

# Department of Computer Science and Engineering

Summer Research Internship

## PROJECT REPORT

Classification of Chicken Sound Using Deep Learning

Using TensorFlow, Librosa, and Comparative Analysis of ResNet and Inception Models

## Submitted By:

Darpan Gupta, Ishita Gupta, Prerna Kewat

Date: 14-07-2023

## Submitted To:

Dr. Neeraj Goel | Assistant Professor - IIT Ropar, Punjab

# 1. Abstract

The project aims at developing a deep learning model to recognize and classify various types of chicken sounds, enabling valuable insights into their emotional states and behaviors. This can facilitate automated monitoring of poultry farms. To accomplish this, a thorough visit to a poultry farm was conducted and multiple sound samples of chickens were collected. The dataset was cleaned and pre-processed and labelled the samples into four distinct categories: alarm calls, egg-laying sounds, heat sounds, and feeding sounds.

For the model training process, the data was split into an 80:20 ratio for training and testing respectively. To analyze the sound samples, Librosa library in Python was employed for generating Log Mel spectrograms for each sample.

For deep learning techniques, TensorFlow framework was used to train the CNN model. Additionally, a comparison of the performance of different deep learning models such as ResNet and Inception was conducted. ResNet50 provided an accuracy of 78.91% and InceptionV3 provided an accuracy of 93.13%. Whereas our self-made model provided the best results with an accuracy of 97%. The comparative analysis allowed us to evaluate and compare the results, contributing to the overall success of the project.

# 2. Introduction

Like human speech, animal vocalizations simultaneously provide others with information that is both semantic and emotional. By understanding how animals communicate it can provide valuable insights into their behaviors. Our research is based on studying vocalizations of chickens and the sounds they produce in different circumstances. The project relates to understanding chicken vocalizations and sets the groundwork for future studies in poultry sound analysis and classification.

In various circumstances, such as sensing danger, experiencing hunger, or during the process of laying eggs, chickens exhibit a range of behaviors and emit distinct sounds. The egg production in India has increased from around 83 billion nos. in 2015-16 to around 88 billion in 2016-17 registering a growth of about 6%. The per capita availability of egg has increased from 61 in 2013-14 to 66 in 2015-16. In 2016-17 it is 69.

This has led to an increasing focus on the application of deep learning and machine learning techniques in the classification of chicken sounds. Chickens primarily communicate through vocalizations and body language. Vocalizations consist of sounds such as "cluck" or "bak-bak-bak," as well as high-pitched sounds like "pcaack." Meanwhile, their body language encompasses behaviors like wing flapping, head throbbing, and beak pecking, among others.

When chickens are in a state of calm, they emit a distinctive natural vocalization characterized by a repeated "bak-bak-bak" sound. Conversely, when experiencing distress, they produce vocalizations of higher frequency. By closely monitoring these vocalizations, valuable insights can be gained to enhance the efficiency of poultry farming practices. Moreover, such monitoring enables farmers to gain a deeper understanding of the factors contributing to these distress vocalizations.

# 3. Materials and Methods

## 3.1. Experimental Setup

The data used in this experiment were collected from a local hatchery located in Bainsa, Shaheed Bhagat Singh Nagar, Punjab. It was recorded on 21st of June 2023. About 500 chickens were kept stacked in cages. Audio collection devices were placed approximately 1 meter above the ground and 0.5 meter away from the chickens. Three devices were used for collecting the audio samples. Along with audio, videos were also collected during this process for easing the process of audio labelling. All audio recordings were sampled at 22.05 kHz with a 16-bit resolution. Dataset collection took place on a sunny day with temperature around 35°C.

