Continuous monitoring of broiler welfare through audio analytics

M. Rizwan[1], T. Darbonne[2,*] and D. V. Anderson[3]
[1]*Auxtend LLC, Snellville, GA, USA*
[2]*AudioT, Inc., Atlanta, GA, USA*
[3]*Georgia Institute of Technology, Atlanta, GA, USA*
[*]Corresponding author: Tom Darbonne, tom.darbonne@audiot.ai

Abstract

Systems to continuously monitor broiler chicken welfare during grow-out are emerging which rely on sensing and machine learning. Audio analytics, one such system, is a promising complement to video analytics and environmental sensing. Machine learned models developed in academic settings often do not perform well in commercial settings, so there is interest in practical examples of audio analytic use cases developed in commercial settings. This paper describes AudioT INCs audio analytic system developed in commercial grow-out house as part of FFAR's Smart Broiler initiative. A description of the data acquisition, model development, and model results is shared around a practical example of respiratory disease detection and tracking.

Keywords: real-time recognition, audio analysis, spectral analysis, signal processing, poultry welfare

Introduction

New systems are emerging to continuously monitor animal welfare for poultry grow-out. This paper reports on experiences and findings during development of audio analytic methods for monitoring poultry grow-out. Many of the perspectives shared in this paper were refined by our experience in phase one of the Foundation for Food and Agriculture Research (FFAR) Smart Broiler program, which was instituted to define the technologies required to "objectively and comprehensively assess broiler welfare worldwide". With the support of McDonalds, Tyson Foods, and others, a number of teams were funded to apply audio- and video-analytic approaches to report key welfare indicators based on evidence of bird behavior. The first phase of the program sought to demonstrate detected outcomes with the availability of recorded data. The second phase, which is currently active, seeks to implement as much as possible with on-farm edge computing. The collective effort may be transformative to the reporting and management of animal welfare globally.

The poultry industry is highly competitive and is facing increasing pressures related to the responsible sourcing of meat. NGOs are actively and successfully campaigning against the industry's largest suppliers to get them to commit to various animal well-being programs, usually with an implementation deadline of 2024. However, many of the requirements of these programs lack thorough scientific validation and could drastically increase the costs of poultry production. This creates a compelling need in the industry for cost-efficient technologies that can monitor the actual state of the birds to provide the traceability and evidence of well-being needed to satisfy customer demands.

In addition to helping address PR and traceability issues, poultry monitoring also has potential to help farmers become more responsive to their birds' needs and improve their production efficiency. The best farmers tend to be skilled at picking up on various cues from a flock to determine what adjustments need to be made. Notifications from a monitoring system would help facilitate these farmers' efforts and enable others to see some of the same benefits.

Audio analytics have shown the ability to highlight important conditions specific to the age of the bird. This includes distress calling during the first week, activity levels and pleasure/pain indicators during mid-life, and activity levels and disease detection toward late life. This holistic approach is enabled through the development of specific classifiers for specific activities—which may include heat stress, respiratory disease, human intrusions, feed outages, and bird activity levels, as well as equipment and environmental events. While it is not anticipated that audio analytics can replace all other forms of monitoring, audio provides a cost-effective way to monitor the condition of poultry in ways not possible, or not easily replicated, by video, chemical, and other sensor systems.

This paper describes a practical implementation of audio analysis in support of a commercial broiler grow-out house. The example chosen is the development of models to support the detection and tracking of respiratory disease – which highlights the simultaneous monitoring of multiple vocalizations and the significance of multiple zones in tracking events through the house.

