A Decoder of Cats' Vocalizations

Haoheng Tang, Joanna Shen, Ziqian Liao

1. Motivation

In this project, we study the following problem: Is it possible to use cat sounds to classify their emotional states? Pet owners frequently encounter challenges in comprehending the needs and emotions of their beloved animals, making the task of ensuring their well beings hard. This is particularly true in cats. Different cats could use completely different sounds to express the same meaning. Since cats' vocalizations can vary widely and carry nuanced meanings that are often lost in communications, we want to use AI to allow people to better understand and take care of their cats.

2. Methodology

Hypothesis

A cat's emotions, such as pleasure or anger, can be directly discerned from its vocalizations- "meow". In other words, without observing its behavior or expression, an AI is able to tell a cat's emotions from its vocalizations.

Goal

We aim at training an AI to accurately classify a cat's vocalizations. Given an audio input, the AI should output one of the following labels that best describes the cat's emotion: 1.happiness, 2.fighting, 3.defensive behavior, 4.anger, 5.warning, 6.resting, 7.pain, 8.calling for its mother, 9.mating, and 10.hunting desire.

Description of the Data and Data Handling

Overview of the dataset

We used a dataset of cat sounds curated for better understanding of cat behavior, or more specifically, the emotion classification of cats. This dataset is organized to include a total of 10 categories (Figure 1), each representing a different emotional state of a cat: happiness, fighting, defensive behavior, anger, warning, resting, pain, calling for its mother, mating, and hunting desire, which will be used as class labels for the classifier.

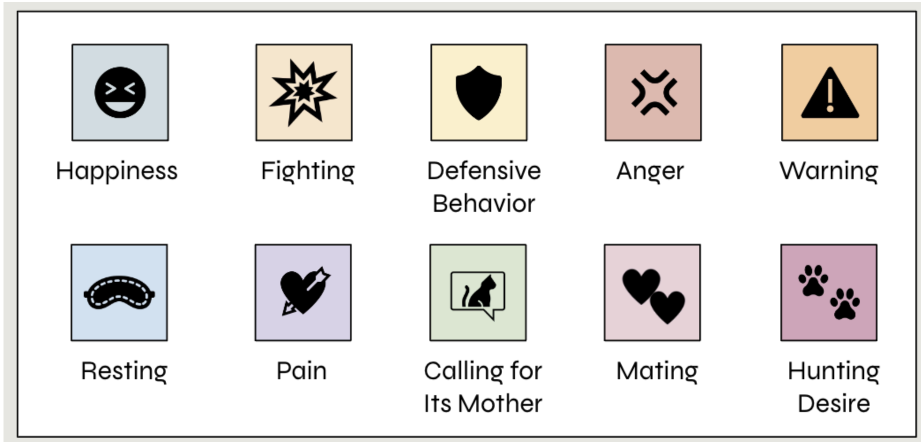

Figure 1. Labels of Cats' Vocalizations

Here list the 10 labels included in our dataset, which also are the emotional states our AI model trying to predict given the input data.

We have listed our data source as follows:

- Yagya Raj Pandeya, Dongwhoon Kim and Joonwhoan Lee, Domestic Cat Sound Classification Using Learned Features from Deep Neural Nets (https://www.mdpi.com/2076-3417/8/10/1949)
- Yagya Raj Pandeya and Joonwhoan Lee, Domestic Cat Sound Classification Using Transfer Learning (http://www.ijfis.org/journal/download_pdf.php?doi=10.5391/IJFIS.2018.18.2.154)

Data Handling

During preprocessing, we have transformed the audio in the form of .mp3 files to .wav files to comply with the python packages. We also converted the stereo input (2 channels) into mono (1 channel). There is no missingness in our dataset.

Exploratory Data Analysis

Our dataset consists of a relatively large number of audio samples, totaling 5,922 individual files. This ample sample size provides a solid foundation for training and evaluating AI models aimed at classifying cat emotions based on vocalizations.

The durations of the audio files vary significantly, ranging from as short as 0.313 seconds to as long as 16.797 seconds. This variability reflects the natural differences in cat vocalizations and adds complexity to the task of emotion classification. Moreover, the distributions of audio lengths differ across the 10 emotional classes. Histograms (Figure 2) illustrating these distributions show that certain emotions are more likely to be expressed through shorter or longer vocalizations, indicating potential patterns worth investigating further.

