CNN-LSTM FOR NONINVASIVE NEONATAL OPIOD WITHDRAWAL SYNDROME (NOWS) DIAGNOSIS THROUGH INFANT CRY

By

James Lu

B.Sc., Brown University, 2022

Thesis

Submitted in partial fulfillment of the requirements for the

Degree of Master of Science in the School of Engineering at Brown University

PROVIDENCE, RHODE ISLAND

MAY 2022

Abstract of "CNN-LSTM For Noninvasive Neonatal Opioid Withdrawal Syndrome (NOWS) Diagnosis Through Infant Cry", by James Lu, Degree ScM., Brown University, May 2022.

Deep learning shifts the way to build signal processing systems from coding or model-centric to data-centric. This paper presents a system to support data-centric deep learning for signal processing. Using new data from an ongoing medical case study, the work sets the direction for objective assessment for diagnosing neonatal opioid withdrawal syndrome (NOWS) through infant cry.

Our approach to the NOWS classification decision combines two deep learning models, a long short-term memory recurrent neural network (LSTM-RNN) and a convolutional neural network (CNN), so that early decisions are not made *a priori*. One issue for this work was obtaining sufficient data, and we are sure we did not have enough. Also, the baseline true decisions may also be questionable. Nevertheless, with the data thus obtained, we were able to achieve nearly a 90% correct classification of verification data. Realistically, however, we are virtually certain that more data will lead to different performance levels and factual assessment of the important parameters of the input data and classifier. As clinical data becomes available over time, more work can be used with this classifier to improve its performance. In addition to the difficulty of training the Acoustic Neural Network (ANN), the work addresses issues in machine learning lifecycle such as the data pipeline, the testing benchmark, performance metrics, and deployment plan.

AUTHORIZATION TO LEND AND REPRODUCE THE THESIS

As The sole author of this thesis, I authorize Brown University to lend it to Other institutions or individuals for the purpose of scholarly research.

Date 04/25/2022 *James Lu*

James Lu, Author

I further authorize Brown University to reproduce this thesis by photocopying or other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

Date 04/25/2022 *James Lu*

James Lu, Author

This thesis by James Lu is accepted in its present form

by the School of Engineering as Satisfying the

thesis requirements for the degree of Master of Science

Date _____ _____

Harvey F. Silverman, Ph.D., Advisor

Approved by the Graduate Council

Date _____ _____

Andrew G. Campbell, Dean of the Graduate School

Vitae

James Lu is a student at Brown University, expecting ScB and ScM in Electrical Engineering in May 2022. In summer 2019, DOE selected James nationwide to Robotics Summer Internship at Idaho National Laboratory. He worked as Design Project Teaching Assistant for Brown's ENGN 0040 Dynamics and Vibrations in spring 2020. He has been Research Assistant at Brown's Laboratory for Engineering Man/Machine Systems (LEMS) since March 2020. In summer 2021, he was chosen to NSF REU Site at University of Massachusetts Dartmouth: Secure, Robust, and Resilient AI-enabled System Engineering. Currently, he works as Brown MATLAB Student Ambassador and serves for E-Board of Brown Undergraduate Research Club. He is an IEEE Student Member since 2015. Lu's research interests include signal processing and machine learning for systems and robotics. He won Journal of Young Investigators (JYI) December 2020 Featured Research and IEEE International Symposium on Technologies for Homeland Security (HST) 2021 Cyber Security Best Paper.

Acknowledgments

Forever am I grateful for the many blessings that have carried me to this life milestone. First, I am deeply indebted to my thesis advisor, Prof. Harvey F. Silverman. First meeting him as a freshman advisee in 2018, Prof. Silverman has continued to provide invaluable guidance and support in my academic journey. Under his supervision, I learned about the research process, designing experiments and navigating challenges. The lessons learned from his mentorship would carry me far and help me succeed in future research experiences. He emphasized the importance of conducting "good science". A researcher must constantly question their data and results, thoroughly understanding implications possible before drawing conclusions. Prof. Silverman's mentorship has been a key factor in my growth as a student and success as an engineer.

Besides my advisor, I would like to thank Dr. Barry Lester and Dr. Stephen Sheinkopf for their continuous support and guidance on this thesis. Without their expert domain knowledge, I would have never been able to reach profound conclusions. I would also like to thank all the Brown University Engineering staff and faculty for supporting my education and paving my path to success. This thesis would not have been possible without their contribution.

Last but not least, I would like to express my deepest gratitude to my family and friends. This thesis would not have been possible without their warm love, trust, motivation, continued patience, and endless support.

Table of Contents

List of Tables

List of Illustrations

1. Introduction

1.1. Background

A newborn's cry is one of the most elementary yet universal forms of communication that humans experience. It is both a primitive acoustic vocalization and complex information channel. The physical action constitutes a complex amalgamation of pharyngeal, laryngeal, and thoracic motion to form a language (Truby and Lind). Cries are deceptively informative, containing layers of emotional and physical implications. However, they suffer from subjective interpretation due to linguistic ambiguity. Nonetheless, infant cries likely hold critical information with significant biomedical potential.

Infant-cry analysis is a growing field in biomedical signal processing. Among several studies, infant cry analysis spans three main directions: cry detection, cry diagnostics, and cry interpretation. *Cry detection* involves analyzing audio to determine whether a baby is crying among other sources of noise. Systems have been designed to detect baby cries in home and car settings to alert caretakers of danger (Cohen and Lavner), (Foo, Yap and Hum). *Cry diagnostics* focuses on detecting acoustic differences in infants experiencing various conditions. These may include deafness or respiratory ailments (Garcia and Reyes Garcia), (Saraswathy, Hariharan and Yaacob). *Cry interpretation* aims to "translate" infant utterances into a comprehensive language. Research efforts in this direction have categorized certain sounds to different infant demands such as hunger or sleepiness (Mima and Arakawa).

Among the three directions, cry diagnosis holds vast potential. It is widely applicable to many conditions where infant cry is a notable biomarker. Cry diagnosis is also seeded

in ground truth by expert domain knowledge depending on the condition. Neonatal abstinence syndrome (NAS), also known as Neonatal opioid withdrawal syndrome (NOWS), is a prominent condition that may be able to be distinguished from a newborn's cry (Devlin, Breeze and Terrin), (Chin Foo, Dansereau and Hawes). Newborns whose mothers may have suffered from Opioid Use Disorder (OUD) can experience mild to severe withdrawal symptoms. Symptoms may vary but typically involve tremors and hyperirritability depending on type and time of exposure among other factors. With a rise in newborns suffering from NAS, it becomes more crucial for standardized and objective treatment (University of North Carolina School of Social Work).

Early NAS detection is difficult because a truly objective diagnosis test currently does not exist. The Finnegan Neonatal Abstinence Scoring Tool (FNAST) is used to predict if, and to what degree, a newborn suffers from NAS. It consists of 21 *observer-rated subjective* assessments with multiple subcategories that quantify NAS severity (Devlin, Breeze and Terrin). Moreover, the FNAST takes an assessor about 40 minutes, so it is expensive. Each assessment is scored between 1 to 5 and these assessments range from central nervous system behaviors to gastrointestinal disturbances. There are also subjective assessments taken from cries of the infants. Depending on the score, practitioners may follow various methods of pharmacologic therapy (University of North Carolina School of Social Work). Table 1 below depicts the standard FNAST assessments that nurses will use to measure NAS severity. Some of the 21 items listed have different severities associated with different scoring contributions.

Table 1: 21 Item Finnegan Test

Item	**Severity**	**Score Max**
High-pitched crying	Excessive	2
	Continuous	3

Sleeps after feeding, h	<3	1
	<2	2
	<1	3
Moro reflex	Hyperactive	2
	Markedly hyperactive	3
Tremors when disturbed	Mild	1
	Moderate to severe	2
Tremors when undisturbed	Mild	3
	Moderate to severe	4
Increased muscle tone		2
Excoriation		1
Myoclonic jerks		3
Generalized convulsions		5
Sweating		1
Body temperature	37.2-38.3	1
	≥38.4	2
Yawning >3 times/scoring interval		1
Mottling		1
Nasal stuffiness		1
Sneezing >3 times/scoring interval		1
Nasal flaring		2
Respiratory rate	>60/min	1
	>60/min with retractions	2
Excessive sucking		1
Poor feeding		2
Regurgitation		2
Projectile vomiting		3
Stools	Loose	2
	Watery	3

Based on the Finnegan test score, a number of treatments may be administered. Initial treatment may include supportive care such as low-stimulation, increased feeding, and other measures to reduce infant stress. However, more severe symptoms call for pharmacologic therapy, and it is found that 50-75% of substance-exposed infants will require this treatment. Common medications include small doses of morphine sulfate and methadone. After frequent monitoring, the patient is discharged after no longer showing

signs of NAS (University of North Carolina School of Social Work). Figure 1 below depicts a standard flowchart clinicians may use when treating NAS.

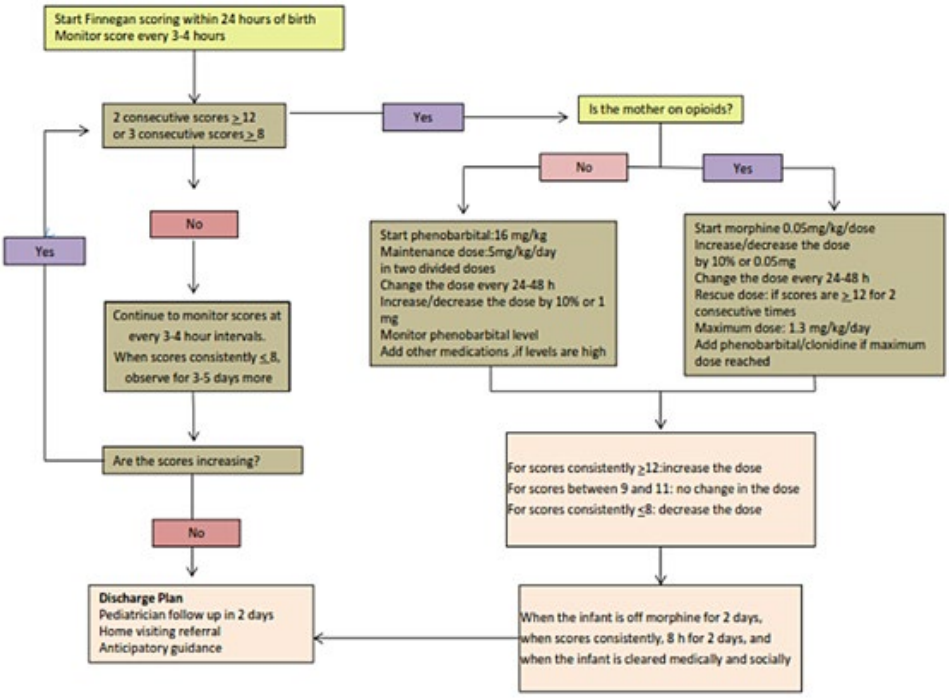


Figure 1: Finnegan Test Treatment Methods (University of North Carolina School of Social Work)

A significant issue with the Finnegan test is that it suffers from scoring subjectivity and high variability between different assessors. This is largely from being a human-administered assessment that is subject to different conditions. Several changes have been proposed to the current Finnegan test that involve editing assessment categories. A recent study compared percent endorsement of FNAST categories among three (Louisville, Tufts, and University of Kentucky) different geographical regions that totaled 424 infants. The three subjective measurements that exhibited the largest discrepancies were convulsions, high-pitched cry, and hyperactive Moro reflex. In particular, high-pitched cry received high marks from the Louisville [77.2%] and Kentucky [79.8%] groups but low marks from the Tufts group [20.7%] (Devlin, Breeze and Terrin). This suggests the high-pitched cry

assessment is likely to be very subjective, depending on nurse judgement. What sounds "high-pitched" might be different for a nurse with less experience treating NAS than one with more.

Another study compared FNAST performance to another NAS assessment called the NICU Network Neurobehavioral Scale (NNNS) (Chin Foo, Dansereau and Hawes). NNNS analyzes an infant's neurologic development and signs of stress or withdrawal. Unlike FNAST, it is applicable to both healthy and substance-exposed patients. The study compared FNAST and NNNS results on the same 78 infants and found several categories to be closely correlated, which included FNAST's cry category and NNNS's high-pitched cry and predominant state categories. This suggests that infant cry may be a valid category when screening for NAS but suffers from subjectivity between nurses. This thesis aims to establish an objective and noninvasive method to assess infant cry for NAS using Machine Learning (ML) with Signal Processing (SP).

1.2. List of Past Work

As a subfield of artificial intelligence (AI), machine-learning applications are increasingly more popular in signal processing analysis. Today, with significant data for training, neural-network technology is being adopted for most of the AI solutions in use. A large segment of the AI work is now used for pattern recognition and the classification of data formulated into images, vector functions of time or combinations of both. Recurrent Neural Networks (RNNs) are feedforward neural that are popular for classifying sequential data such as speech where temporal information and context is important (Sutskever). More sophisticated RNN architectures have also been proposed negating the need for pre-

segmented data or output post-processing (Graves, Fernandez and Gomez). As training methods improve, neural networks approach human-level performance as observed in studies in object recognition (Geirhos, Janssen and Schutt). There has also been great success in using RNNs for acoustic modeling, sound event detection, speech recognition and speaker diarization (Graves, Mohamed and Hinton), (Hinton, Deng and Yu), (Nguyen, Nguyen and Phan), (Zhang, Wang and Zhu).