Audio acquisition

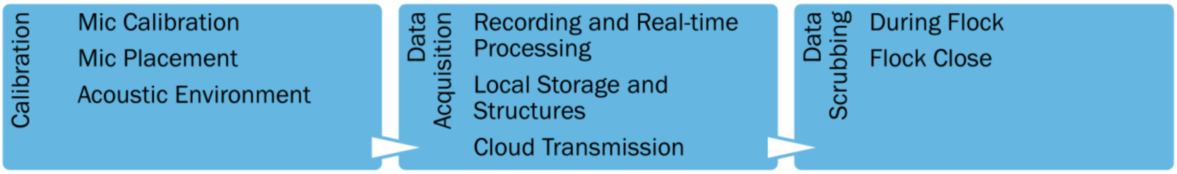


Figure 1: Audio acquisition methods

Figure 1 shows the fundamental steps used to acquire audio suitable for machine learning from a commercial environment. Many of these steps are not unique to broiler grow-out, but some are. Microphones should be closely matched in frequency response and sensitivity to support the generality of models to function as expected. As a broiler house may nominally be over 100 meters long, a single microphone cannot be used to assess the state of animals in the house. AudioT typically deploys 16 microphones to establish 16 "zones" in which animal behavior can be monitored. Placement should be logical to establish low correlation between zones with the exception of loud sounds which permeate more of the house.

It is also useful to characterize the existing environmental systems in isolation by taking measurements before the animals are placed. During growout, the recordings are coordinated by a purpose-built logger that synchronizes the recordings to a real time clock, and stores them in a defined file structure for easy retrieval. Data is periodically uploaded to the cloud for subsequent storage and processing. AudioT uses AWS as a cloud service provider for the benefits not only of storage, but the use of AWS Sagemaker ML development tools.

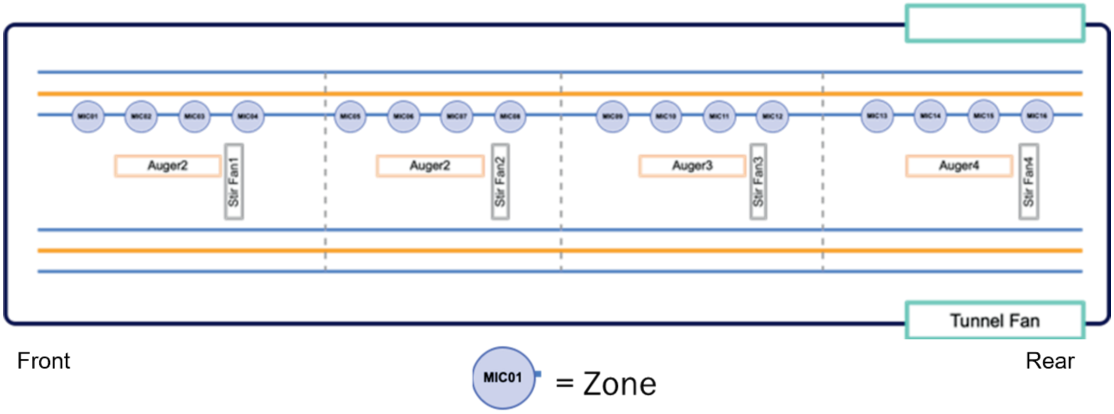


Figure 2: Example of 16 microphones creating 16 Zones within grow-out house for localized monitoring and more spatially precise metrics.