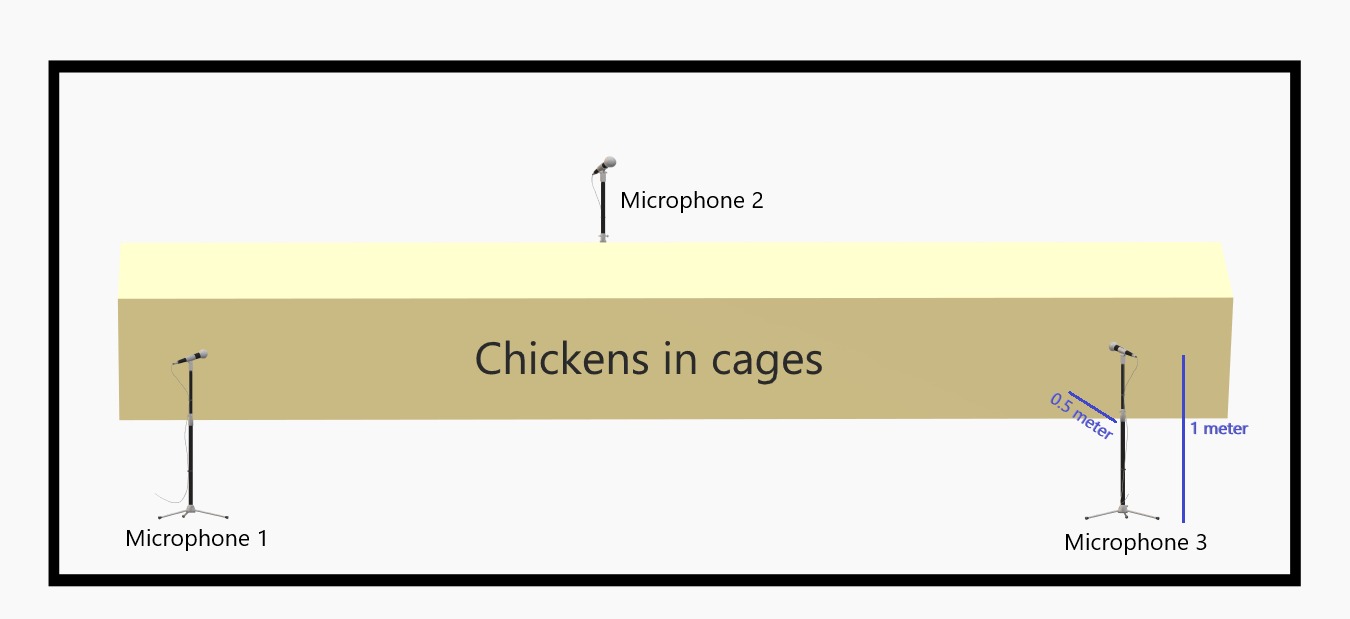


Figure 3.1. (a) Experimental setup



Figure 3.1. (b) Tripod placement



Figure 3.1. (c) Chickens housed in individual cages

## 3.2. Dataset Collection

To capture each type of sound, we followed a systematic recording process. Upon entering the room, we noticed that the chickens exhibited a state of alertness and emitted distinct sounds which are referred to as their alert calls/alarm calls. Figure 1 represents the spectrogram corresponding to these alert calls.

To record the specific feeding sounds, we established a feeding schedule, with the first feeding session taking place at 9 am and the subsequent session at 4 pm. We recorded the characteristic "tak-tak-tak" sounds produced by the chickens while feeding. Figure 2 displays the spectrogram representing these feeding sounds.

To capture the sounds of distress caused by heat, we deliberately closed the fans. During this time, we observed the chickens breathing with their mouths open, as a response to the elevated temperature in the room. Consequently, we collected and labeled these distress sounds as heat distress sounds. The corresponding spectrogram of heat sounds is depicted below in Figure 3.

The sound produced during egg laying has a peculiar "bak bak bak pcaack" sound. It possesses a unique quality that sets it apart, making it easily recognizable to anyone who hears it. Consequently, we made sure to record the egg laying sound to capture its distinctiveness accurately. Figure 4 displays the spectrogram representing these feeding sounds.

## 3.3. Data Annotation

Data annotation or data labeling was conducted using Audacity software. The audio samples collected were divided into 1.5-second segments for annotation purposes. This segment length was chosen to ensure that each segment could effectively capture distinct characteristics of various sound types. Annotating each segment involved assigning one of 4 sound types, specifically determined to meet the classification requirements of the project. Manual annotation was performed by carefully listening to the audio samples. In addition to audio recordings, video inputs and Mel spectrograms were utilized to classify chicken voices. Video data provided supplementary information regarding the chicken's body language and facial expressions, thereby enhancing the classification accuracy. Mel spectrograms, which visualize the frequency content and temporal dynamics of audio, were employed to extract relevant features for classification purposes.

The data annotation process encountered a challenge in dealing with noise present in the audio data. Noise refers to unwanted sounds that could potentially interfere with the desired chicken sounds, consequently hindering accurate classification. Examples of noise include sounds such as "fan noise," "workers talking," "sounds of other animals," and "sounds produced when chickens interfere with cages."