However, RNNs are often difficult to train. They may suffer from the exploding or vanishing gradient phenomenon resulting in loss of long-term correlations (Pascanu, Mikolov and Bengio). RNNs are also susceptible to overfitting and can fail to learn over a generalized range of domains (Bronstein, Bruna and LeCun). Increasing model complexity or data size may also decrease performance as observed in the double-descent phenomenon (Nakkiran, Kaplun and Bansal). Like other machine learning networks, training is data-centric, heavily dependent on quality and quantity of data used. Often, the most labor comes from data preprocessing to ensure good RNN performance and the removal of unintended correlations (Press). Ethics and end users' trust become important considerations if the data is security-sensitive (Paleyes, Urma and Lawrence). For example, hospital data may contain private patient information. There are also several adjustable hyper-parameters impacting network performance.

One type of RNN commonly used to analyze audio data is the Long Short-Term Memory (LSTM) RNN. LSTMs can learn long-term dependencies and avoid exploding or vanishing gradients by having error backpropogated using linear memory cells (Hochreiter and Schmidhuber). With memory blocks and peephole connections, it can identify correlations despite temporal distance. Figure 2 from Christopher Olah's post illustrates

the four layers making up an LSTM unit where each layer is denoted as a column in the flowchart (Olah).

In the figure, the memory of the LSTM, c, is modeled by a group of N memory cells, N a user parameter, that will be updated as one progresses through the discrete-time intervals. The input, x, is made up from vectors of length M that are a function of discrete time, and the output is indicated by vectors of length k that are functions of discrete time as well. The first layer uses input data x_t and former output data h_{t-1} decide what information the cell state, c_t , will "forget" where the cell state is indicated at the top horizontal row of the figure. This is done by multiplying the state by the output of a sigmoid (σ) function whose output is between zero and one. The next sigmoid and tanh update the current state with new information by addition. The last sigmoid takes data from the input and former output and uses it to modulate truncated state data to produce the next output, h_t.

BiLSTMs are a common variation that utilize both forward and back propagation through the LSTM layers for both training and testing. As we expect baby-cry context in both directions to be important, the forward-backward algorithms were used here throughout our research.

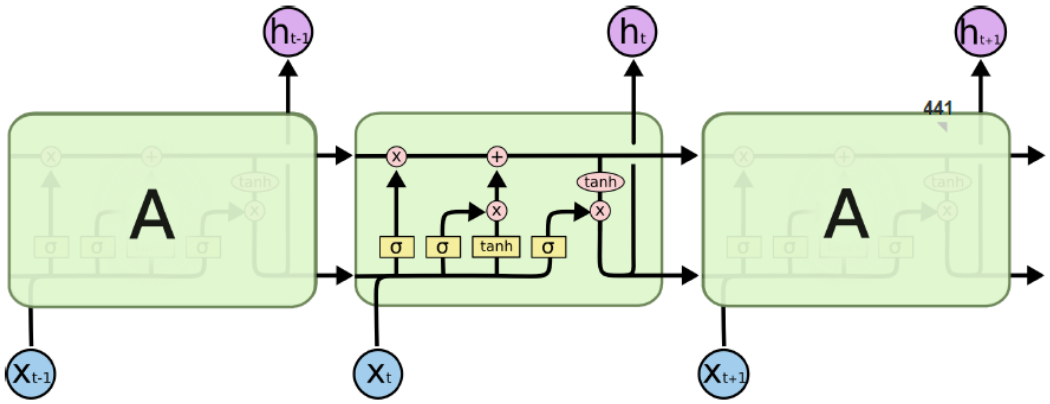


Figure 2: LSTM Architecture (Olah)

Long-term dependency capabilities make LSTMs excellent for analyzing speech. BiLSTMs have been used in voice conversion where acoustic context is important to maintain. For example, effective text-to-speech must consider long-term intonation and sentence context. In one case, important speech parameters in the source signal were extracted, processed through a BiLSTM, and resynthesized into converted speech. By utilizing the BiLSTM, the converted speech maintained quality and continuity (Sun, Kang and Li). Deep LSTMs are very effective in end-to-end speech recognition where phoneme sequences are context dependent. The Deep BiLSTM RNN broke performance records on phoneme recognition for the TIMIT phoneme recognition benchmark spanning 61 labels (Hinton, Deng and Yu).

Convolutional neural networks (CNNs) offer an unsupervised method of image feature extraction without domain knowledge. In each layer, a convolutional kernel moves over the input mapping to a reduced output, called a “feature map”. These networks send data through several convolutional layers filter and reduce the input towards its key features (MathWorks). These layers are enhanced by Rectified linear unit (ReLU) and Pooling layers that normalize and down-sample the feature map as shown in Figure 3 (MathWorks).

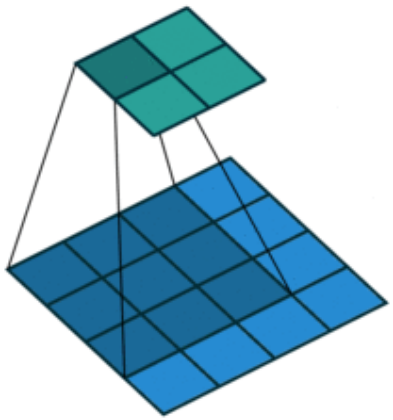


Figure 3: Convolutional kernel tiling over input space (MathWorks)

CNNs are becoming increasing popular in audio spectrogram feature extraction for classification. One study classified identified audio sources in a noisy environment by sending the audio spectrogram through a CNN. The model effectively identified sirens, car engines, dog barks, and much more at high accuracy (Piczak). Another study followed a similar approach but classified YouTube videos by genre based on the video's audio spectrogram (Heershey, Chaudhuri and Ellis).

1.3. NAS cry identification with Machine Learning

The three baby cry analysis directions all share a workflow similar to the first three stages of the Ashmore Machine Learning deployment workflow: data management, model learning, and model verification. Data management involves collecting, preprocessing, augmenting and analyzing data to be used for training and validation. Model learning refers to selecting, training, and tuning a type of machine learning network. Lastly, verification includes tests and other performance analytics (Ashmore, Calinescu and Paterson).

Two prominent issues machine learning for baby cries encounter are parameterization during data management and categorization during model learning. Based on an application, different acoustic parameters may be more relevant than others. Several baby cry detection systems have used pitch-related parameters such as fundamental frequency and vocal-tract related parameters such as formants. Mel-Frequency Cepstral Coefficients (MFCCs), the most widely used set of parameters for speech recognition, are also frequently used across many applications such as differentiating cries between normal and deaf infants (Garcia and Reyes Garcia). The MFCCs are a power spectrum representation on the mel scale that are taken to the cepstral domain. The mel scale adjusts

for hearing importance of frequency bands typically for adult speech production (Saraswathy, Hariharan and Yaacob). Some studies have also brute-forced parameters to find ones most effective between cry categories (Tuduce, Cucu and Burileanu). Reggiannini et al. puts together a standardized set of parameters for baby cry analysis. With vital parameters identified, studies have explored several probabilistic classification algorithms, from K-Nearest Neighbors (KNN) to Random Forest (Bano and RaviKumar). Machine learning has yet to be thoroughly utilized in baby cry categorization, although there are indeed some papers in the literature.

This thesis proposes a novel approach to neonatal abstinence syndrome diagnosis utilizing signal processing and machine learning. It will introduce a more objective and non-invasive means for analyzing if a newborn's cry is symptomatic or healthy. Given the data, the thesis will investigate which acoustic parameters effectively correlate between these classes to determine if cry is a reliable biomarker for the Finnegan Neonatal Abstinence Scoring Tool (FNAST).

2. Pre-Processing Procedure and Programs

2.1. Early and Final Goals for the Proposed Research

The goal is to create an executable computer program that clinicians may use to gain an objective probability that an infant's cry shows signs of NAS. The program should run in near-real time so it can be used conveniently in a hospital. Moreover, it should easily be run from a laptop or similar hardware and perform well in a noisy hospital environment, objectively measuring the FNAST high-pitched cry category.

As Artificial Neural Network (ANN) techniques are to be used for the system, a major component will necessarily be the gaining of a suitable and sufficiently large and diverse set of training data as an essential first milestone for this research. Once the audio for the cry dataset has been obtained there is a substantive amount of work that needs to be done curating and preprocessing infant cry data. Data management is crucial to ensure the ANN model is not finding correlations other than ones tied to NAS. There are several methods used to homogenize each sample of data before constructing a data base. It is also important that each subject is equally represented in the data base. Another key milestone is to discover the right set of processed features to use as input data to the ANN system.

2.2. Step 1: Data Collection

Healthy and symptomatic infant cry audio data specifically for the purpose of investigating the NAS syndrome were collected over the course of nearly 3 years by nurses at Rhode Island Women & Infant's hospital. Single cry episodes were typically induced by the nurses or, occurring naturally occasionally, and recorded using a custom-built audio recorder sampled at 24,000 Hz, with PCM data at 16 bits per sample. Other cries were

captured by the same hardware devices but recording over long periods of time (up to 24 hours of continuous recording), attempting to capture more cries of the natural type. The data was manually labeled as either having the syndrome or not by clinicians using FNAST scoring and expert domain knowledge as ground truth. As FNAST is far from a pure-truth assessment, clearly this ground truth is not ideal, but it is all we have right now for early assessment.

2.3. Step 2: Preprocessing

Data quality and quantity is highly variant over different recording conditions. Many of the captured cries have interfering noise, mostly from the nurses sweetly talking to the infant as she made an assessment. Then, the long recordings have to undergo a preprocessing sequence to make the data useful for the ANN classifier.

All raw audio data is processed through the "stripper" program. All instances of acoustic activity are written into a new .wav file with two seconds (48,000 samples) of silence spaced in between. The output includes everything from cries to noise. The process extracts instances automatically using signal processing techniques and required no user involvement. As shown in Table 2, file information is stored in the 48,000 zeros representing the two seconds of silence at the beginning of each stripped audio episode starting at index 1,000. After processing selected files, the "stripper" concatenates all the resulting sections and writes the reduced data into a new .wav file, named *header*_100. The output of the "stripper" reduces, for example, a 12-hour recording to about 40 minutes with about 40-80 loud passages, each separated for later processing.

Table 2: Information stored in preceding zero section before each file's data

Z(1000)	Day of recording (2 digits)

Z(1001)	Month of recording (2 digits)
Z(1002)	Year of recording (4 digits)
Z(1003)	Hour of recording in military time (2 digits)
Z(1004)	Minute of recording (2 digits)
Z(1005)	Seconds of recording (2 digits)
Z(1006)	Floor(length of episode/32768)
Z(1007)	Mod(length of episode/32768)
Z(1008)	Infant classification (1 char)

Note: stored length is just the episode length, does not include length of zeroes section

The stripped output is then further processed in the "observer" program. It plays each sound utterance of a stripped file and allows the user to keep, but mark as corrupted, edit, or discard the utterance. Corrupted denotes a recorded cry that also has significant noise. Given the data-centric nature of machine learning, it became crucial to meter data quality before training (Tuduce, Rusu and Cucu). Corrupted cries are later excluded from the data set to eliminate one incontrollable noise variable. The editing feature uses a MATLAB GUI, its screenshot shown in Figure 4, that allows the user to trim the utterance between two selected boundaries. The output of an observed file are good quality infant cry episodes with two seconds of silence spaced in between to be used for training. This typically left 15-20 minutes of audio with 20-30 saved episodes. This was the most time-consuming and important part of the project as poor data in gives poor performance out.

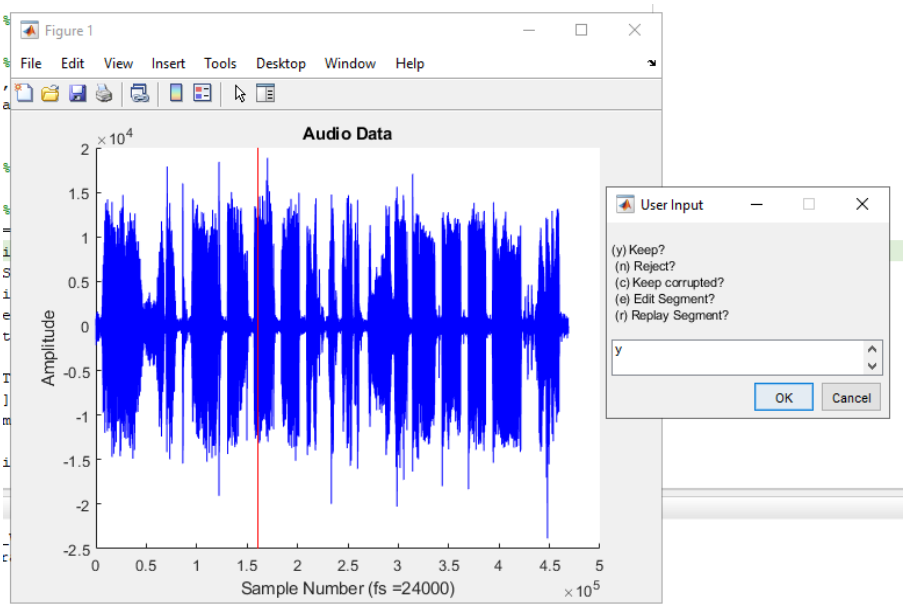


Figure 4: Observer Edit GUI

For the dataset that is used in this thesis, the data from the output of the "observer" totaled over 15 hours of .wav audio, amounting to about 3 Gb spanning 168 unique infants. The 168 infants were classified as 84 healthy controls, 72 symptomatic NAS, and 12 with ambiguous classification which were later excluded from analysis. Each infant has one or more cry episodes lasting between a few seconds to several minutes each. A cry episode primarily consists of a number of "wahs" which we call "long utterances", and are separated briefly by pauses or breaths, which we label as either short utterances, when there is energy present, or silence when there is no.