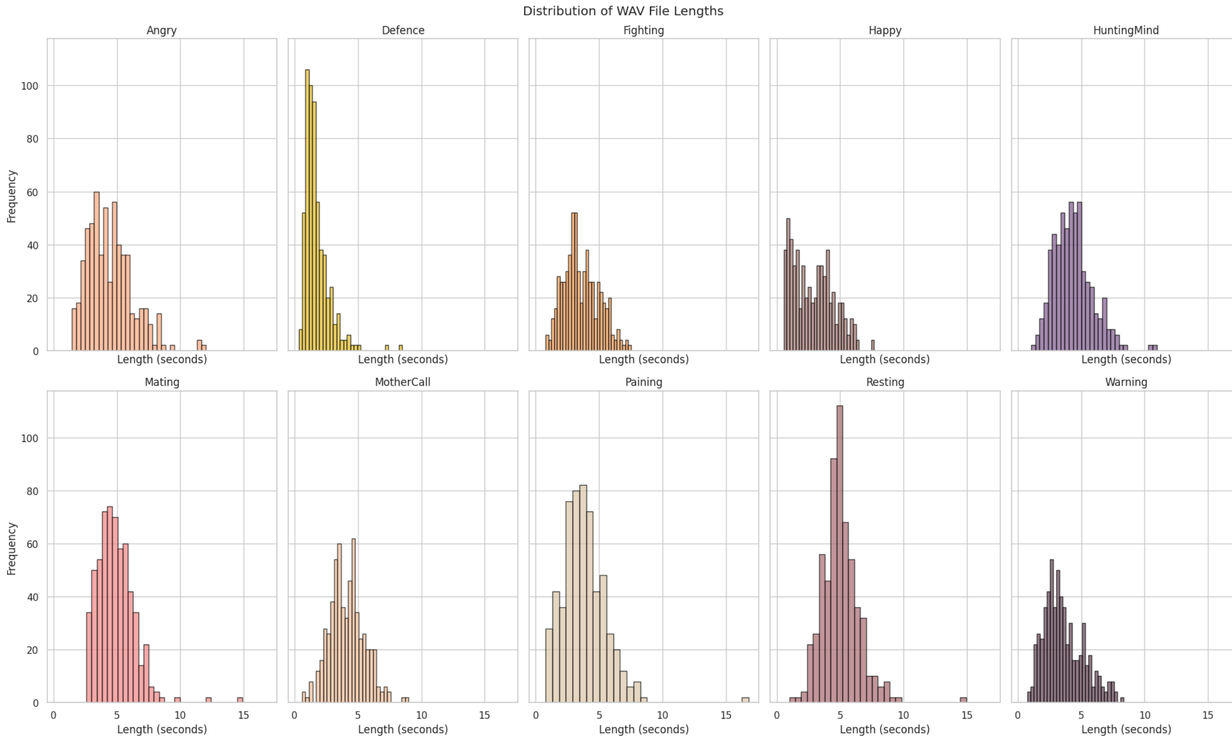

Figure 2. Distribution of WAV File Length

An important characteristic of the dataset is its class balance. The 10 emotional categories are evenly represented, ensuring that the classification model is not biased toward any particular emotion. This balance is visually confirmed through the following bar graph (Figure 3), which shows roughly equal counts for each class.

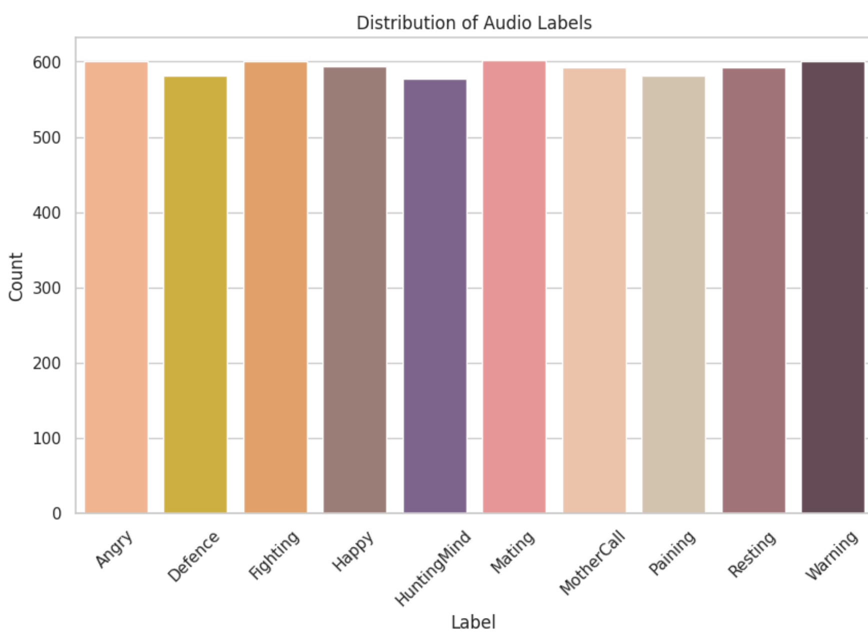

Figure 3. Distribution of Audio Labels

However, one of the key challenges revealed by our analysis is the significant variation within each emotion class, coupled with relatively small differences between some classes. This is especially evident in the spectrograms (Figure 4). For example, the first two spectrograms, though visually distinct, both represent vocalizations associated with happiness. In contrast, the last spectrogram, which closely resembles one of the happy vocalizations, actually corresponds to a cat in pain. This suggests that while there is diversity in how a single emotion may be expressed, similarities across different emotions may complicate classification, highlighting the nuanced nature of feline vocal expressions.

