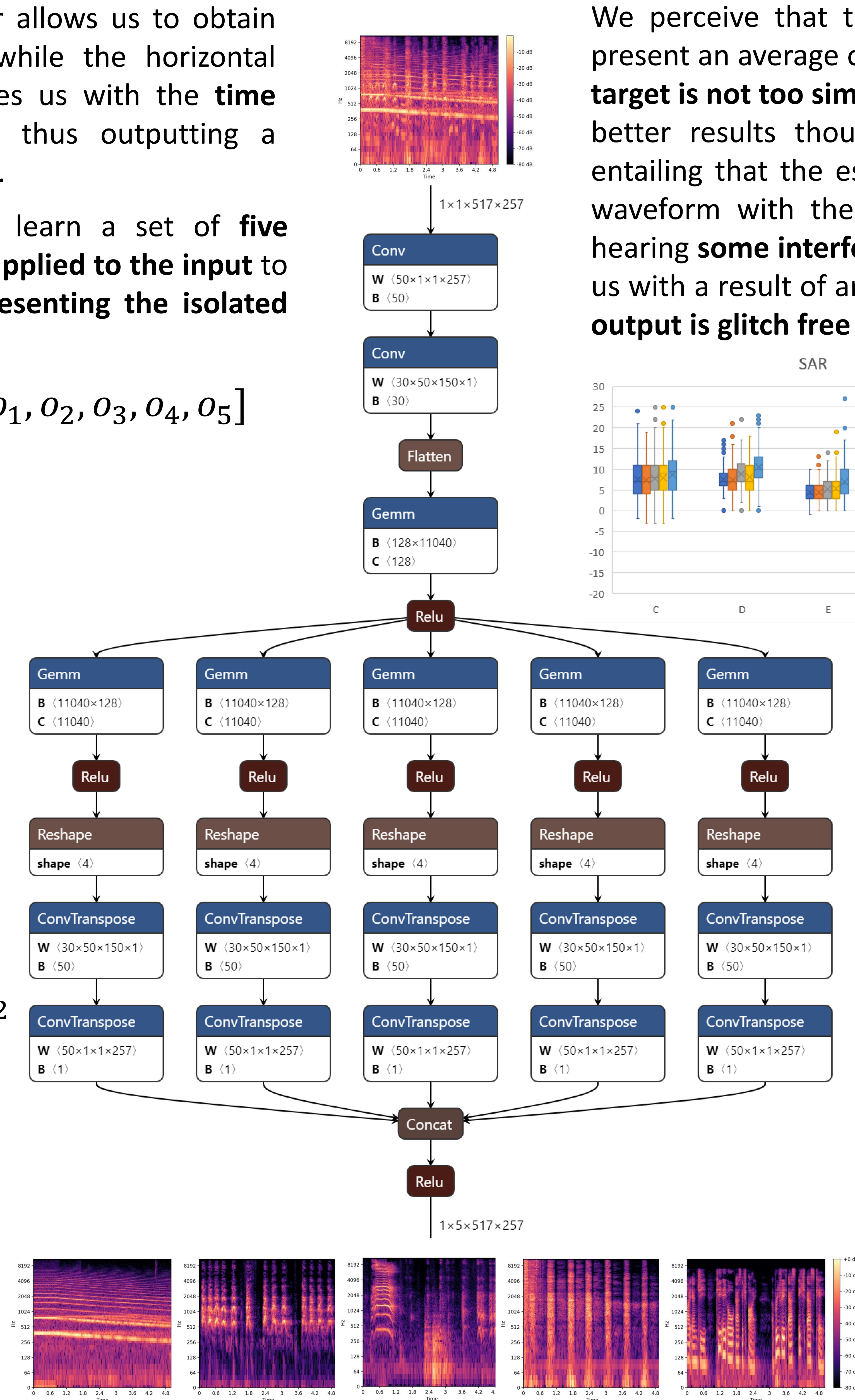
$$concat\_output = [o_1, o_2, o_3, o_4, o_5]$$

$$f_i = \frac{o_i}{\sum_{n=1}^{5} o_n}$$

$$\tilde{y}_i = f_i * x$$

Losses are computed per class by measuring the **Mean Squared Error (MSE)** between the calculated and the provided target output.