As there were eventually 2,903 episodes from all the data, clearly, each infant had some variable number of episodes in the dataset, each of variable length. Each episode has labelling data which follows a strict naming scheme that lists the patient ID, date of recording, time of recording, and clinical diagnosis, although the specific patient identity is totally hidden from the researchers by hospital privacy rules.

3. Analysis for Features

3.1. Feature Analyzer Iterations

Observed audio files are then processed by a DSP analyzer. The analyzer calculates several acoustic parameters on each audio sample that are outputted as a .csv file. Popular speech analysis parameters include mel-frequency cepstral coefficients, short-time energy, and formant information but are more tailored to adult speech rather than to infant cries (Saraswathy, Hariharan and Yaacob). We use three iterations of analyzers for this research guided by signal processing domain knowledge. Initially, the analyzer that has been developing at Brown for over 10 years was used to provide the features for the later ANN classification system. This has been fully described in Reggiannini et al (Reggiannini, Sheinkopf and Silverman). It was seen quite early in the research that the more high-level output from Phase 2 from this analyzer were too sparse to be able to train the ANN system. Thus, a subset of the denser Phase 1 parameters was used in the early part of this research.

Figure 5 highlights the three versions of our Analyzer systems with the common data preprocessing utilities in black, Stripper and Observer. Each version progresses towards data-centric, away from model-centric, becoming less dependent on the domain expert knowledge. The leftmost thread in red involves two phases: Phase 1 based on Reggiannini Parameters to detect energy-cased acoustic features of infant cries and Phase 2 analysis on an utterance level. Section 3.2 describes this earliest developed version. The middle thread in yellow shifts the focus to pitch-based parameters more applicable to NOWS diagnosis, which is explained in Section 3.3. The rightmost thread in blue adds CNN to extract features from the raw data instead of defining features by the domain experts. Section 3.4 details its design, exemplifying the success of data-centric machine learning approaches.

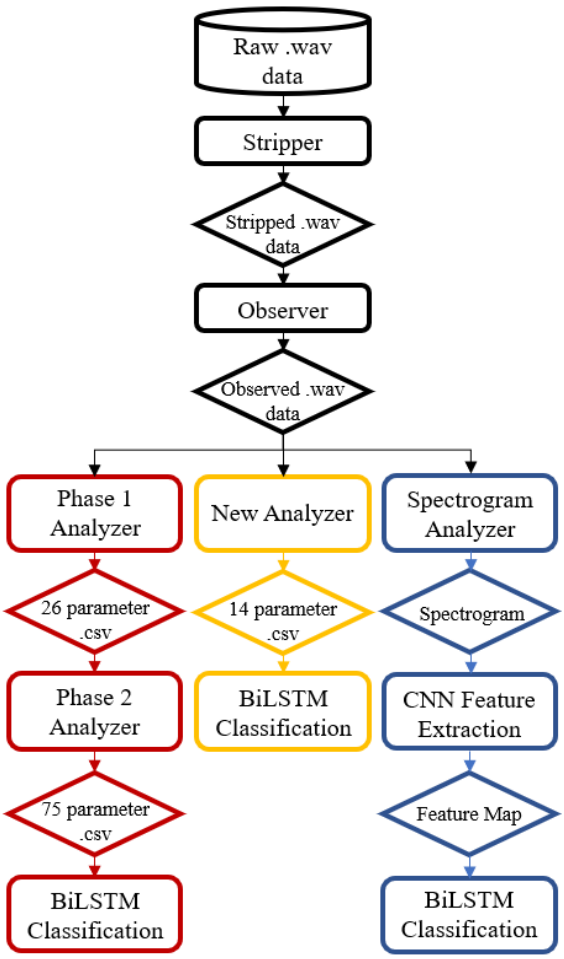


Figure 5: Analyzer systems

3.2. Reggiannini Parameter Analyzer

Initially, combinations of Reggiannini et al. Phase 1 infant cry analysis parameters were used and are listed in Table 3 below (Reggiannini, Sheinkopf and Silverman). This system was started in 2011 as a general analysis tool, accommodating for all kinds of infant data and sampling rates. As a result, it became a rather bloated system fitting many features and coauthored by over 20 people in the last ten years. Coded in MATLAB, this analyzer takes observed data and operates on individual cry episodes. After resampling at 48,000 Hz at floating point precision, the analyzer conducts calculations over a 25 ms hamming window with an advance rate of 600 samples, or 12.5 ms. It uses cepstral analysis (features

from the inverse log spectrum) and several other heuristics to account for a wide range and variety of infant cries. The output would be more uniform estimates on pitch and pitch-like parameters for analysis. This includes measures of amplitude, pitch estimate confidence, spectral ranges, and formants. The 26 parameter vector is shown in the table below as "phase 1" output.

Table 3: 26 Phase 1 Parameters

Fr#	Frame number
Time (ms)	(Frame number)*12.5msecqua
Pitch (Hz)	Fundamental frequency of frame
Pitch En. (dB)	Energy (power in dB) for pitch identified in frame
Confidence	Confidence level, # from [0,1] that describes pitch identification confidence
Hyper-Pitch (Hz)	[1000,3000Hz] range
Hyper-Pitch Energy (dB)	Energy (power in dB) for hyper-pitch identified in frame
Hyper-Pitch Confidence	Confidence level, # from [0,1] that describes hyper-pitch identification confidence
Peak En. (dB)	Energy (power in dB) for frame
Tot. En. (dB)	Total energy (power in dB) for frame
.5-10kHz Energy (dB)	Energy across [0.5-10kHz]
0-.5kHz Energy (dB)	Energy across [0-0.5kHz]
.5-1kHz Energy (dB)	Energy across [0.5-1kHz]
1-2.5k Hz Energy (dB)	Energy across [1-2.5kHz]
2.5-5k Hz Energy (dB)	Energy across [2.5-5kHz]
5-10k Hz Energy (dB)	Energy across [5-10kHz]
FM1 (Hz)	First formant's frequency in frame
Mag (dB?)	First formant's energy (power in dB)

FM2 (Hz)	Second formant's frequency in frame
Mag (dB?)	Second formant's energy (power in dB)
FM3 (Hz)	Third formant's frequency in frame
Mag (dB?)	Third formant's energy (power in dB)
Vuv	Logical vector containing 1 for voiced and 0 for unvoiced frames pitched frames
hvuv	Logical vector containing 1 for voiced and 0 for unvoiced frames for hyper-pitched frames
Spectral Change	Change in spectral power
Cepstral Change	Change in cepstral power

Another set of parameters is derived from phase 1 parameters, called “phase 2”. The phase 2 parameters are more aligned with current research on infant-cry analysis (Lester). Instead of a per-episode basis, each long utterance, short utterance, and silence interval produced a feature vector consisting of 75 parameters. These parameters emphasize utterance contour and hyper pitch among others. Table 8 in Appendix A lists the 75 phase 2 parameters.

This is a significant data reduction from phase 1. For example, a three-second-long utterance will produce 240 26-parameter phase 1 vectors but only from 1→5 75-parameter phase 2 vectors. Thus, we will have much less training data available to sufficiently train the ANN. Moreover, making early decisions on the data instead of using data itself increases the effects of any error, which harms ANN performance. Another analyzer is needed to better parameterize NAS classification.

3.3. Pitch-Based Analyzer

In the summer of 2020, another analyzer was coded using a different set of pitch-based parameters more applicable to NAS diagnosis. In discussion with Prof. Silverman, it was agreed that emphasizing pitch, pitch contour, and pitch confidence as much as possible is a step in the right direction. It also greatly simplifies the analyzer code and produces a "phase 1-like" output of fourteen parameters that did not include spectral bands, formants, or hyper pitch. The fourteen consists of amplitude (1), spectral change (1), spectral harmonic peaks (4), simplified cepstrum (4), and comb-filter correlation (4). Each of the latter three analyses have the following four measures: Highest pitch value, amplitude of the highest pitch peak, second best estimator of the pitch, amplitude of the second-highest peak. Each measure includes three parameters: actual estimate, first harmonic peak, and second peak. Peak measurements are a measure of confidence. These are listed in Table 4 below.

Table 4: Pitch-based Parameters

Amplitude	log-scale power of frame in frequency domain
Spectral Change	Difference in spectral power from last frame
Spectral Low Peak	Frequency of lowest harmonic in spectral space
Spectral Low Value	Height of lowest harmonic in spectral space
Spectral High Peak	Frequency of second lowest harmonic in spectral space
Spectral High Value	Height of second lowest harmonic in spectral space
Cepstral High Peak	Frequency of highest cepstral peak
Cepstral High Value	Height of highest cepstral peak
Cepstral Second High Peak	Frequency of second highest cepstral peak

Cepstral Second High Value	Height of second highest cepstral peak
Comb Filter High Peak	Frequency of highest comb-filter peak
Comb Filter High Value	Height of highest comb-filter peak
Comb Filter Second High Peak	Frequency of second highest comb-filter peak
Comb Filter Second High Value	Height of second highest comb-filter peak

The analyzer is run on all observed .wav files to produce 164 .csv files. A screenshot of an analyzer output is shown below in Figure 6. Each column contains the 14 acoustic parameters calculated over 12.5 millisecond frames that make up a row. 160 rows of zeros separated each episode in the analyzed .csv.

Ampl(dB)	Spec chng	SpecLowP	SpecLowV	SpecHighF	SpecHigh\	CepsHighI	CepsHigh\	Ceps2Pk(I	Ceps2Val	CombHigh	CombHigh	Comb2Pk(	Comb2Val
124.03	0	350.1	1	687.74	0.93301	352.94	0.13638	387.1	0.084338	341.03	0.078974	367.11	0.063254
116.4	0.86299	380.86	0.82358	756.59	0.70085	369.23	0.093287	328.77	0.076984	606.64	0.012172	353.92	0.012025
112.04	0.49617	158.94	0.56814	689.94	0.63624	228.57	0.039205	250	0.034388	251.97	0.013135	588.06	0.013038
107.8	0.28929	179.44	0.44592	697.27	0.55031	827.59	0.073988	338.03	0.011346	250	0.054526	277.26	0.051437
102.91	0.19307	172.12	0.23075	694.34	0.29546	827.59	0.024488	615.38	0.012727	253.13	0.050004	249.67	0.045545
102.79	0.11852	162.6	0.3621	621.83	0.29732	585.37	0.05295	311.69	0.01549	635.76	0.046642	293.58	0.040267
104.18	0.12337	169.19	0.33681	682.62	0.35792	347.83	0.10253	888.89	0.034743	279.27	0.033282	280.5	0.029577
104.04	0.12501	205.81	0.23814	676.03	0.37599	888.89	0.04214	480	0.039729	356.22	0.007842	304.04	0.006817
105.46	0.15143	164.79	0.36462	689.21	0.29828	888.89	0.13435	214.29	0.024757	277.26	0.025036	386.71	0.018681
106.17	0.15879	169.92	0.35914	680.42	0.25894	545.45	0.054985	224.3	0.054939	379.82	0.015069	280.5	0.01138
104.09	0.15375	150.88	0.46152	211.67	0.28624	960	0.022746	413.79	0.009934	303.8	0.002519	378.33	0.002109
104.77	0.14783	231.45	0.19488	908.2	0.20434	600	0.077382	413.79	0.058113	470.59	0.073972	303.8	0.056426
101	0.13216	150.88	0.31664	902.34	0.19464	615.38	0.089483	320	0.021849	385.16	0.094206	378.33	0.079997
99.389	0.095933	247.56	0.28537	326.66	0.21578	705.88	0.032658	1000	0.031151	423.84	0.043135	358.21	0.038236
100.75	0.092093	180.18	0.31803	240.97	0.20424	585.37	0.076294	800	0.04729	354.57	0.067131	303.56	0.064649
98.173	0.080426	175.05	0.2509	235.84	0.16542	272.73	0.021015	558.14	0.012728	279.07	0.04449	308.19	0.037534
96.33	0.056951	150.88	0.33396	229.25	0.20456	585.37	0.024932	387.1	0.021213	309.18	0.058759	313.21	0.050207
98.829	0.062852	188.96	0.29749	391.85	0.20279	157.89	0.047086	585.37	0.007977	322.69	0.093392	308.93	0.076152
95.028	0.067849	375	0.25931	736.82	0.033548	436.36	0.074306	380.95	0.044915	260.34	0.14907	498.7	0.1364

Figure 6: Analyzer Output Example

3.4. CNN Analyzer

In the summer of 2021, another approach was taken that brought the analysis closer to the raw data level. Instead of making parameter decisions, the analyzer is adapted to output audio spectrograms of observed .wav data. The spectrograms spanned 0 to 8,000 Hz and is computed using a 2,048 sample padded discrete Fourier transform (DFT) to get 683 frequency samples. This gives an 11.718 Hz frequency resolution. Each spectrogram is

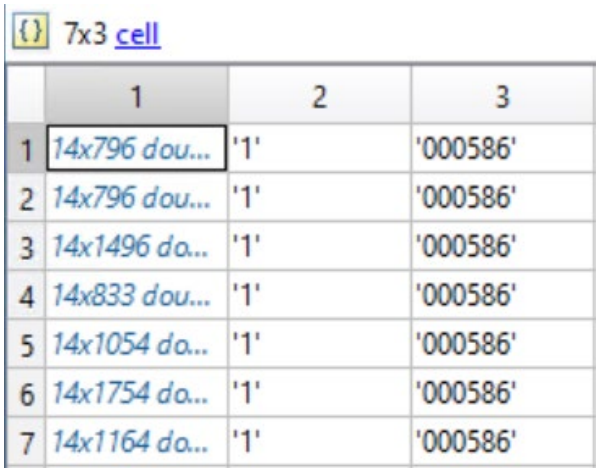
stored as a 683xN matrix where N is the number of frames. Like the previous analyzer, each frame is 12.5ms. This input minimized user decision making except, perhaps, to selecting the frame size and frame advance.