Modeling Approach

Given the task of classifying the emotional state based on cat sound, we decided on combining an audio feature extractor and a Convolutional Neural Network (CNN) classifier to build our model.

Audio signal processing commonly uses a set of audio features to represent the original audio effectively, and this is the traditional approach in transforming waveform into a matrix that is more compatible with machine learning models.

Our approach included the calculations of:

- *Zero Crossing Rate*: Measures the rate at which audio signal changes from positive to negative or vice versa, implying the smoothness of the sound.
- *Chroma Frequencies*: Measures the harmonic content in the sound.
- *Mel-Frequency Cepstral Coefficients (MFCCs)*: Captures the timbre and texture of the audio.
- *Spectral Rolloff*: Measures the distribution of total energy, implying the brightness of a sound.
- *Spectral Bandwidth*: Quantifies the width of the band of frequencies that contain most of the energy of the signal.
- *Root Mean Square Energy (RMS)*: Measures the signal's power, reflecting its loudness.
- *Mel-Spectrogram*: Numerically represent the sound with frequencies converted to the Mel scale, a scale that approximate human ears' response to frequencies.

All the calculations were implemented through a pipeline with functions from the package librosa. The resulting vectors were transposed and averaged over time frames to reduce the dimensionality while retaining the most significant characteristics that represent the audio sample. Finally, we concatenated these compressed statistics to form an array representative of the audio file. The array was then taken as the input to our CNN classifier to make predictions. A diagram illustrating the whole pipeline can be found below (Figure 5).