Model development

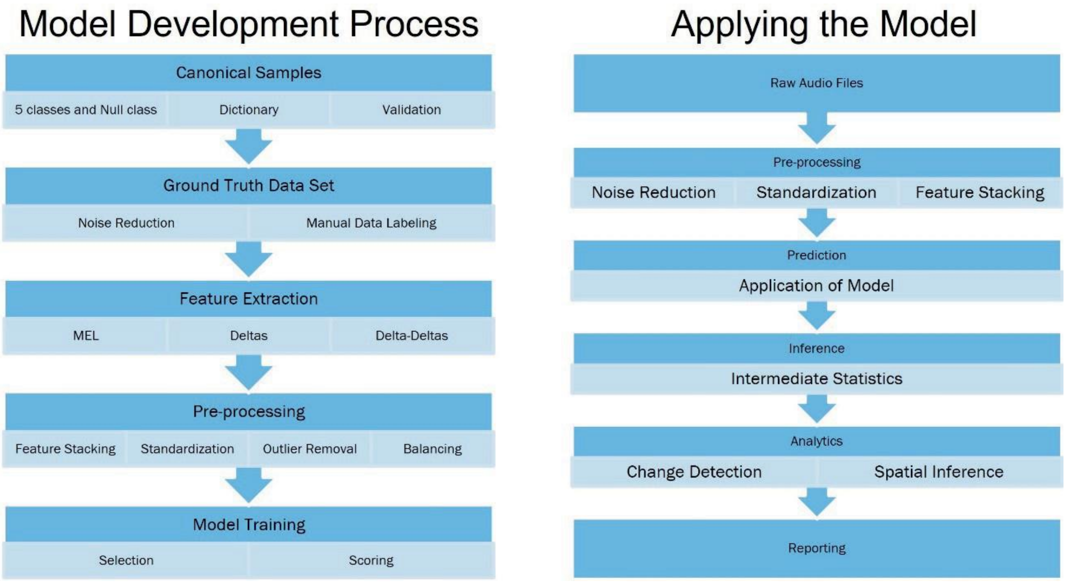


Figure 3: Model Development (left) and Application of the model for inference and reporting (right) The goal of the process is to move from identification of sounds corresponding to disease progression to a multi-classification model which can automatically predict individual instances from raw audio.

Figure 3 shows AudioT's Model Development process particularly as it applies to the complex task of detecting and tracking progression of respiratory disease. Other projects will have analogous steps, but specific dictionary, feature extraction, and pre-processing steps may differ,

Establishing canonical samples

Consistent labeling is critical to model performance, and difficult to achieve. There are not many poultry vocalization experts, so it is important to create a dictionary of vocalizations which can be used by a team for labeling, and then to cross-check the accuracy of modeled output against the dictionary templates. The first task was to characterize changes in vocalizations as the disease progressed through individual animals, and to establish 5 canonical audio signatures corresponding to the progression. The five signatures ranged from the initial "snick" [class 1] to the final "gasp" [class 5]. All non-targets were categorized as noise [class 0].

From the first two days of active infection, 100 randomly selected raw audio files (each one minute in duration) were selected from a single microphone. These files were graded and down selected for the lowest background noise. A small team used conventional audio analytic techniques (visual analysis of spectrograms and listening) to individually mark vocalization patterns of interest in the audio files, develop consensus on label names and structure, and compile a small dictionary of representative examples of each. The dictionary was validated by domain experts: poultry veterinarians and farm managers. The resulting dictionary was detailed enough to allow data labelers to differentiate "coughs" (technically chickens do not cough, but this is a common descriptor) as type 1, 2, 3, 4, 5 corresponding to the disease progression.

Ground truth

A larger set of files was down-selected from three days where the infection was active, and noise reduction algorithms which were effective at removing the relative stationary noise of environmental control and feed systems was applied, which left the vocalizations intact. 117 files were split between the team members for manual labeling using audio analytic techniques, and a final audit by team members to ensure consistent practice. The result was the initial ground truth data for model development. While ideally this ground truth would remain unaltered, in practice subsequent model development steps will highlight data which has been mislabeled – a form of secondary audit – and corrections will be made to the ground truth data during the model development phase. The files are divided into training and validation sets.

Feature extraction

Training files were filtered to the range of 100- to 4000-Hz, and then mapped by a "feature extraction" step to 10 Log-MEL features. As a check, these features were plotted and the mean of each feature was identified, within a range or -3 to -4.3.

More detailed features would be needed to differentiate all five classes, thus Deltas were obtained by taking the first derivative of the Log-MEL, and Delta-Deltas their second derivative. Understanding the speed and direction of energy changes added the missing information. Scaling was applied to log-MEL features, then Deltas, then Delta-Deltas to bring them more in conformance which aided some models.