Key features in the spectrogram are extracted using a convolutional neural network (CNN) instead of calculating them through biomedical signal processing domain knowledge. Instead of making decisions on what to analyze, we allow the convolutional layers to identify correlated trends which may be more prevalent among the audio spectrograms. The 683 features are reduced to ~21 key features over the span of N frames by the convolutional ANN. The output resembles analyzer outputs from the previous analyzer and are then sent through a similarly structured BiLSTM classification network. Moving away from decisions requires more data but allows the neural network to learn sufficiently and make better decisions than the human experts.

4. Neural Network Architectures

4.1. Data Set Construction and Normalization

Once all .csv files are collected, the data undergoes preprocessing and formatting for BiLSTM training. Another program reformats the .csv data into MATLAB cell arrays. It also allows the user to choose which acoustic parameters to include in the data set. Each infant has a unique .mat file named by the subject ID. The .mat files are matrices with three columns and as many rows as there are cry episodes. The first column has cells containing analyzer parameter data for the corresponding episode. The second column has a Boolean for the episode's ground truth diagnosis, "1" if symptomatic or "0" if healthy. The third column holds the subject ID as given by the file name. Figure 7 below gives a screenshot of the data organization. It shows subject 000586 is symptomatic and has seven episodes. Eight parameters are included, and the episodes span from 796 to 1754 samples.


{} 7x3 cell

	1	2	3
1	14x796 dou...	'1'	'000586'
2	14x796 dou...	'1'	'000586'
3	14x1496 do...	'1'	'000586'
4	14x833 dou...	'1'	'000586'
5	14x1054 do...	'1'	'000586'
6	14x1754 do...	'1'	'000586'
7	14x1164 do...	'1'	'000586'

Figure 7: Infant Cell Array Example

The final data set consists of 34 symptomatic and 34 healthy subjects, each containing one or more episodes of various lengths. Several steps are taken such that every subject is equally represented in the data set during network training. A program

normalizes several data qualities and eliminates unintended bias. The number of episodes each subject contributes is determined so the BiLSTM would not correlate infants by episode quantity. Three episodes were chosen from analyzing the distribution of subjects per number of episodes as shown below in Figure 8. This histogram covers all available data which includes 94 healthy infants and 34 symptomatic infants. 71% of subjects in the data set contain at least three episodes. Subjects with fewer than three episodes have their final episode copied to reach three. Given adequate training random selection, repeated episodes had negligible effect on training. After several random selections among the entire data set, it is also observed the number of repeats in the validation set have little effect on training. There is insufficient data to fully avoid repeated episodes while maintaining equal infant representation in the data set.

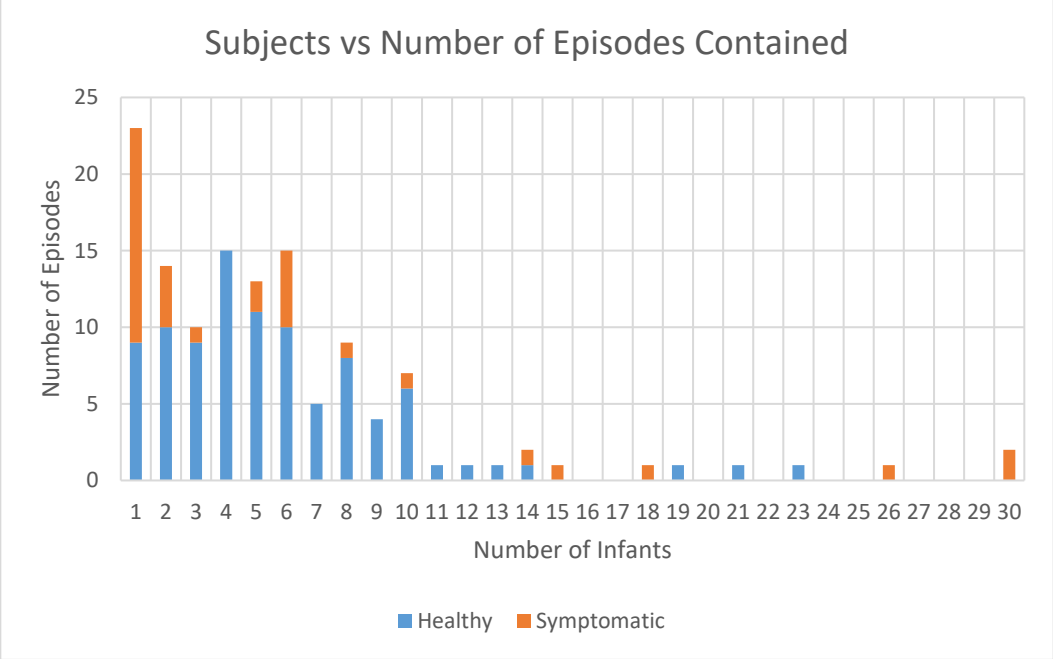


Figure 8: Subjects vs Number of Episodes Contained

The episodes are also normalized by length so the network would not correlate infants with significantly short or long episodes. Based on the episode length histogram in

Figure 8, lengths are normalized to be between 702 and 1,600 frames (8.78-20 seconds). Figure 9 displays the number of episodes with lengths falling in each bin for 94 healthy and 34 symptomatic infants. Episodes under 702 frames are replaced with another episode in the length threshold if the subject has more than three episodes. Otherwise, the longest episode is used as replacement. Episodes over 1,600 frames are cut to the threshold. The length threshold also improves data set quality by removing short episodes that hold little information. According to expert domain knowledge, the most important acoustic features occur at the beginning of utterances for NAS.

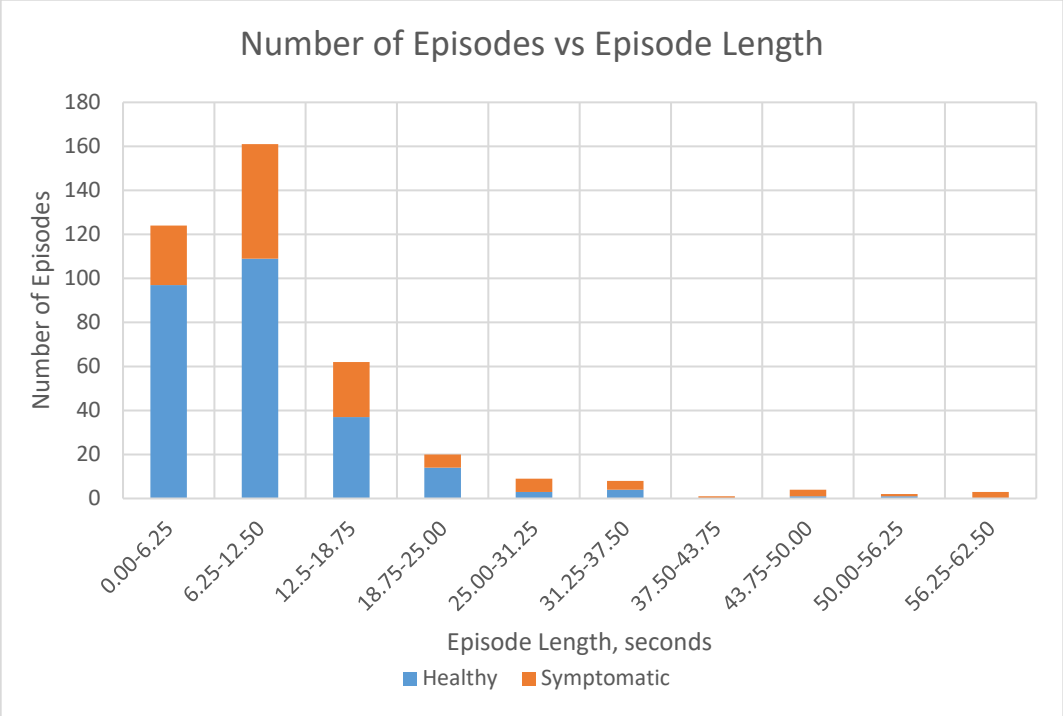


Figure 9: Number of Episodes vs Episode Length

Lastly, the number of subjects in each category (healthy or symptomatic) is also normalized. The class with more subjects has some removed until both classes have the same amount, ensuring both classes were equally represented in the data set. Afterwards,

the normalized data set is concatenated into one .mat file for the BiLSTM training code. This resulted in training sets with 34 healthy and 34 symptomatic.

4.2. LSTM Architecture

Given the sequential nature of infant cry data, the BiLSTM is the natural choice for classification. MATLAB Deep Learning Toolbox is used to create the machine learning network, consisting of a sequence input layer, BiLSTM layer, dropout layer, fully connected layer, softmax layer, and classification layer. The sequence input layer defines how many features are in the data which is one for this application. The BiLSTM layer conducts forward and backward propagation across the LSTM gates for a predefined hidden unit size. Hidden unit size determines how much information is retained between time steps. Each propagation direction traverses through the input, forget, cell candidate, and output gates. A dropout layer is used after the BiLSTM layer to randomly set input elements to zero and prevent overfitting. A fully connected layer applies weights and bias to the input, using Glorot initializer. The output is sent through a softmax layer normalizing values as probabilities between 0 and 1. Finally, the processed episodes are sent through a classification layer that outputs the network's decision on each. Figure 10 illustrates the machine learning architecture used.

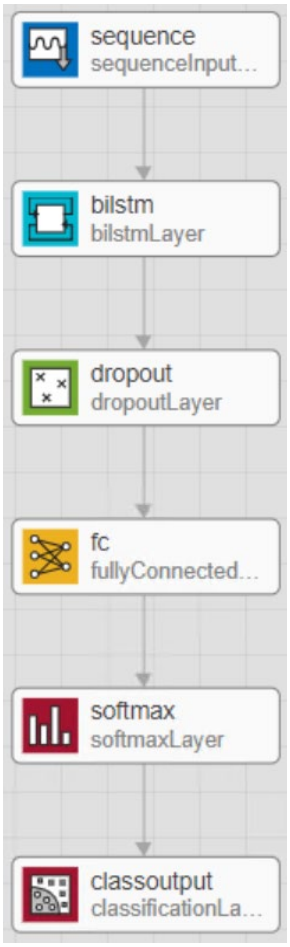


Figure 10: LSTM Network Architecture

Most hyper-parameters referenced MATLAB default values, but several are tuned during experimentation. Other architectures, such as two BiLSTM layers with two dropout layers, were tested but did not yield as high training/validation accuracy during training. Table 5 below summarizes the chosen hyper-parameters.

Table 5: LSTM Network Hyper-Parameters

Hidden Units	120
Dropout Probability	0.4
MiniBatch Size	10
Learning Rate	0.0002
Gradient Threshold	1
Optimizer	adam
Max Epochs	400

Data is randomly split between training and validation for the LSTM. This is done on a subject basis rather than episode basis, so each infant (contributing a fixed number of episodes) is assigned to the training or validation sets. A program takes in data and a user-specified number of subjects for validation and outputs two data sets along with a seed indicating which subjects are assigned to the validation set. The seed feature makes it possible to retrace the random selection and verify how different assignments reflected on training behavior. The training and validation data sets are then sent into the BiLSTM.

4.3. LSTM Training

MATLAB displays real-time deep learning training progress once a training program starts. Updating each iteration (forward and backward propagation traversal), the plots record accuracy and loss for the training and validation data sets. The horizontal axis is also subdivided by epoch which is a complete pass through the data set. The training accuracy denotes the network's accuracy on the training data set which is divided into mini-batches. The validation accuracy is the network's classification performance on the validation set. This is assessed after a specified number of iterations, called the validation frequency. The loss curves represent the cross entropy loss for each data set. Cross entropy loss maps a classification model prediction probability to a logarithmic scale that better represents accuracy. Error distance between predictions and ground truth is exponentially penalized.

Because there are two classes, training accuracy is expected to begin at 50%. 50% accuracy indicates both classes are equally likely in the binary categorization, so there is

no correlation between the parameters. As training progresses, accuracy should converge towards 100% while the loss should decrease to 0. The curves should smoothly resemble logarithmic growth and decay functions, reaching a steady state after sufficient epochs. The best number of epochs is decided from when training accuracy gain plateaued. Limiting training epochs also avoids overfitting, when validation accuracy began to decay after plateauing. Moreover, each pair of training and validation curves should closely follow each other, indicating the validation data is representative of the training data. Figure 11 gives an example of a training plot performing well. The blue data lines represent the performance after so many runs through the data for the training set itself. The black dots represent the performance of the independent validation set of 15 control and 15 symptomatic infants. The lower curve shows the error performance for the training set in orange and the black dots for the validation set.