To mitigate the impact of noise, filtering techniques were implemented. Low-pass, high-pass, or band-pass filters were selectively applied to eliminate undesired frequencies likely to be associated with noise. These filters played a crucial role in enhancing the clarity and distinguishability of the chicken sounds. Overall, the data annotation process proved to be a challenging yet vital step in this project.

## 3.4. Input Preparation

For data preprocessing, we transform the audio signals into log-Mel spectrograms, which provide a visual representation of sound as time-frequency images. This representation is suitable for CNNs, as they excel at learning patterns and features from image-like inputs and not audio input. The log-Mel spectrogram captures important acoustic characteristics of the audio signals and enables the CNN to effectively discriminate between different sounds.

 Firstly, the Short-Time Fourier Transform (STFT) is applied to the audio waveforms using a Hann window function. The STFT allows us to analyze the frequency content of the audio signal over short, overlapping time segments. By using a window function like Hann, we ensure that the edges of each window taper smoothly to reduce artifacts in the resulting spectrogram. This helps prevent a sudden spike in the audio. The choice of a window length of 2,048 points and a hop length of 256 points was based on the trade-off between temporal and frequency resolution.

From the STFT, we extract the magnitude spectrum, which represents the amplitude of different frequency components over time. The magnitude spectrum provides information about the energy distribution in the audio signal and is a common representation for spectrogram computation.

STFT with a hop length of 256 points results in overlapping frames. In this case, 130 temporal frames are generated for each audio sample. Overlapping frames help capture temporal dynamics in the audio, allowing CNN to learn patterns and changes over time.

convert the magnitude spectrum into a log-Mel spectrogram, we apply a set of 128 log-Mel scale band filters. These filters divide the frequency range into distinct bands, following the perceptual characteristics of human hearing. The logarithmic scale emphasizes lower frequencies, which are more discriminative for many audio analysis tasks.

By applying the log-Mel scale band filters to the magnitude spectrum, we obtain a log-Mel spectrogram with dimensions of 128 (frequency bins) by 130 (time frames). This spectrogram represents a two-dimensional image-like input that can be fed into a CNN. The vertical axis corresponds to the spectrum of frequencies (in Hz), while the horizontal axis represents time (in seconds). The intensity of each pixel in the spectrogram image represents the amplitude (in dB) of the chicken call at that specific time-frequency point.

After cleaning and pre-processing the dataset, we labelled the samples into four distinct categories: alarm calls (722 samples), egg-laying sounds (336 samples), heat sounds (629 samples), and feeding sounds (415 samples). For the model training process, we split the data into an 80:20 ratio, with 80% allocated for training (1680 samples) and 20% for testing (422 samples).

