Speech Processing Project

Project Name

Submitted By-

Group no. 20

Members

Anurag Pandey[23PGAI0059]

Kuldeep Semwal[23PGAI0124]

Problem statement:

There is a growing need for accurate classification of cat and dog sounds due to their prevalence as household pets and their potential impact on human well-being. Currently, there are limited tools available to automatically differentiate between the sounds made by cats and dogs, which can lead to misinterpretation of the animal's behaviour and needs. This can result in negative consequences for both the pets and their owners, such as miscommunication, ineffective training, and even injury. Developing a robust and reliable cat and dog sound classifier model can help address these challenges by enabling accurate and efficient identification of the sounds made by cats and dogs, which can lead to improved communication and care for these beloved pets. Additionally, such a model can have broader applications, such as in animal rescue operations and wildlife conservation efforts, where accurate identification of animal sounds is crucial for effective management and protection. Therefore, there is a pressing need for research and development of cat and dog sound classifier models to enhance the welfare and well-being of these animals and their human counterparts.

Datasets: We have created our own dataset by downloading cat and dog sound from different websites and then converting those mp3 files into .wav files.

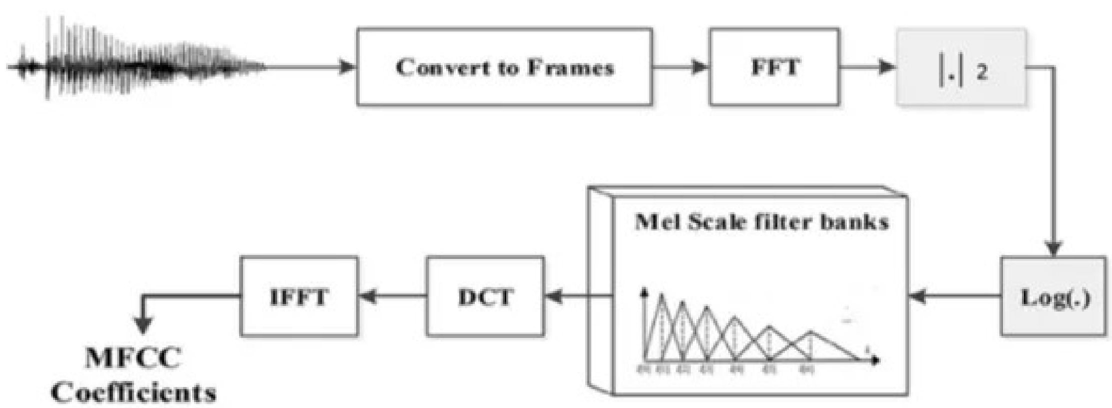
We have used data set in the link-

https://github.com/anuragdhirubhai/speech_classification_prjct/blob/master/dataset_link

GitHub Link for Project- https://github.com/anuragdhirubhai/speech_classification_prjct

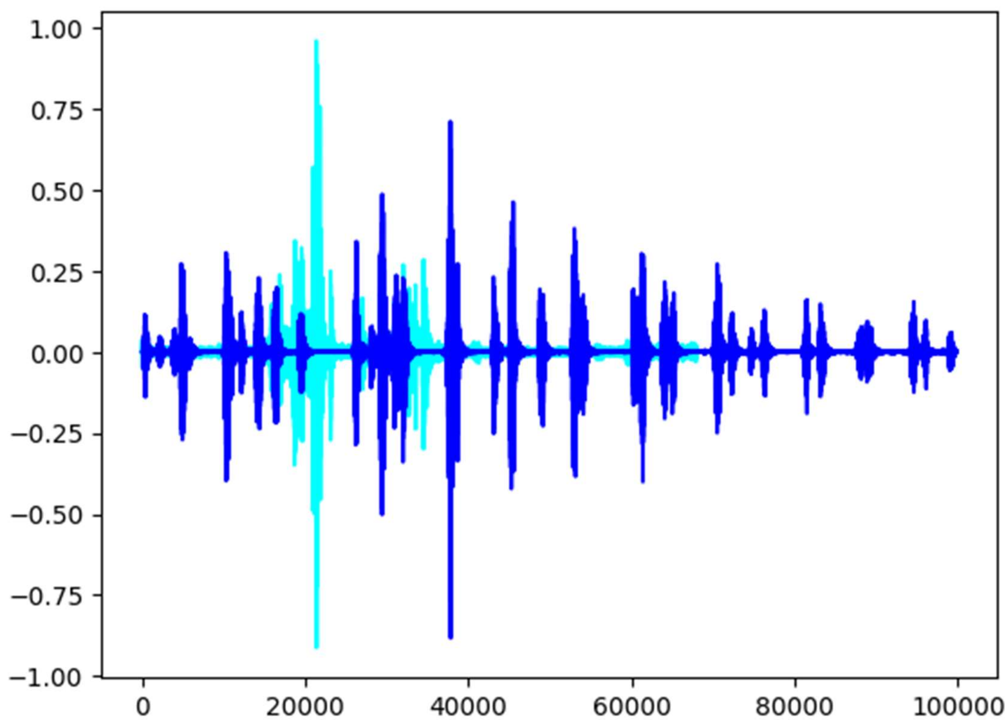
Solution Approach:

For ML Model-



MFCCs are commonly used to extract features from audio signals that are relevant for classification tasks. The MFCCs represent the spectral characteristics of the audio signal and can capture important information related to the pitch, timbre, and other acoustic properties of the sound. To implement our solution, we have first pre-processed the audio data to extract the MFCC features from the raw audio signals, by using a windowing function to segment the audio signals into smaller frames, and then computed the MFCC coefficients for each frame. These MFCC coefficients were then used as input features for the logistic regression classifier.