Preprocessing

Standarization. To generalize the model to apply to other microphones, a standardization step was added after feature extraction. *Normalization* typically rescales the values into range [0,1]. *Standardization* typically rescales data to have a mean of 0 and a standard deviation of 1. After standardization, each feature column will have a mean of 0 and the standard deviation of 1 which eliminates realistic scale differences. Standardization accomplished with the following simple formula: mu (μ) is the mean and sigma (σ) is the standard deviation of the values.

$$z = \frac{\$\mathrm{RS}}{\mathrm{T}}$$

where $\mu = \text{Mean}$ and $\sigma = \text{Standard Deviation}$

Stacking multiple audio frames. Multiple "narrow" frames need to be concatenated to cover single instances of some vocalizations. "Stacking" enhances model performance. The trade-off is more complexity. 3

contiguous frames are stacked into one, so instead of 30 features in single 50 millisecond frame, 90 features are in each stacked 150 millisecond frame.

Remove abnormal outliers from background using random cut forest (RCF) algorithm. Many of the target vocalizations were being missed by an early model, so a random cut forest (RCF) model was added to the pre-processing chain to increase the number of true positive vocalizations captured. Non-coughs were categorized as background noise. RCF was a non-supervised algorithm chosen for convenience because it was a built-in feature of AWS Sagemaker which provided an anomaly score for each data point. The concept behind the RCF is to use random cut (on dimensions/ features) to separate groups of data points (or trees) from a sampled set of data points (called bounding box) [5]. Each separated group becomes the next bounding box and continues to be separated until all trees are isolated. The closer the points (distance measured as the number of cuts) to the original data set, the higher its anomaly score. A final score is calculated as the average across scores. Outliers were removed by thresholding using the standard deviation from the mean for the anomaly score to separate the background noise (using a value of 1).

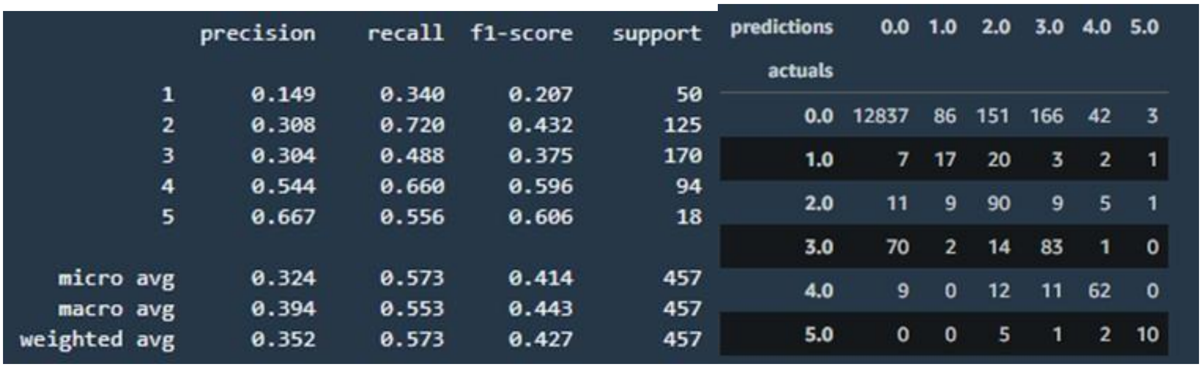
Balancing. The nature of the data creates a very unbalanced data set, with the null case ("0") being the majority class by a couple more orders of magnitude. Under sampling was performed on class 0.

Model training

Several algorithms including K-nearest neighbor (KNN), Neural Network (NN), and XGboost were investigated. After several steps of data pre-processing and tuning, finally XGboost model was the best performer. The weighted average F1-score achieved more than 0.4 for 5 classes of coughs.

Table 1: Summary of weighted average F-1 score for XGboost model

	no tuning no Standardization		with tuning no Standardization		with tuning with Standardization	
Models	Original audio	Noise reduced	Original audio	Noise reduced	Original audio	Noise reduced
Xgboost	0.403	0.418	0.445	0.440	0.399	0.427



```
              precision    recall  f1-score   support

           1      0.149     0.340     0.207        50
           2      0.308     0.720     0.432       125
           3      0.304     0.488     0.375       170
           4      0.544     0.660     0.596        94
           5      0.667     0.556     0.606        18

   micro avg      0.324     0.573     0.414       457
   macro avg      0.394     0.553     0.443       457
weighted avg      0.352     0.573     0.427       457
```

predictions actuals	0.0	1.0	2.0	3.0	4.0	5.0
0.0	12837	86	151	166	42	3
1.0	7	17	20	3	2	1
2.0	11	9	90	9	5	1
3.0	70	2	14	83	1	0
4.0	9	0	12	11	62	0
5.0	0	0	5	1	2	10

Figure 4: Individual f1-score for XGboost model with hyperparameter tuning (on noise-reduced audio with standardization) [left]. Confusion matrix [right].