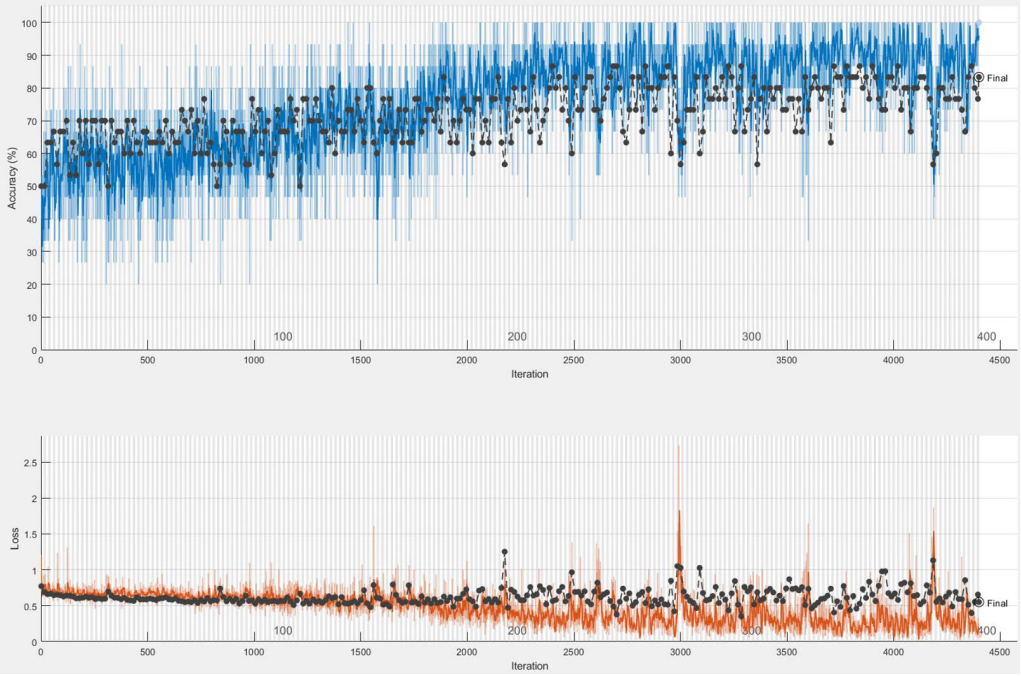


Figure 11: Example Training Plot

A test was conducted to verify BiLSTM effectiveness for classifying parameterized cry data. An artificial data set was constructed from one cry recording. The cry recording was copied 68 times and half of them had their pitch attenuated by some percentage to simulate a symptomatic category. The categorization network was then run on the artificial data set for various percentages and perfectly separated the classes down to a 20% attenuation. A plot for 20% is shown in Figure 12. This test demonstrated that the proposed MATLAB system is indeed capable of categorizing pitch-based parameters.

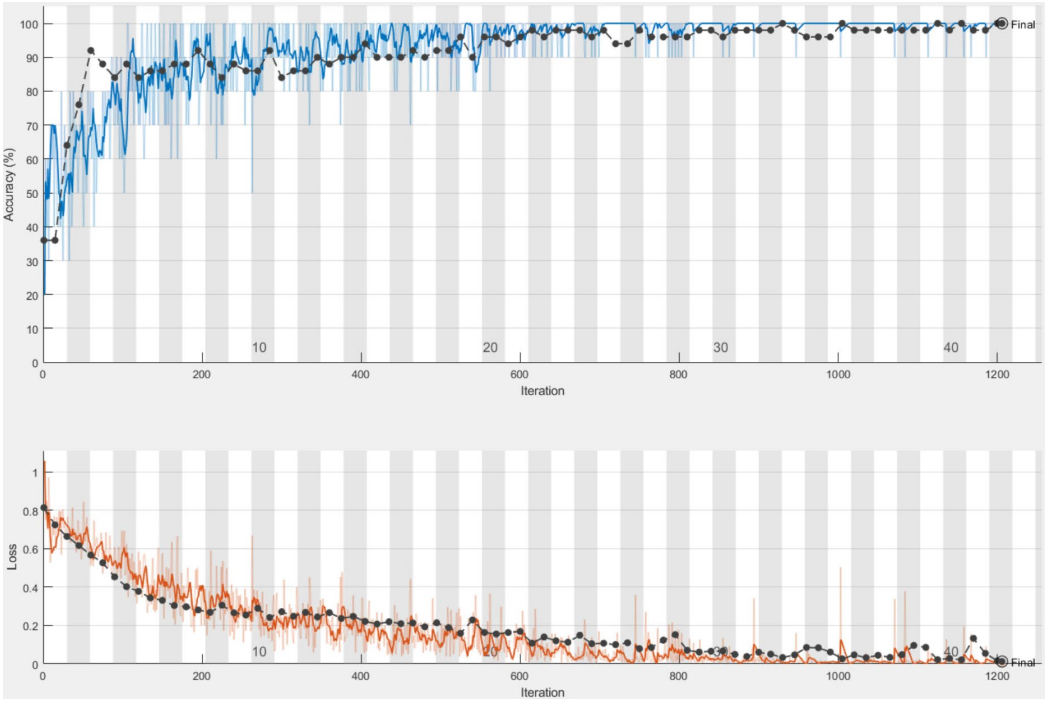


Figure 12: Training Curve for Artificial Data Set

Extensive testing is conducted to determine which of the fourteen acoustic parameters contributed most to classifying healthy or symptomatic infant cries. Each variation is verified at three degrees of randomization: same training/validation split with same control subjects, random training/validation split with same control subjects, and random training/validation split with random control subjects. Training performance is

ideally similar across these three randomizations, indicating data set homogeneity. However, some variation is inevitably encountered due to having insufficient data.

Based on domain knowledge, different sets of acoustic parameters are tested and yield varying performance. BiLSTM input weights are inspected to determine which parameters have more influence during training than others. These weights do not completely indicate correlation so the training curve itself provided valuable information. Good performance in training plots and large weight values indicated a parameter is contributing largely to training and helping classification.

Table 6 below highlights which parameters are removed for better performance. Peak parameters for spectrum, cepstrum, and comb are excluded because they contributed little magnitude in the input layer. Similar parameters are grouped together and ran in individual training runs. The groupings are highlighted in different colors below. Amplitude is highly correlated and had the greatest order of magnitude in the input weights matrix. Spectral parameters (change, low value, and high value) are fairly correlated. Cepstral and comb filter parameters are vaguely correlated but on orders of magnitude lower than amplitude and spectrum. These eight correlated parameters are used in the final network.

Table 6: Eight Pitch-Based Features Used

Amplitude	log-scale power of frame in frequency domain
Spectral Change	Difference in spectral power from last frame
Spectral Low Value	Height of lowest harmonic in spectral space
Spectral High Value	Height of second lowest harmonic in spectral space
Cepstral High Value	Height of highest cepstral peak
Cepstral Second High Value	Height of second highest cepstral peak

Comb Filter High Value	Height of highest comb-filter peak
Comb Filter Second High Value	Height of second highest comb-filter peak

Once parameters are determined, the number of episodes each subject contributed is adjusted. Subjects with more than three episodes would have more unique data to offer for the data set. However, subjects with three or less episodes will provide more repeats to compensate. It is trade-off between having more data to train on and overfitting on repeated episodes. Performance for five and seven episodes is compared to three episodes. Training with more episodes added an insignificant amount of new data to the data set compared to the number of added repeats. It is concluded that three episodes is optimal based on the episode per infant distribution in Figure 8. A test is conducted to verify that the number of repeated episodes in the validation set did not cause major variation in performance.

Different episode length thresholds are also tested. Initial training uses episodes between 702 to 1,600 samples in length. A stricter length threshold is implemented to check whether episode length is a hidden correlation being picked up. Training curves from a 702-1,600 sample data set are compared to ones of 702-800 samples under same network conditions. The plots are very similar, indicating length is not an unintended correlation. The 702-1,600 threshold is kept as it provides more data for each episode based on the episodes per length bin distribution in Figure 9.

LSTM network architecture underwent several iterations before reaching the final design choices described before. The BiLSTM's forward and back propagation yield consistently higher results than a LSTM layer. Systems with two BiLSTM layers have much different behavior compared to one layer as each epoch underwent twice as much

propagation. Two BiLSTM layers also yield less consistent training behavior. A dropout layer is added to reduce overfitting. This also improves learning consistency over randomized training data. 15-85% and 30-70% validation/training data splits are compared and yield similar performance.

4.4. Cry Spectrogram CNN-LSTM

The convolutional neural network (CNN) consists of three layers. These layers will apply a number moving filters of specified size onto each spectrogram. Each CNN layer is followed by corresponding batch normalization, rectified linear unit (ReLU), and 2D max pooling layers. Batch normalization accelerates CNN training and reduces overfitting by normalizing mini-batch data. The ReLU layer sets any negative inputs to zero. 2D max pooling layers down-sample the filtered and normalized input. The 683xN feature matrix is reduced to 171xN after the first, 43xN after the second, and 22xN after the third.

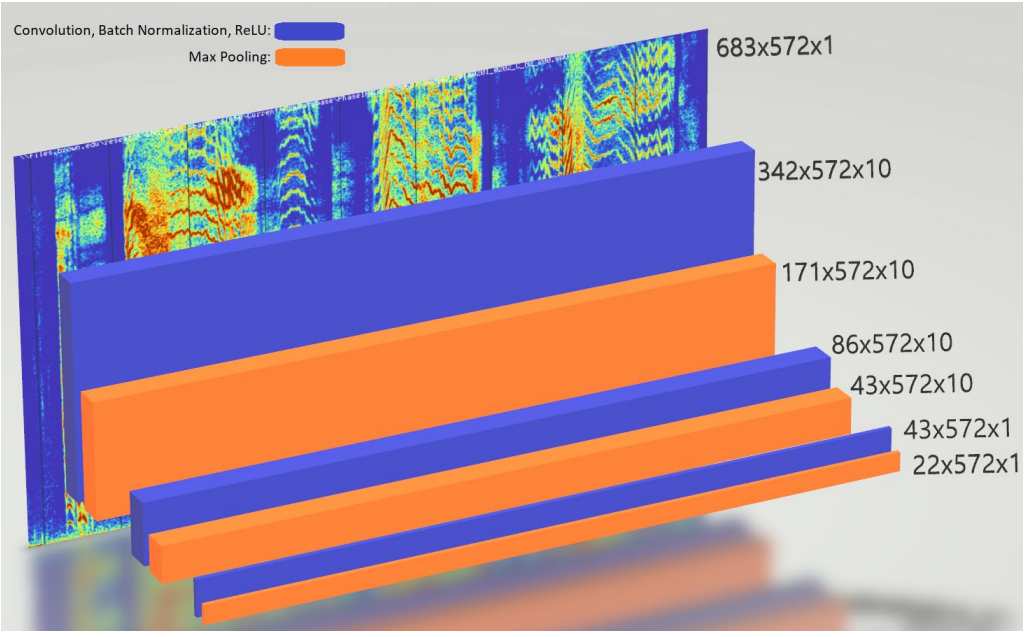


Figure 13: Convolutional Neural Layers for Spectrogram Analyzer

The reduced output is sent to the BiLSTM system, consisting of the same layers discussed before. The CNN and BiLSTM are joined by sequence folding/unfolding and flattening layer. These layers bridged the 2D CNN output with the sequential BiLSTM input. A table of the LSTM component of the system is shown below.

Table 7: CNN-LSTM Network Hyper-Parameters

BiLSTM Hidden Units	200
Dropout Probability	0.4
MiniBatch Size	15
Learning Rate	0.001
Gradient Threshold	1
Optimizer	adam
Max Epochs	200

Data set construction is adapted to function with the new architecture while preserving validity. The most challenging task was implementing a cry episode length restriction for CNN compatibility. The CNN layers require data matrices to have a fixed dimension despite cry episodes varying in length. Resizing was considered but not used to preserve important temporal components in the data.

A tradeoff formed between decreasing the episode length threshold to include more unique episodes but losing the amount of information each episode contributed. Infants may lose representation in the data set if all its episodes are shorter. Increasing the length threshold discards episodes below that length but increases the information each episode contains. Moreover, the length threshold should be representative of the average infant cry length to not contribute unintended bias. Inevitably, data is lost from discarding short episodes and trimming long episodes.

A 1,000-frame cutoff is tested and left 77 symptomatic and 109 control episodes in the data set. This is further lowered to 650 frames resulting in 127 symptomatic and 217 controls. Based off the episode length distribution, an episode length threshold of 572 frames (7.15 seconds) is selected to preserve as many symptomatic infant episodes within realistic cry length. The final data set consists of 23 symptomatic infants and 23 controls after balancing the classes. Each infant contributes 3 episodes to result in 138 episodes total compared to the 204-episode data set used in the BiLSTM-only network. Though it is hard to compare data size between systems, it is possible to compare infant representation in each.

5. Results

5.1. Reggiannini LSTM Performance

The Reggiannini analyzer output did not consistently perform well using the BiLSTM system. Training and validation accuracies consistently ranges between 50% and 80%. This data set is made up of 68 infants, half control and half symptomatic. Each infant contributes three episodes between 702 and 1,600 frames in length. These episodes contain 14 parameters highlighted in Table 3. It is concluded that the Reggiannini parameters may not be indicative of symptoms of NAS. Though these parameters offer a large batch of metrics, the RNN training performance did not adequately indicate that a difference could be found with them.

Figure 14 below displays a MATLAB training curve for the BiLSTM system on the Reggiannini output data set. The light blue line shows model accuracy on the training set. The dark blue line is a smoothed version of the training accuracy. This line better highlights long-term learning trends. The black dotted line shows the network's accuracy on the validation set, measured after every 15 epochs. It is an important measure on how resilient the network is to new data. In the plot below, the network quickly plateaus within the 500^{th} iteration and hovers between 70% and 80%. The results are insufficient to justify a correlation between the two classes from the incorporated parameters.