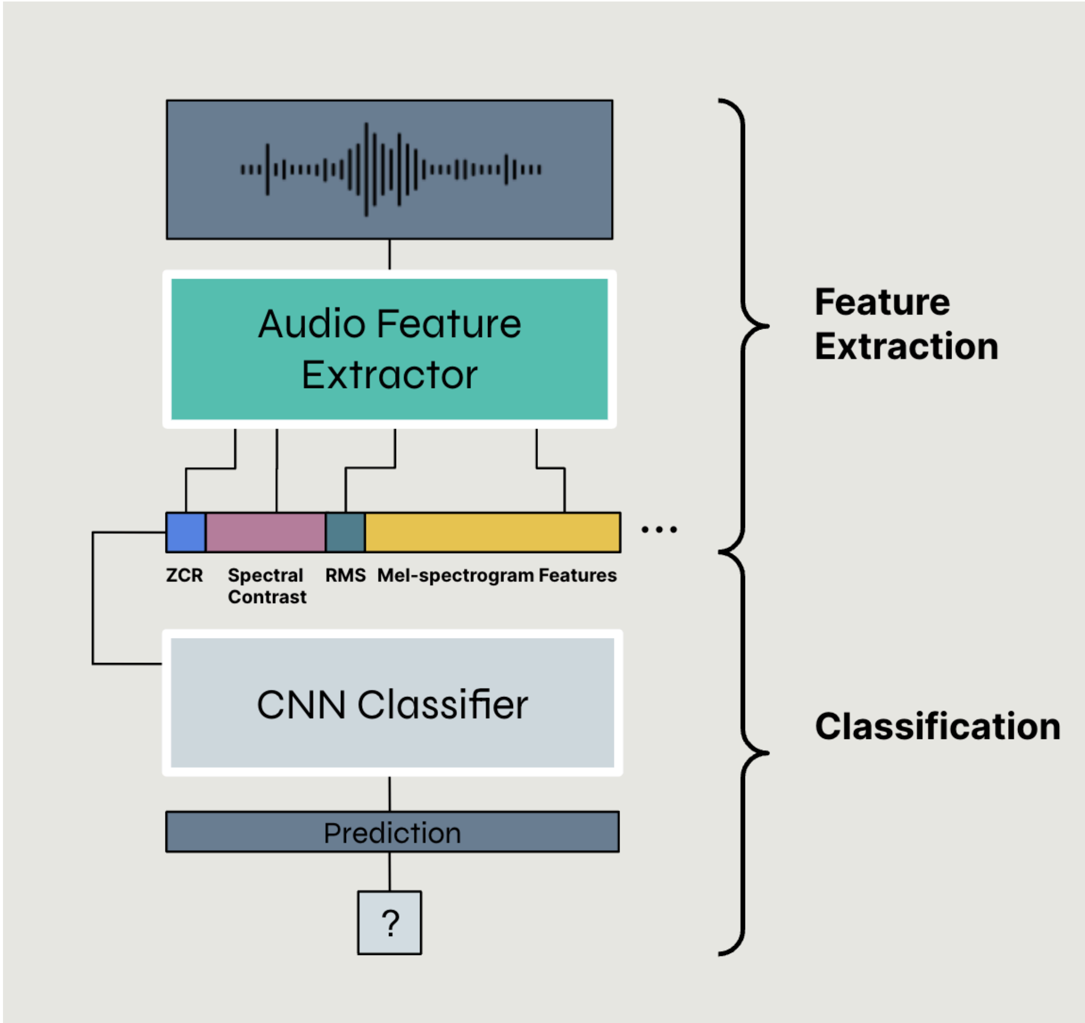

Figure 5. Model Architecture

3. Evaluation results

We tested our model on the test set, which contains 1185 data samples. The overall accuracy rate is 85.15%.

A confusion matrix clearly shows the model's performance on different classes of data. A clear diagonal line on the confusion matrix indicates that our model makes correct predictions on most of the test samples.

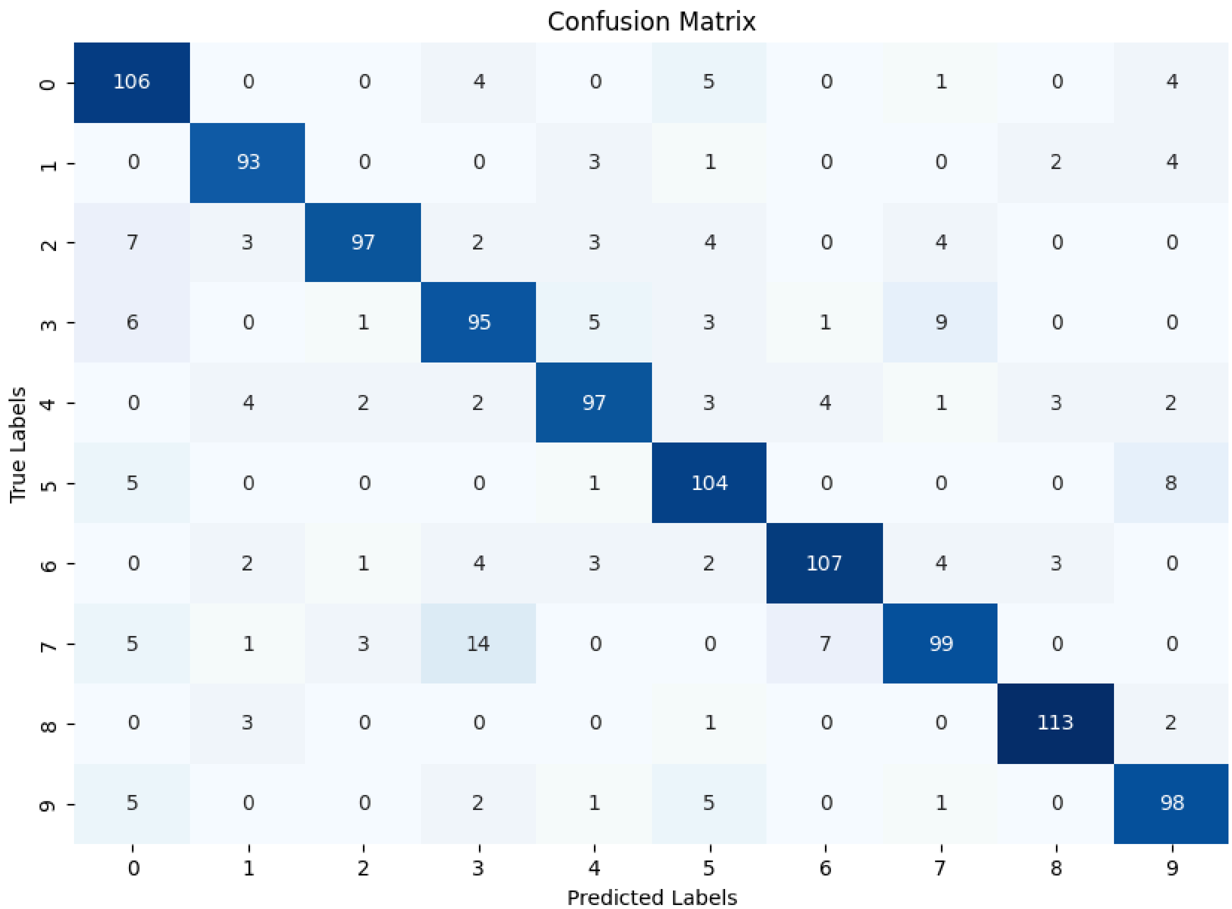

Figure 6. Confusion Matrix