Adding a standardization step slightly decreased the F-1 score from 0.440 to 0.427, even it is not the best performer; however, the standardization is necessary for applying the model into new data prediction from other microphones. Therefore, this is the best model (F-1 = 0.427) we selected to use practically for all the prediction.

Applying the model

Referring to Figure 3, the basic operation steps of inference and reporting are described below.

Raw data and pre-processing

The pool of raw audio data to be processed includes all of the microphone in the house. In this instance eight microphone were used. While, all of the modeling work was conducted using data from a single microphone, the pre-processing noise reduction and standardization steps prescribed earlier have the second benefit of allowing the resultant model to adequately predict results on the remaining microphones in the house. Feature stacking is also applied to enable the differentiation of all 5 vocalization categories.

Prediction

Applying the model to the pre-processed data is straight-forward as AWS platform has the requisite data storage and data processing capabilities. Scripts and Lamdba functions invoke computational resources, funnel required data to processing from S3 or EFS data storage, apply the model, store results in S3, and tear down/close the processing resources. The results are predicted labels for all five classes and simple statistics that count the number instances of each one-minute file.

Inference

Intermediate Statistics: Daily total cough counts detected from different microphones and hourly counts for 5 stages of coughs in each microphone were plotted. Log of cough counts served as a LT outbreak degree measurement and generated heatmaps with both hour and date axis for different microphones. Summarized prediction results were saved as csv files and loaded into Sagemaker Jupyter notebook as pandas dataframes.

Results and discussion

While there are several useful outputs from this effort, two are particularly illustrative.

Change Detection: A practical objective was to provide the farm manager with tools to detect the onset of disease earlier. Level 1 coughs can be confused with normal vocalizations, and the middle stages will need to be caught as early as possible, the model can be customized to a desired threshold for each type of cough and alert the manager when they have been reached.

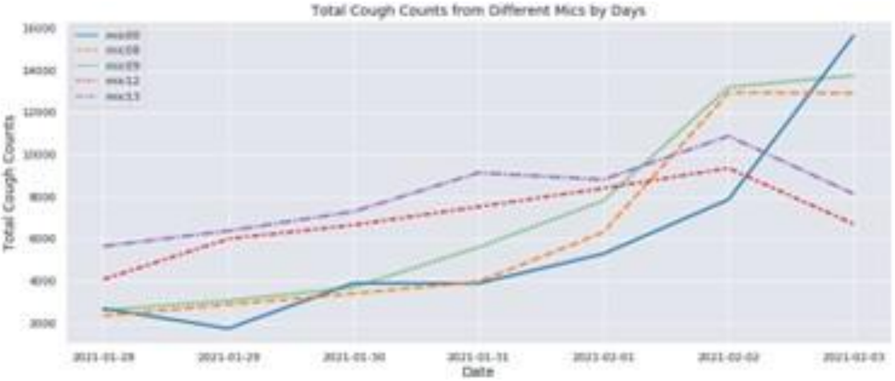


Figure 5: Total "cough" counts from different microphones by day.

Spatial Analysis: The value of multiple microphones in the grow-out house is that conditions can be monitored to determine their origin and study changes at various locations in the house. By way of example, if all the coughs at each mic were tallied on hourly basis (Figure 5), the impression is that the LT outbreak is getting worse. The cough counts trend from all microphones were increasing [note: 2021-02-03 declines because de-population happened on that day]. The purple line (mic 13 at rear of house) is consistently higher than the blue line (mic 0 at the front of the house).

While there is no obvious breakthrough point observed from Figure 5, the more detailed daily plots provide the detail (Figures 6a and 6b). The rear of the house has much more activity throughout the entire day, and the rate of infection is advanced. New infections (the blue line) are still advancing, so the disease cycle remains very active.