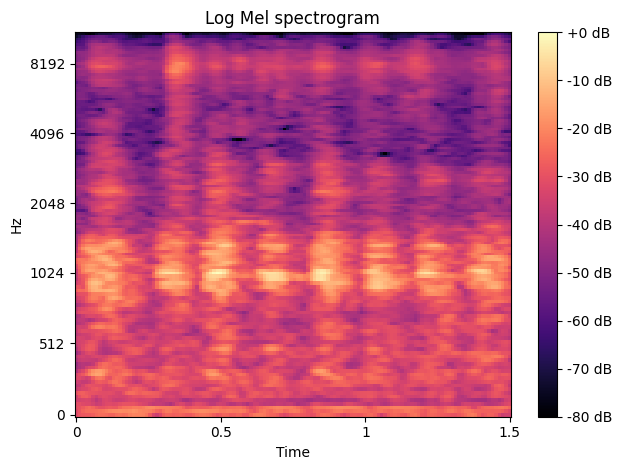


Figure 4.4. (a) Alert calls spectrogram

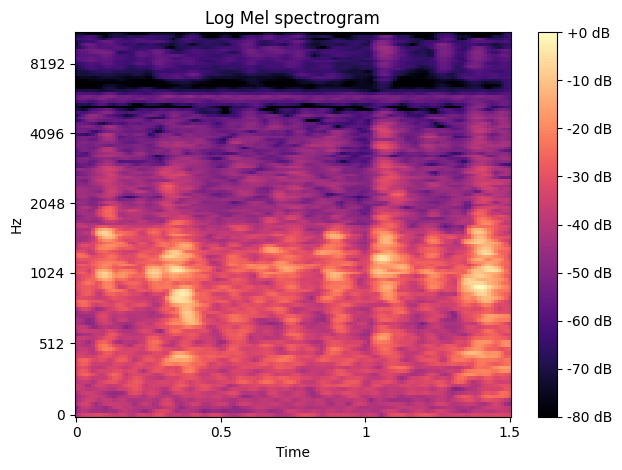


Figure 4.4. (b) Feeding sounds spectrogram

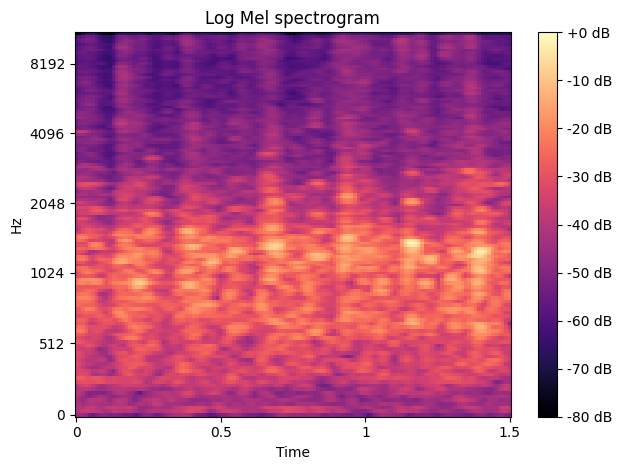


Figure 4.4. (c) Heat distress sounds spectrogram

Figure 4
Figure 4

Figure 4.4. (d) Egg laying sound spectrogram

3.5. Model Architecture

The project adopted two types of popular CNN models, namely ResNet50 and InceptionV3, and built a custom model as well. The default input image size for ResNet50 and InceptionV3 models is (224, 224, 3) and (299, 299, 3) respectively. However, the log-Mel spectrograms used have an image size of (128, 130, 1). Also, these models have thousands of classifications of images whereas the project has four classifications. After modifying input and output shapes of these models, the total number of trainable parameters in ResNet50 was 24,053,892 and in InceptionV3 was 22,293,348.

However, these models are complex and possess a challenge to computation efficiency due to their structure and a large number of parameters. For instance, ResNet50 has 50 deep neural network layers and InceptionV3 has 96 deep neural network layers. Thus, a custom CNN model was created with two convolutional layers. The first convolutional layer has 32 filters of size 3x3, and the second convolutional layer has 64 filters of size 3x3. The output of each convolutional layer is then passed through a max pooling layer to reduce the size of the feature maps. The flattened output of the max pooling layers is then passed through two dense layers with 64 and 4 neurons respectively.

The final layer uses a SoftMax activation function to output the probability of each class. The optimizer used is Adam, which is a popular optimizer for training deep learning models. The loss function used is categorical cross entropy, which is the standard loss function for classification problems. This model has significantly fewer parameters, approximately 3,828,420, and faster detection speed compared to the complex ResNet50 and InceptionV3 models.

# 4. Discussion and Conclusion

Chickens exhibit various sounds to convey their physical and emotional states. Through the utilization of a deep learning model, these sounds were successfully detected and classified into four distinct classifications:

Alarm sounds: These sounds occur when chickens experience high levels of alarm, such as during attacks or moments of immediate stress and fear.

Feeding sounds: These sounds were recorded during the chickens' feeding times and typically consisted of low-frequency eating sounds, including pecking sounds.

Egg laying sounds: These are the well-known "bak bak bak pcaack" sounds produced by hens after laying their eggs. These high-frequency sounds have a distinctive pattern, with equally spaced and repetitive intervals. They are commonly referred to as the "egg song."

Distress sounds: General distress caused by factors like heat, the presence of strangers, or changes in the environment were categorized as distress sounds.

Through the deep learning model, accurate classification of these chicken sounds was achieved, enabling a better understanding of their behaviour and emotional states.

As the industry expands, we also need to keep animal welfare in mind. Distressed chickens also impact the production and yield of the quality of eggs.

We utilized deep learning classification methods to automatically identify chicken distress calls using audio recordings of chickens.

# 5. Results

The project has implemented the network architectures of ResNet 50, Inception V3, and a self-made model. Using these three different deep learning CNN models, comparison and analysis of the results and accuracy of each one was done. Models were trained on the log-Mel spectrograms converted from raw audio signals. Out of the three, the self-made model has seemingly performed the best and given relatively better results with 99% accuracy in training and 97% accuracy on test data. Inception V3 performed the second best with 98.33% accuracy in training and 93.13% accuracy on test data. ResNet 50 performed with 96.49% accuracy in training and 78.91% accuracy on test data.