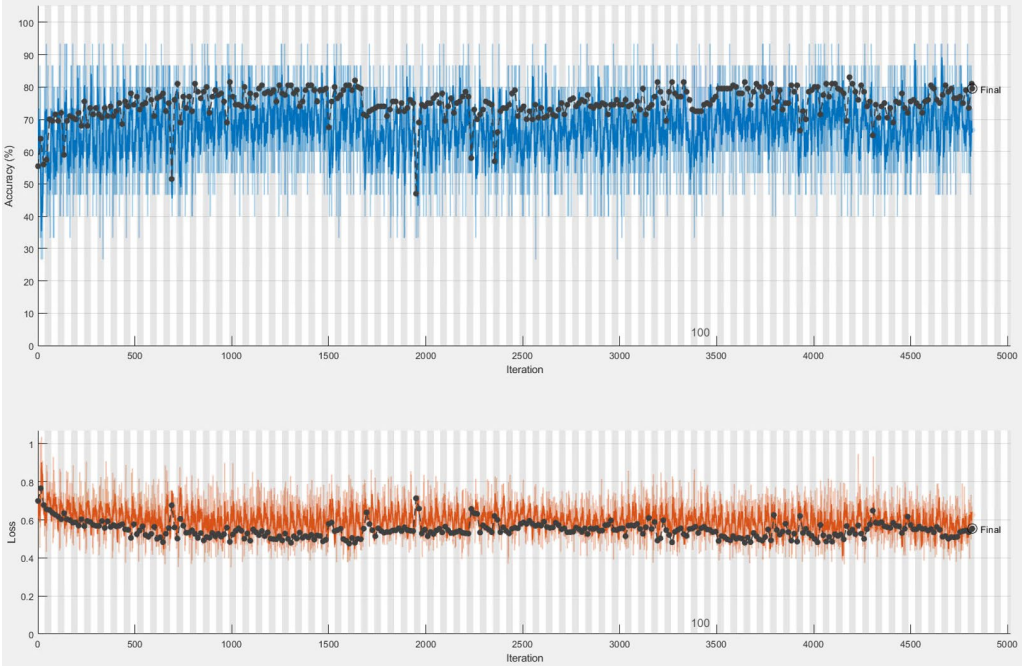


Figure 14: Reggiannini Parameter Training Curve

The classification network is applied in a testing program allowing the user to pass in episode data and get a % confidence diagnosis on subjects. A receiver operating characteristic (ROC) curve illustrates binary classification accuracy between true and false positive rates. ROC curves are generated for a randomly partitioned validation set of 20 infants, as shown in Figure 15. It is observed that most of the curves have ROC's ranging from 0.7 and 0.8.

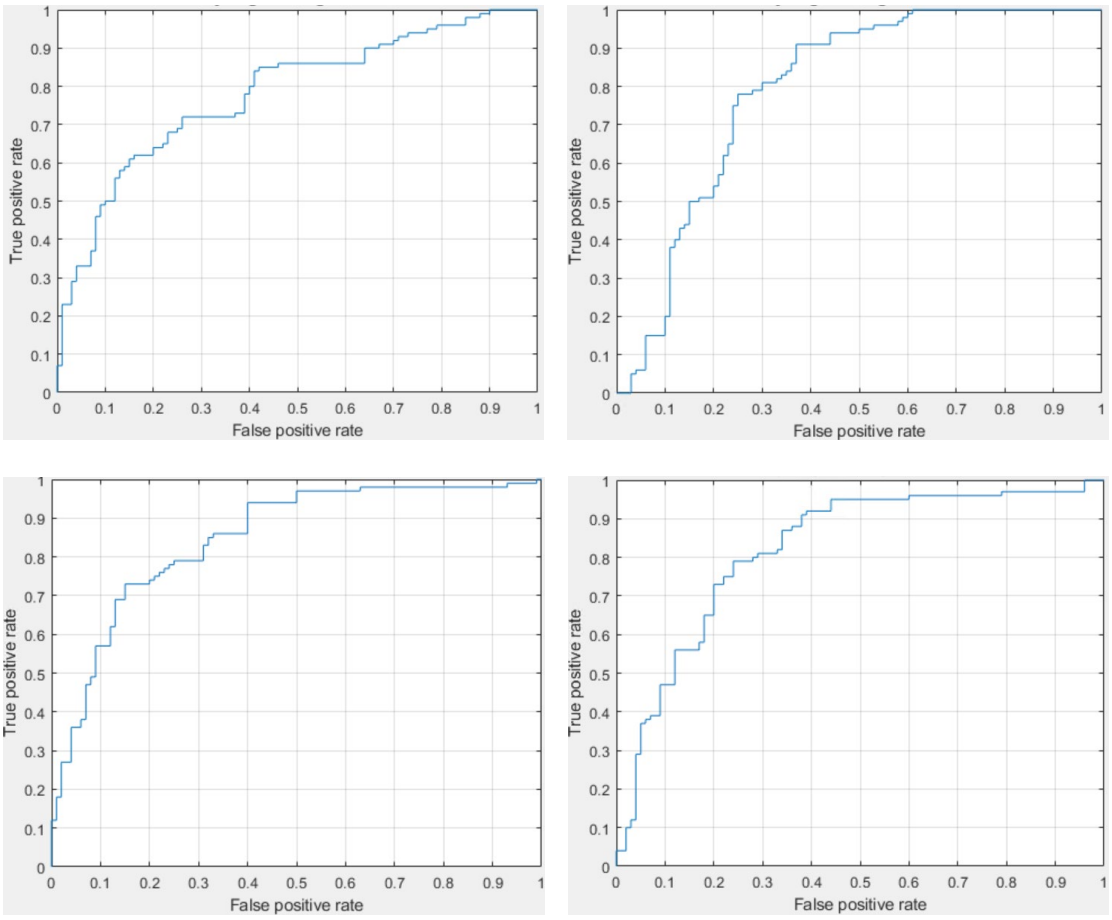


Figure 15: Reggiannini Parameter ROC Curves

5.2. Pitch-Based LSTM Performance

After iterative training and improvement, the pitch-based analyzer yielded good classification accuracy using the BiLSTM system. Like the previous data set, 68 infants (half control and half symptomatic) providing three episodes each between 702 and 1,600 frames are used. These episodes contain 8 parameters listed in Table 6. These parameters were selected from Table 4 after repeatedly running the training program on each individually. Parameters like spectral energy values performed well as shown in Figure 16 below. These parameters held significantly classifiable power in the data set.

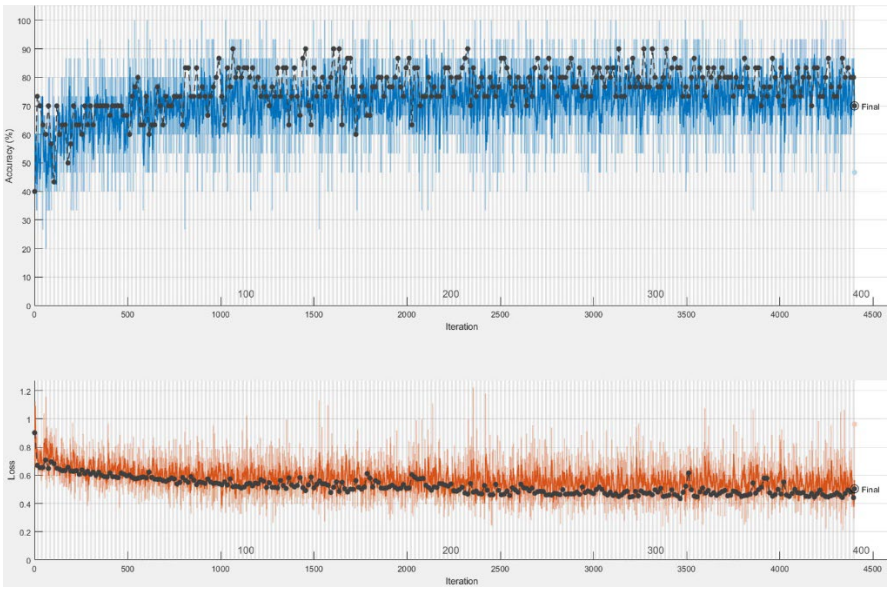


Figure 16: Training Curve of Spectral Low Value Parameter

On the other hand, Figure 17 shows training performance on Cepstral Second High Peak which was not a very correlated parameter in the data set.

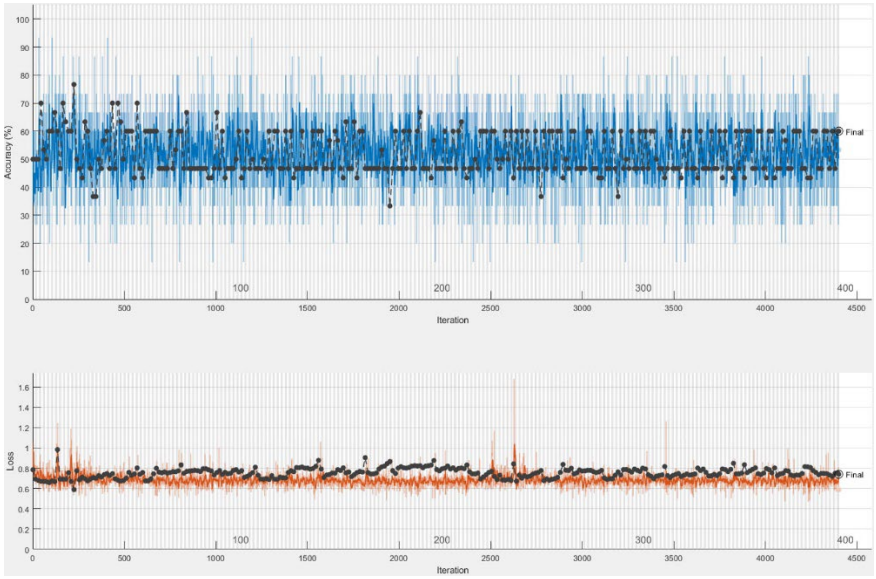


Figure 17: Training Curve of Cepstral Second High Peak Parameter

Given the available data and computation capabilities, there may be a significant correlation between healthy and symptomatic infant cries based on pitch parameters. This system performs well under three levels of randomization: fixed training and validation

data, randomized training and validation data, and randomized training and validation data with randomized control infants. A training plot for the randomized data split and control infants is shown in Figure 18. In each of the trials, accuracy consistently ranges between 80% and 100%. Loss continuously decreases towards 0.2. Unlike the prior training curve, the dark blue quickly rises after the 1000th iteration and continues to gradually rise with the validation curve until the last iteration. Training is halted after the validation accuracy exceeded the training accuracy to avoid overfitting.

Unsurprisingly, there is some difference in training and validation curves due to lack of data. Insufficient data causes the model to marginally overfit to the training set and perform slightly worse on the validation set. With more data, I expect my model to be more generalized with smoother accuracy/loss curves. The training and validation curves will be much closer as from generalization.

Figure 18: Pitch-Based Training Curve

The network is run in a testing program allowing the user to pass in episode data and get a % confidence diagnosis on subjects. A ROC curve is generated for a randomly partitioned validation set of 20 infants. The curves highlight the improved network accuracy from the more generalized parameter set. However, the curve is made up of less points as each infant contributes three episodes instead of five.

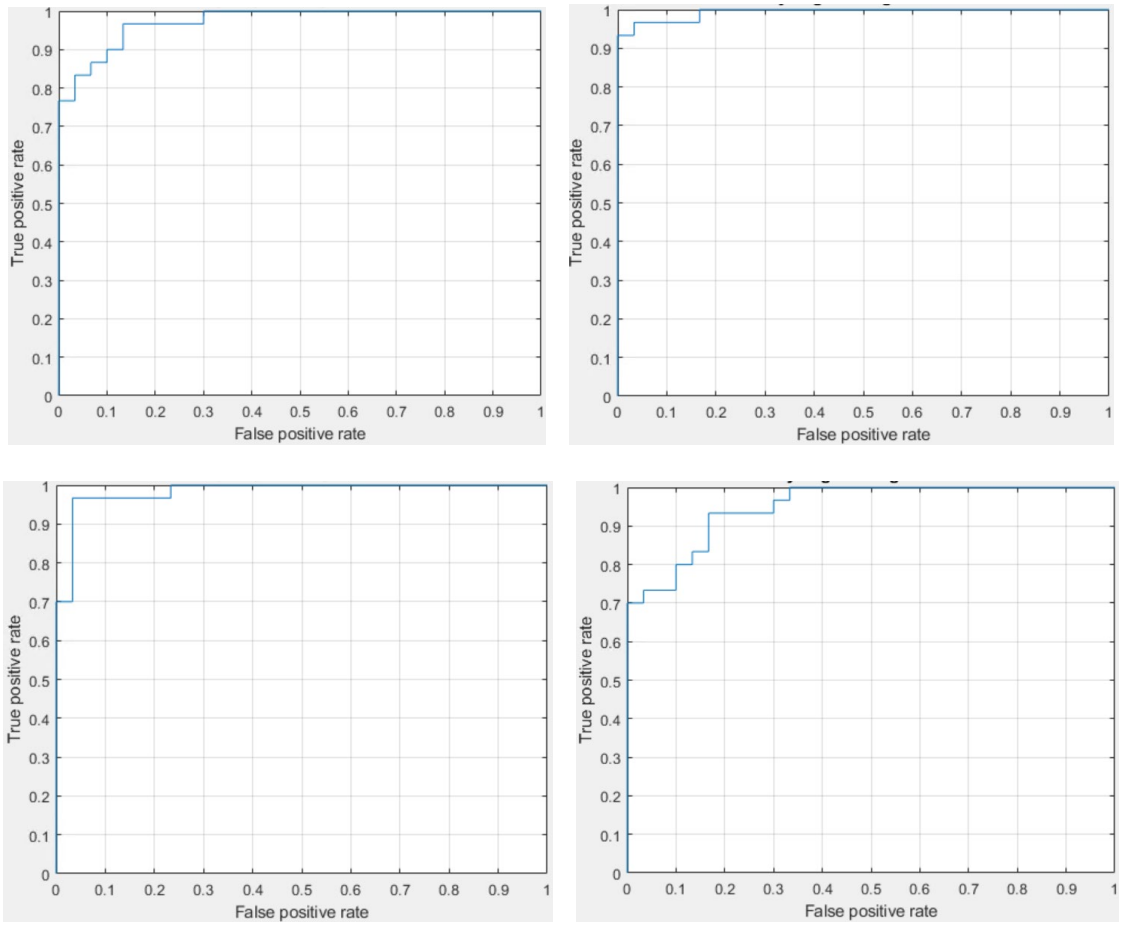


Figure 19: Pitch-Based ROC Curves

5.3. CNN-LSTM Performance

With the available spectrogram data, it is concluded that the audio spectrogram of cry episodes may significantly correlate between healthy and symptomatic infants. The network operates on the data set with 68 infants, half healthy and half symptomatic. Similarly, each infant contributes three episodes but each 572 frames in length.