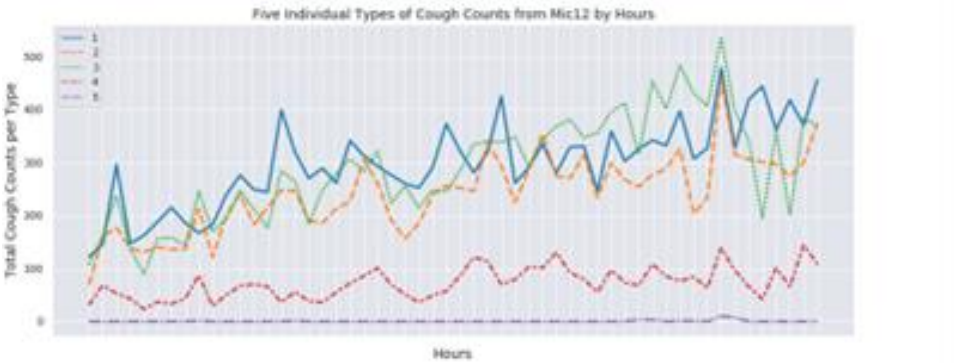


Figure 6a: Rear of house (mic 12)

By comparison, the front of the house is relatively quiet until the evening. Infection is still not at the level of the rear of the house (Y-axis scales are different), but clearly the disease has taken hold.

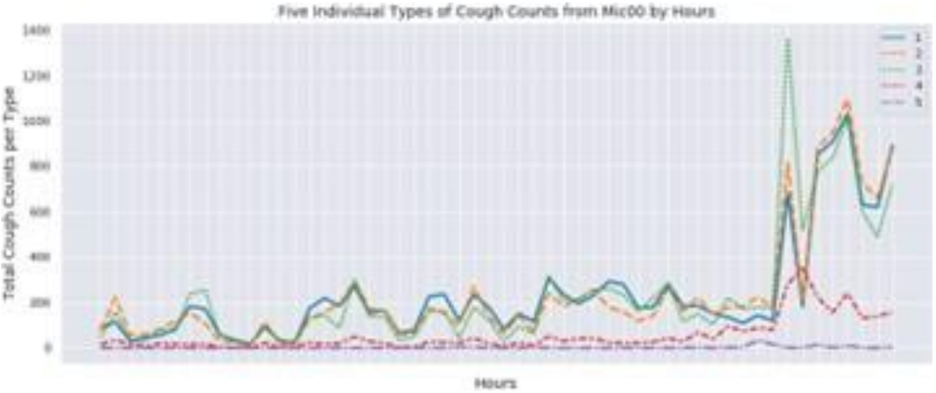


Figure 6b: Front of house (Mic 00) Figure 6a: Front of house (Mic00)

The spatiotemporal characteristics, extracted by generating the daily information and tracking the severity and spread, have implications for bio-security and operational planning. Knowing where the disease entered the house helps reduce repeat events. Early detection may result in greater medical intervention options or planning options.

Conclusions

Audio Analytics will prove a cost effective and useful tool for continuously monitoring broiler welfare during grow-out. This paper reviewed the practical elements of a system from audio acquisition, and machine learned model development. The example of tracking respiratory disease highlighted some of the considerations and challenges of model development.

Acknowledgments

This project was funded by the Foundation for Food and Agricultural Research (FFAR), and the Georgia Research Alliance (GRA). This material is based upon work supported by the National Science Foundation under Grant No. 1919235.

References

Virtanen, T., Plumbley, M., AND Ellis, D. (2017) Computational Analysis of Sound Scenes and Events.
Springer. Dinev, I. Diseases of Poultry. *https://www.thepoultrysite.com/publications/diseases-of-poultry/*
AWS Sagemaker Documentation - How random cut forest works:
https://docs.aws.amazon.com/sagemaker/latest/dg/rcf_how-it-works.html
XGBoost Documentation - *https://xgboost.readthedocs.io/en/latest/parameter.html*