The self-made model outperformed the ResNet50 and Inception V3 because these models can be difficult to train on small datasets. This is due to their large number of parameters, and it requires a lot of data to learn these parameters. If the dataset is too small, the model may not learn the parameters accurately, and it may not generalize well to new data.

Also, they can be computationally expensive to train on small datasets. This is because they have a large number of operations, and it requires a lot of computing power to train the model. If the dataset is too small, the training process may be too slow or too expensive.

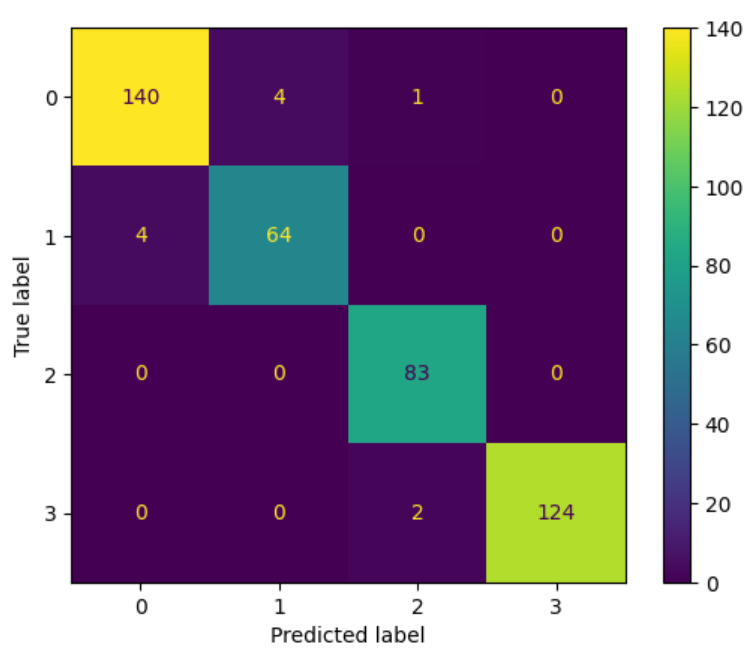


Figure 5. (a) Confusion Matrix for the Self-Made Model

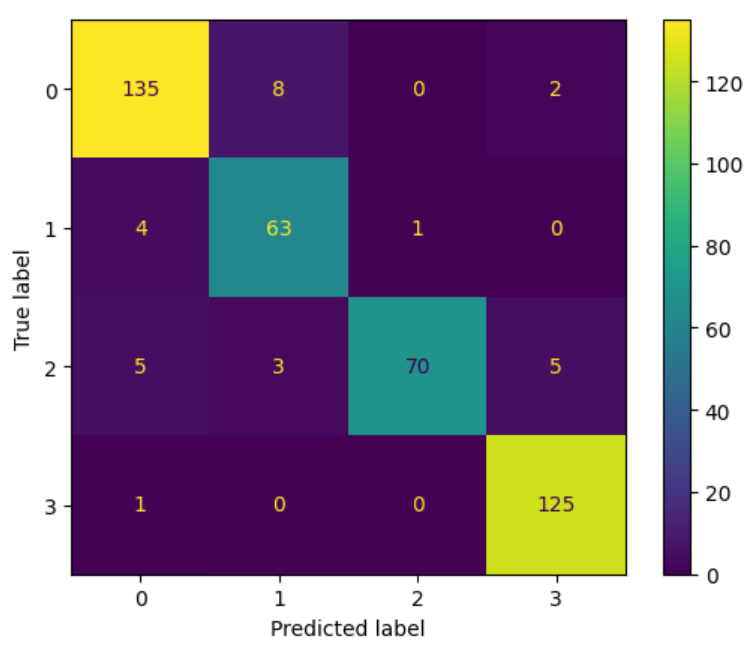


Figure 5. (b) Confusion Matrix for the Inception V3 Model

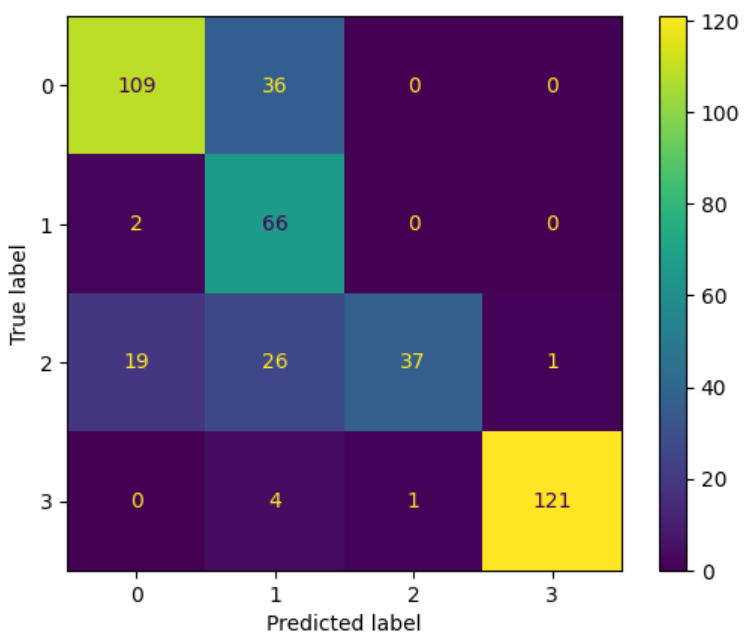


Figure 5. (c) Confusion Matrix for the ResNet 50 Model

Table 1. Performance Metrics of ResNet 50

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | PRECISION | RECALL | F1 SCORE | SUPPORT |
| Alarm  Sounds | 0.84 | 0.75 | 0.79 | 145 |
| Egg  Sounds | 0.5010:56 AM | 0.97 | 0.66 | 68 |
| Feeding  Sounds | 0.97 | 0.45 | 0.61 | 83 |
| Heat  Sounds | 0.99 | 0.96 | 0.98 | 126 |

Table 2. Performance Metrics of Inception V3 Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | PRECISION | RECALL | F1 SCORE | SUPPORT |
| Alarm  Sounds | 0.93 | 0.93 | 0.93 | 145 |
| Egg  Sounds | 0.85 | 0.93 | 0.89 | 68 |
| Feeding  Sounds | 0.99 | 0.84 | 0.91 | 83 |
| Heat  Sounds | 0.95 | 0.99 | 0.97 | 126 |

Table 3. Performance Metrics of Self-Made Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | PRECISION | RECALL | F1 SCORE | SUPPORT |
| Alarm  Sounds | 0.97 | 0.97 | 0.97 | 145 |
| Egg  Sounds | 0.9410:56 AM | 0.94 | 0.94 | 68 |
| Feeding  Sounds | 0.97 | 1.00 | 0.98 | 83 |
| Heat  Sounds | 1.00 | 0.98 | 0.99 | 126 |

Table 4. Comparison of Performance Metrics for Different Models