This system also is tested under three levels of randomization. A training plot for the randomized data split and control infants is shown in Figure 20. The accuracy is consistently highly, between 80% and 100%. Loss continuously decreases towards 0. There is also difference in training and validation curves from insufficient data. The model is slightly overfit to the training set which is why the validation accuracy is slightly below the training accuracy. Increasing available data will expand model generalization and bring the training and validation curves closer.

Unlike the past two systems, the CNN-LSTM is quick to converge and plateaus around the 300th iteration. Moreover, the overall accuracy is far higher and more consistent than previous cases. From the consistently high accuracy and rapid convergence, it appears moving towards the raw data level resulted in a more resilient and accurate model. However, more data would be required to further validate these trends, especially given that the parameters were CNN-generated.

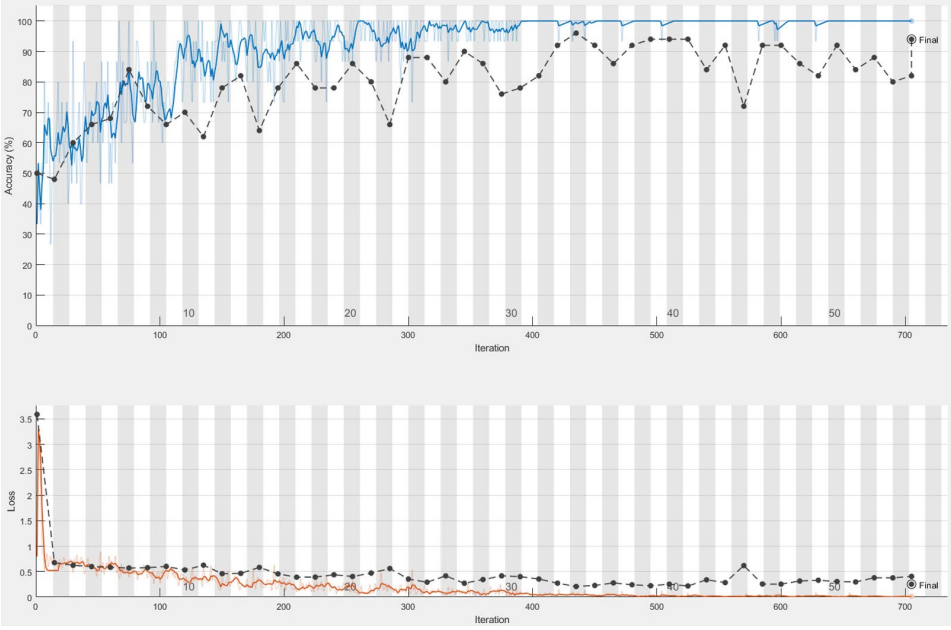


Figure 20: CNN-LSTM Training Curve

A ROC curve is generated for the validation set used in the prior curve. For a randomly partitioned validation set of 20 infants, it is less accurate than the pitch-based analyzer but more accurate than the Reggiannini analyzer. This is unsurprising as the CNN-LSTM needs more raw data to avoid overfitting.

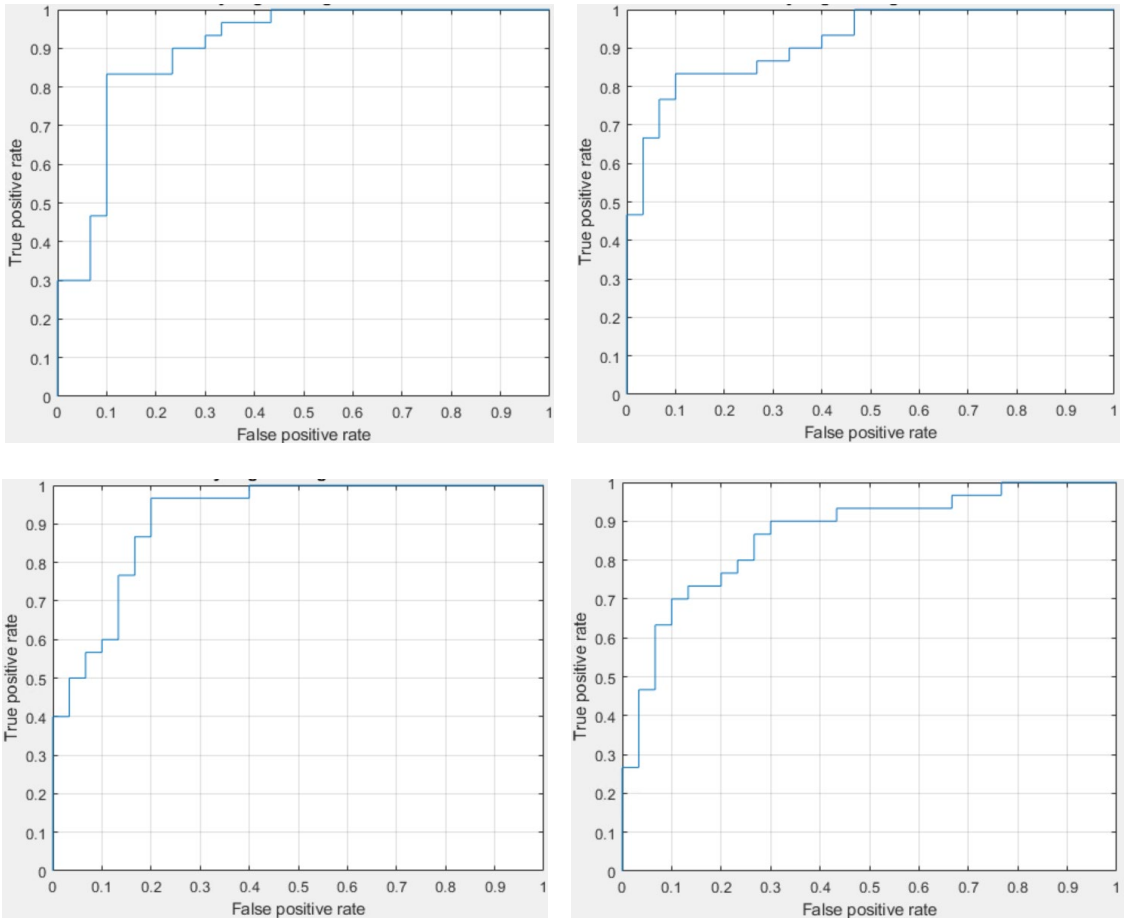


Figure 21: CNN-LSTM ROC Curves

5.4. Discussion

From these experiments, pitch-based parameters and audio spectrograms appear to be strongly correlated with NAS diagnosis given the data set. The Reggiannini parameters are too generalized for the LSTM to find a sufficient correlation, but the pitch-based parameters result in good classification. Taking steps towards the raw data level dramatically improved network performance as the CNN-LSTM system highlighted,

which confirms the current trend of deep learning for data-centric instead of model-centric. The parameters identified are pitch-based. These findings align with Finnegan test domain knowledge concerning high pitched cries as a symptom of NAS. A “shriller” pitch may be reflected in higher harmonic peaks captured by the spectral, cepstral, or comb estimates. This may also be attributed to greater energy in the higher bands of the audio spectrum.

However, these conclusions are limited to the scope of the data set. A larger quantity of data will solidify conclusions drawn from deep learning trends. The most limiting quantity in this study is the number of symptomatic infant cry samples which restrict how many healthy cries could be incorporated. A considerable amount of raw data is discarded during preprocessing to eliminate unintended bias. The lack of data is clear given inconsistent behavior between training runs among signs of overfitting. Outlier effects are more influential in a smaller sample size. These limitations are compensated with extensive randomization and repetition. As more decisions are made by the neural network, more data is required for valid conclusions.

6. Conclusion and Future Work

6.1. Conclusion

Neonatal abstinence syndrome is a growing health crisis among infants that calls for extensive and careful treatments. Otherwise, newborns may suffer from a wide range of harmful symptoms The Finnegan test, the current NAS metric, holds significant potential for further objectivity in its scoring mechanism. The assessment's subjectivity greatly risks mistreatment that will attribute crucial health and financial concerns among others. One highly subjective assessment in the Finnegan test is "high pitched crying" that varies depending on the environment and human assessor. Cry diagnostics is a growing field in infant cry analysis that may offer solutions to the Finnegan test's subjectivity issues. Signal processing for machine learning to classify infant cries offers a more objective assessment that will improve treatment effectiveness for infants suffering from NAS.

The conclusions from this study are limited by the available data for training and validation. The initial raw data is greatly curated to focus on this study's application. Experimental consistency over randomization and repetition further validates this paper's conclusions over a general scope. With the available data, we can achieve nearly 90% classification accuracy on the LSTM and CNN-LSTM systems. This thesis presents a novel application of signal processing for machine learning in biomedical diagnosis. Furthermore, it offers a more objective solution to the Finnegan test using infant cry that can be widely deployed. It also highlights powerful data-centric machine learning capabilities to solve classification tasks in noisy conditions.

6.2. Future Work

This study has large potential for improvement from expanding the functioning data set. Due to the on-going COVID 19 pandemic, infant cry collection was particularly limited at the time of this study. A large quantity of available data was excluded during preprocessing due to patient privacy and quality control. In the future, increasing the amount of usable data will vastly improve this paper's goals. An alternative way to resolve data shortage is for signal reconstruction by a generative model, successfully used in image recovery (Xu, Zeng and Romberg). However, the data-centric movement leads machine learning to an opposite direction: "Small Is the New Big" says AI pioneer Andrew Ng (Strickland). By systematically engineering the good data, although small, big issues in machine learning such as model accuracy can be solved.

The next step in the deployment of this paper is testing in a simulated environment. The classification system can be compiled to a readily usable executable program that will be run on a laptop. The program will be able to record in real time and produce a probability that the sampled infant cry displays symptoms of NAS. Cries may be replicated through playing recordings with a high-definition speaker. The speaker must emulate an infant cry without discarding essential audio qualities. Noise may also be injected to measure the classification system's resilience.

After verifying performance in a simulated environment, the system will be deployed in a real hospital environment with infants. Tests will compare the systems performance to the simulated environment. It is an important measure if the system is feasible for its intended application for nurses in real time diagnosis. Other considerations include distribution shifts when the distribution of training data differs from that of testing data,

which degrade the model's accuracy in deployment (Koh, Sagawa and Marklund). Therefore, the machine learning lifecycle extends with a deployment plan to track distribution shifts and retrain the model incrementally. These steps progressively bring the study closer to its real deployment.

Bibliography

Ashmore, Rob, Radu Calinescu and Colin Paterson. "Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges." *ACM Computing Surveys* 54.5 (2021): 111:1-39.

Bano, Sameena and K M RaviKumar. "Decoding baby talk: A novel approach for normal infant cry signal classification." *2015 International Conference on Soft-Computing and Networks Security (ICSNS),*. IEEE, 2015. pp. 1-4. doi: 10.1109/ICSNS.2015.7292392.

Bronstein, Michael M, et al. "Geometric Deep Learning: Going beyond Euclidean data." *IEEE Signal Processing Magazine* July 2017: 18-42. <https://ieeexplore.ieee.org/document/7974879>.

Chin Foo, Claire A, et al. "Improving the Assessment of Neonatal Abstinence." *Children* (2021): 8, 685. doi: 10.3390/children8080685.

Cohen, Rami and Yizhar Lavner. "Infant cry analysis and detection." *2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel*. IEEE, 2012. pp. 1-5. doi: 10.1109/EEEI.2012.6376996.

Devlin, Lori A, et al. "Association of a Simplified Finnegan Neonatal Abstinence Scoring Tool With the Need for Pharmacologic Treatment for Neonatal Abstinence Syndrome." *JAMA network open vol. 3,4 e202275* (2020). doi:10.1001/jamanetworkopen.2020.2275.

Foo, Lee Sze, et al. "Real-Time Baby Crying Detection in the Noisy Everyday Environment." *2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC)*. IEEE, 2020. pp. 26-31. doi: 10.1109/ICSGRC49013.2020.9232488.