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$$



RESULTS

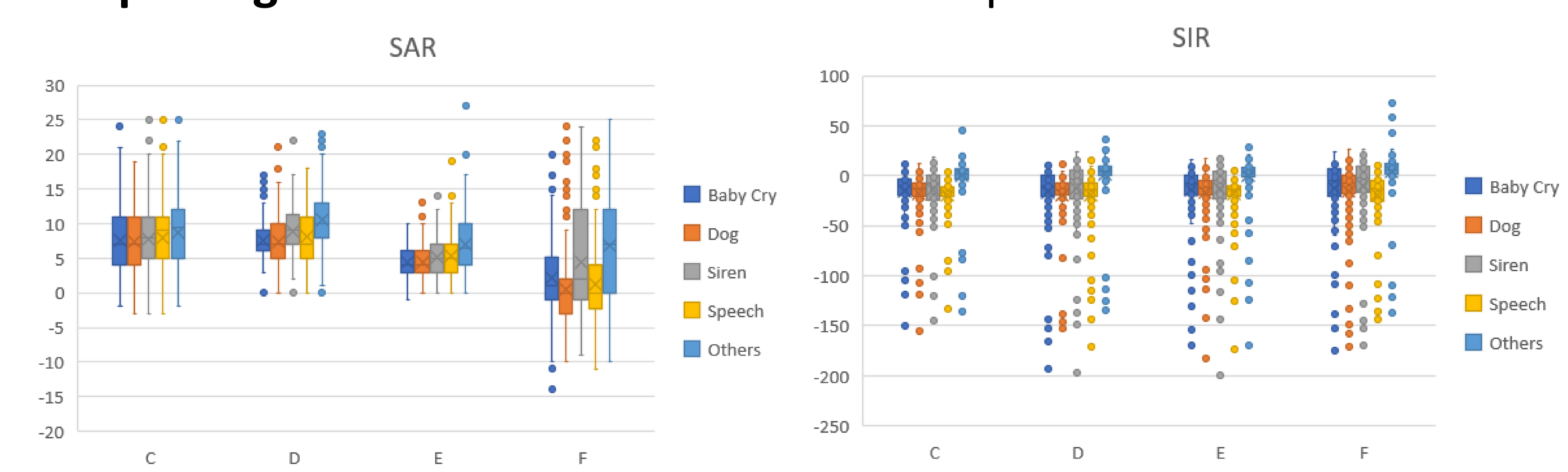
Summary of the results corresponding to a subset of experiments evaluated is presented, implying that **the model is overfitting** as we see a **variance issue** and **necessitating a bigger dataset to train** on as a possibility to remedy this issue.

	Experiments	Train Avg. Loss	Dev. Avg. Loss
	Base Model	> 100000	-
A	Exchanged Zero Padding + Convolution Layers with De-convolution Layers	40000	-
B	Training on Filters of the Model Output	1700	-
C	Using ADADELTA as Loss Optimizer and apply ReLU to the concatenated model output	240	272
D	Exchanged MIN MAX with CMVN Normalization	219	204
E	Increased size of FC Features to 376	237	225
F	Exchanged CMVN with MIN MAX and ADADELTA with ADAM at LR 0.001 as well as decreased size of FC Features to 128	66	129

SDR: *Signal to Distortion Ratio*
SIR: *Source to Interference Ratio*
SAR: *Sources to Artifacts Ratio*

DISCUSSION

We perceive that the **SDR** for all of the models experimented present an average of around **-150**, entailing that the **ground truth target is not too similar to the estimated output**. The **SIR** presents better results though on average appears to be around **-10**, entailing that the estimated output shares some portions of the waveform with the other estimated outputs, as in we will be hearing **some interference amongst the classes**. The **SAR** provides us with a result of around **8** on average, meaning that the **isolated output is glitch free** with **minimal artifacts** present.



FUTURE WORK

- Obtain a **Mel Spectrogram representation** of the input transforming the sounds onto the **Mel Scale**
- Training the model on a **larger data set** to remedy the variance issue observed
- Explore the use of **LSTM** in order to train the network on shorter waveforms for near real-time inference
- Accounting for **discriminatory loss terms per class** to further decrease SDR and SIR

REFERENCES

[1] Harris, Lara; Bones, Oliver Charles (2018): Making Sense Of Sounds: Data for the machine learning challenge 2018. figshare. Dataset. "https://doi.org/10.17866/rd.salford.6901475.v4"
[2] Köhn, A., Stegen, F., & Baumann, T. (2016). Mining the spoken wikipedia for speech data and beyond.
[3] Moreaux, M. (2017, October). Audio Cats and Dogs, Version 5. https://www.kaggle.com/mmoreaux/audio-cats-and-dogs
[4] donateacry-corpus, (2015), GitHub repository, https://github.com/gveres/donateacry-corpus
[5] Siren-Identification-Localization, (2016), GitHub repository, https://github.com/Siren-Identification-Localization/Siren-Identification-Localization/tree/master/datasets