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| MODEL |  | PRECISION | RECALL | F1 SCORE | SUPPRORT | ACCURACY | PARAMETERS |
| RESNET50 | Micro  Average | 0.83 | 0.78 | 0.76 | 422 | 0.79 | 22,293,348 |
| Weighted  Average | 0.86 | 0.79 | 0.79 | 422 |
| INCEPTION V3 | Micro  Average | 0.93 | 0.92 | 0.92 | 422 | 0.93 | 24,053,892 |
| Weighted  Average | 0.93 | 0.93 | 0.93 | 422 |
| SELF-MADE | Micro  Average | 0.97 | 0.97 | 0.97 | 422 | 0.97 | 3,828,420 |
| Weighted  Average | 0.97 | 0.97 | 0.97 | 422 |

# 6. Future Additions

To enhance the project's effectiveness, an expanded data collection approach would have been preferred. Ideally, gathering audio samples from various breeds of chickens, encompassing both broiler chickens and egg-laying hens, would have been valuable. Additionally, recording sounds over extended periods, such as on a day-to-day basis for more than a week, across different seasons, could have significantly enriched our dataset. This augmentation would have allowed our model to accommodate diverse breeds, incorporate a wider range of sound classifications, and even recognize distinct seasonal variations in the sounds.

A pivotal aspect of our model lies in its potential to assist farmers by detecting alarm and distress signals, alerting them when the chickens are in jeopardy or experiencing discomfort. For example, during situations involving potential threats like animal attacks or intrusion by predators.

The application of using speech recognition to identify respiratory diseases in chickens shows promise. Integrating all these classifications into the model would enhance its effectiveness and create a potential impact on poultry farming and beyond. This solution not only improves animal health but also encourages sustainable and ethical practices in the agricultural industry, creating a positive influence on the welfare of chickens and farmers alike.