Garcia, J O and C A Reyes Garcia. "Mel-frequency cepstrum coefficients extraction from infant cry for classification of normal and pathological cry with feed-forward neural networks." *Proceedings of the International Joint Conference on Neural Networks, 2003*. IEEE, 2003. pp. 3140-3145 vol.4. doi: 10.1109/IJCNN.2003.1224074.

Geirhos, Robert, et al. "Comparing deep neural networks against humans: object recognition when the signal gets weaker." *Conference on Neural Information Processing Systems (NeurIPS)*. 2018. <https://arxiv.org/pdf/1706.06969.pdf>.

Graves, Alex, Abdel-rahman Mohamed and Geoffrey Hinton. "Speech Recognition with Deep Recurrent Neural Networks." *International Conference on Acoustic, Speech and Signal Processing (ICASSP)*. 2013. 6645-6649.

Graves, Alex, et al. "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks." *Proceedings of the 23rd International Conference on Machine Learning (ICML)*. Pittsburgh, PA, 2006. 369–376. <https://www.cs.toronto.edu/~graves/icml_2006.pdf>.

Heershey, Shawn, et al. "CNN architectures for large-scale audio classification." *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017. pp. 131-135. doi: 10.1109/ICASSP.2017.7952132.

Hinton, Geoffrey, et al. "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups." *IEEE Signal Processing Magazine* November 2012: 82 - 97.

Hochreiter, Sepp and J□urgen Schmidhuber. "Long Short-Term Memory." *Neural Computation* 9.8 (1997): 1735-1780.

Koh, PangWei, et al. "WILDS: A Benchmark of in-the-Wild Distribution Shifts." *Proceedings of the 38th International Conference on Machine Learning (ICML), Proceedings of Machine Learning Research (PMLR)* 139 (2021): 5637-5664. <http://proceedings.mlr.press/v139/koh21a/koh21a.pdf>.

Lester, Barry. "Acoustic Cry Analysis Predicts Diagnosis of Neonatal Opioid Withdrawal Syndrome." *Internal Communications*. 15 December 2020.

MathWorks. *Specify Layers of Convolutional Neural Network*. n.d. <https://www.mathworks.com/help/deeplearning/ug/layers-of-a-convolutional-neural-network.html>.

—. *What is a Convolutional Neural Network?* n.d. <https://www.mathworks.com/discovery/convolutional-neural-network-matlab.html>.

Mima, Yuki and Kaoru Arakawa. "Cause Estimation of Younger Babies' Cries from the Frequency Analyses of the Voice - Classification of Hunger, Sleepiness, and Discomfort -." *2006 International Symposium on Intelligent Signal Processing and Communications*. IEEE, 2006. pp. 29-32. doi: 10.1109/ISPACS.2006.364828.

Nakkiran, Preetum, et al. "Deep double descent: Where bigger models and more data hurt." *International Conference on Learning Representations (ICLR)*. 2020. <https://openreview.net/pdf?id=B1g5sA4twr>.

Nguyen, Thi Ngoc Tho, et al. "A General Network Architecture for Sound Event Localization and Detection using Transfer Learning and Recurrent Neural Network." *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021. <https://arxiv.org/pdf/2011.07859.pdf>.

Olah, Christopher. *Understanding LSTM Networks*. 27 08 2015. <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.

Paleyes, Andrei, Raoul-Gabriel Urma and Neil D Lawrence. "Challenges in Deploying Machine Learning: a Survey of Case Studies." *ML Retrospectives, Surveys & Meta-Analyses (ML-RSA) Workshop at Conference on Neural Information Processing Systems (NeurIPS)*. 2020. <https://arxiv.org/abs/2011.09926>.

Pascanu, Razvan, Tomas Mikolov and Yoshua Bengio. "On the difficulty of training Recurrent Neural Networks." *Proceedings of the 30th International Conference on Mahcine Learning (ICML)*. 2013. 1310-1318. <https://arxiv.org/abs/1211.5063>.

Piczak, Karol J. "Environmental sound classification with convolutional neural networks." *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2015. pp. 1-6. doi: 10.1109/MLSP.2015.7324337.

Press, Gil. "Andrew Ng Launches A Campaign For Data-Centric AI." *Forbes* 16 June 2021. <https://www.forbes.com/sites/gilpress/2021/06/16/andrew-ng-launches-a-campaign-for-data-centric-ai/?sh=630a23c674f5>.

Reggiannini, Brian, et al. "A Flexible Analysis Tool for the Quantitative Acoustic Assessment of Infant Cry." *Journal of Speech, Language, and Hearing Research* 56.5 (2013): 1416–1428.

Saraswathy, J, et al. "Automatic classification of infant cry: A review." *2012 International Conference on Biomedical Engineering (ICoBE)*. IEEE, 2012. pp. 543-548. doi: 10.1109/ICoBE.2012.6179077.

Strickland, Eliza. "Andrew Ng, AI Minimalist." *IEEE Spectrum* April 2022: 22-25 & 50.

Sun, Lifa, et al. "Voice conversion using deep Bidirectional Long Short-Term Memory based Recurrent Neural Networks." *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2015. <https://ieeexplore.ieee.org/document/7178896>.

Sutskever, Ilya. *Training Recurrent Neural Networks. PhD thesis*. University of Toronto, 2013. <https://www.cs.utoronto.ca/~ilya/pubs/ilya_sutskever_phd_thesis.pdf>.

Truby, H M and J Lind. *Cry Sounds of the Newborn Infant*. Stockholm, Sweden: Wenner-Gren Research Laboratory, Norrtull's Hospital,, 1965.

Tuduce, Rodica Ileana, et al. "Automated Baby Cry Classification on a Hospital-acquired Baby Cry Database." *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2019. pp. 343-346. doi: 10.1109/TSP.2019.8769075.

Tuduce, Rodica Ileana, Horia Cucu and Corneliu Burileanu. "Why is My Baby Crying? An In-Depth Analysis of Paralinguistic Features and Classical Machine Learning Algorithms for Baby Cry Classification." *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2018. pp. 1-4. doi: 10.1109/TSP.2018.8441363.

University of North Carolina School of Social Work. *North Carolina Pregnancy & Opiod Exposure Project*. 2018. WebPage. 2022. <https://ncpoep.org/guidance-document/neonatal-abstinence-syndrome-overview/neonatal-abstinence-syndrome-nas/>.

Xu, Shaojie, Sihan Zeng and Justin Romberg. "Fast Compressive Sensing Recovery Using Generative Models with Structured Latent Variables." *IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP). 2019. <https://ieeexplore.ieee.org/document/8683641>.

Zhang, Aonan, et al. "Fully Supervised Speaker Diarization." *2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2019. <https://ieeexplore.ieee.org/document/8683892>.

Appendix A: Long Tables

Table 8: 75 Parameter Phase 2 Output

Cry Analyzer Output Parameter Descriptions (in order) (For each utterance)		
Five Various Sound Segment Classifier outputs [1-5]		
1	**Start Fr#**	Start frame of sound segment
2	**End Fr#**	End frame for sound segment
3	**Length (frames)**	Length of sound segment in terms of # of frames
4	**Length (ms)**	Length of sound segment in terms of ms
5	**Sound Segment Class**	0 is silence, 1 is short utterance, 2 is long utterance
Eight Timing Parameters [6-13]		
6	**Inter-Utterance(ms)**	Spacing between current long utterance and previous long utterance (0 for silence and short utterance)
7	**# of Short Utterances**	# of short utterances from last long utterance
8	**# of Low-confidence (unvoiced) frames**	# of low voicing confidence frames in sound segment
9	**# Fricative Voiced Sub-Segments**	# of fricative voiced sub-segments identified in sound segment (silence, short utt, or long utt)
10	**Percentage of Voiced Fricative frames**	% of sound segment (silence, short utt, or long utt) that has voiced frication present
11	**Voiced Fricative Dominant Decision**	assigned label "1" if percentage of voiced fricatives frames > 60%
12	**Longest Voiced Fricative Sub-Segment Start**	longest voiced fricative sub-segment start conveyed as fraction of sound segment (silence, short utt, or long utt)
13	**Longest Voiced Fricative Sub-Segment End**	longest voiced fricative sub-segment end conveyed as fraction of sound segment (silence, short utt, or long utt)
Seven Pitch Parameters [14-20] (short and long utterances)		
14	**# Voiced frames**	Number of voiced frames in sound segment that are subject to having a meaningful pitch value
15	**Pitch Avg (Hz)**	Avg pitch for sound segment
16	**Pitch Max (Hz)**	Max pitch for sound segment
17	**Pitch Min (Hz)**	Min pitch for sound segment
18	**Pitch STD (Hz)**	pitch STD for sound segment
19	**Avg Pitch Confidence**	Average pitch confidence
20	**Pitch Avg-Energy (dB)**	may not be computing dB correctly from power

Seven Hyper-Pitch parameters [21-27]		
21	**# Hyper Frames**	Number of hyper-pitched frames
22	**Hyper Avg (Hz)**	Avg hyper-pitch for sound segment
23	**Hyper Max (Hz)**	Max hyper-pitch for sound segment
24	**Hyper Min (Hz)**	Min hyper-pitch for sound segment
25	**Hyper STD (Hz)**	hyper-pitch STD for sound segment
26	**Avg Pitch Confidence**	Average pitch confidence
27	**Hyper Avg-amp (dB)**	may not be computing dB correctly from power
12 Formant Parameters [28-39]		
28	**FM1 Avg (Hz)**	First formant's average frequency for sound segment
29	**FM1 Max (Hz)**	First formant's max frequency for sound segment
30	**FM1 Min (Hz)**	First formant's min frequency for sound segment
31	**FM1 STD (Hz)**	First formant's STD for sound segment
32	**FM2 Avg (Hz)**	Second formant's average frequency for sound segment
33	**FM2 Max (Hz)**	Second formant's max frequency for sound segment
34	**FM2 Min (Hz)**	Second formant's min frequency for sound segment
35	**FM2 STD (Hz)**	Second formant's STD for sound segment
36	**FM3 Avg (Hz)**	Third formant's average frequency for sound segment
37	**FM3 Max (Hz)**	Third formant's max frequency for sound segment
38	**FM3 Min (Hz)**	Third formant's min frequency for sound segment
39	**FM3 STD (Hz)**	Third formant's STD for sound segment. **Negative 1 when non-existent ?
28 Frequency-Band Amplitude Parameters [40-]		
40	**Energy Avg (dB)**	Energy Avg across all frequencies present
41	**Energy Max (dB)**	Energy Max across all frequencies present
42	**Energy Min (dB)**	Energy Min across all frequencies present
43	**Energy STD (dB)**	Energy STD across all frequencies present
44	**0.5-10kHz Energy Avg (dB)**	Energy Avg across [0.5-10kHz]
45	**0.5-10kHz Energy Max (dB)**	Energy Max across [0.5-10kHz]
46	**0.5-10kHz Energy Min (dB)**	Energy Min across [0.5-10kHz]
47	**0.5-10kHz Energy STD (dB)**	Energy STD across [0.5-10kHz]
48	**0-0.5kHz Energy Avg (dB)**	Energy Avg across [0-0.5kHz]
49	**0-0.5 kHz Energy Max (dB)**	Energy Max across [0-0.5kHz]
50	**0-0.5kHz Energy Min (dB)**	Energy Min across [0-0.5kHz]

51	**0-0.5kHz Energy STD (dB)**	Energy STD across [0-0.5kHz]
52	**0.5-1kHz Energy Avg (dB)**	Energy Avg across [0.5-1kHz]
53	**0.5-1kHz Energy Max (dB)**	Energy Max across [0.5-1kHz]
54	**0.5-1kHz Energy Min (dB)**	Energy Min across [0.5-1kHz]
55	**0.5-1kHz Energy STD (dB)**	Energy STD across [0.5-1kHz]
56	**1-2.5kHz Energy Avg (dB)**	Energy Avg across [1-2.5kHz]
57	**1-2.5 kHz Energy Max (dB)**	Energy Max across [1-2.5kHz]
58	**1-2.5kHz Energy Min (dB)**	Energy Min across [1-2.5kHz]
59	**1-2.5kHz Energy STD (dB)**	Energy STD across [1-2.5kHz]
60	**2.5-5kHz Energy Avg (dB)**	Energy Avg across [2.5-5kHz]
61	**2.5-5kHz Energy Max (dB)**	Energy Max across [2.5-5kHz]
62	**2.5-5kHz Energy Min (dB)**	Energy Min across [2.5-5kHz]
63	**2.5-5kHz Energy STD (dB)**	Energy STD across [2.5-5kHz]
64	**5-10kHz Energy Avg (dB)**	Energy Avg across [5-10kHz]
65	**5-10 kHz Energy Max (dB)**	Energy Max across [5-10kHz]
66	**5-10kHz Energy Min (dB)**	Energy Min across [5-10kHz]
67	**5-10kHz Energy STD (dB)**	Energy STD across [5-10kHz]
15 **Linear Fit/Approximation Data**		
68	**Slope 1**	Slope of the leftmost line.
69	**Slope 2**	Slope of the middle line
70	**Slope 3**	Slope of the rightmost line
71	**Frame Start 1 (Middle Line)**	Starting frame number within the long utterance for the middle line
72	**Frame Start 2 (Right Line)**	Starting frame number within the long utterance for the right line
73	**Y-Intercept 1**	Y-intercept used for the leftmost line (Pitch of Frame 1)
74	**Y-Intercept 2**	Y-intercept used for the middle line (Pitch of Frame Start 1)
75	**Y-Intercept 3**	Y-intercept used for the middle line (Pitch of Frame Start 2)