For Dl Model-



We utilized convolutional neural networks (CNNs) and TensorFlow/TensorFlow-io for feature extraction. Our approach converted the classification problem into a binary one, and we extracted audio features using spectrograms while also reducing amplitude and frequency. These features were then utilized as input for our neural network to predict how frequently a dog or cat made sounds in a particular clip. To apply windowing on the clips, we sliced the clips.

Workload Division:

Anurag Pandey - Research in domain, Model building, feature extraction and ML/Dl model implementation

Kuldeep Semwal- Domain research, data collection, data pre-processing and ML model development.

Abstract:

This project aimed to develop two machine learning-based classifiers for differentiating between cat and dog sounds. For the first model, we used logistic regression and extracted relevant features from the audio signals using Mel-frequency cepstral coefficients (MFCCs). The audio data was pre-processed to segment the signals into smaller frames, and the MFCC coefficients were calculated for each frame. The logistic regression classifier was trained on these features to classify cat and dog sounds. For the second model, we used convolutional neural networks (CNNs) and TensorFlow/TensorFlow-io for feature extraction. We converted the classification problem into a binary one and extracted audio features using spectrograms while also reducing the amplitude and frequency of the signals. The resulting features were then input to the neural network to predict the frequency of cat and dog sounds in each clip. To apply windowing to the clips, we sliced them.

Our evaluation of both models showed high accuracy in differentiating between cat and dog sounds. The logistic regression model achieved an accuracy of 83%, while the CNN-based model achieved an accuracy of 97%. Our results demonstrate the effectiveness of using both logistic regression and CNNs with spectrogram-based feature extraction for audio signal processing tasks. These models have the potential for a wide range of applications, including pet training, animal behaviour analysis, and wildlife conservation.

Introduction and Motivation:

Distinguishing between animal sounds is essential in animal behaviour analysis, wildlife conservation, and pet training. While trained individuals can differentiate between the vocalizations of cats and dogs, manual classification can be challenging, time-consuming, and subject to human bias. This project aims to develop machine learning-based models for differentiating between cat and dog sounds. We will create models that can accurately identify and classify different types of sounds produced by these animals, such as barking, meowing, growling, and purring. Our models can have various applications, including monitoring the behaviour of domestic pets, analysing the communication patterns of wild animals, and identifying the presence of feral cats and dogs in urban areas. Additionally, these models can aid in training pets and reducing the instances of nuisance barking or meowing. The success of this project can pave the way for developing similar models for other animal species, opening new avenues for research and conservation efforts.

Dataset:

Our system is trained on a dataset that includes various cat and dog sounds recorded in different situations, such as playing, barking, and meowing. Our training dataset contains .wav files for dogs and cats which we have used to test on .mp3 file or a longer clip to predict how many times a dog and a cat sound were present in the clip. Prior to training, we pre-processed the dataset by converting the sound files into spectrogram to extract features from sound signals and are created by applying the Fourier Transform of the sound signal at different time points.

Code Walkthrough:

For ML model-

code performs audio classification using MFCC features and logistic regression. First, it imports the necessary libraries such as scikit-learn, Librosa, NumPy, and pandas. Then it sets the root directory for the audio files and reads the list of file names. An empty data frame is initialized to store the MFCC features extracted from each audio file. The code then loops over each audio file, loads it using Librosa, extracts the MFCC features, and calculates the mean and standard deviation of each coefficient. The mean and standard deviation values are concatenated to form a feature vector, which is then appended to the final dataset along with the label. The final dataset is split into training and testing sets, and the feature variables are normalized. A logistic regression model is trained on the training set and evaluated on the testing set using accuracy, precision, recall, and F1 score metrics. The code outputs the evaluation metrics for the trained model.

For DL model-

Loading and Pre-processing Audio Data

Define file paths for cat and dog audio files. Define function to load WAV files and return audio signal tensor. Load cat and dog audio files and plot their waveforms. Define file paths for directories containing cat and dog audio files. Create TensorFlow Dataset objects for cat and dog audio files. Combine datasets into a single dataset. Calculate mean, minimum, and maximum length of cat audio files. Define function to pre-process audio data and return spectrogram tensor and label. Pre-process audio data using TensorFlow's Dataset API.

Building a Keras Model

Define Keras model using the Sequential API. Compile Keras model with adamax optimizer, binary Cross entropy loss, and accuracy and binary cross entropy metrics.

Training and Evaluating the Model.

Define custom call back class to stop training if accuracy on the training set exceeds a certain threshold. Train Keras model using fit () method and custom call back. Evaluate Keras model on test set using evaluate () method.

Summary and Future Scope:

This report provided an overview of the project and related work, followed by a detailed explanation of the learning and performance tasks involved in classifying cat and dog audio files. The approach used involved a deep neural network that achieved an accuracy of 97%.

In the future, we plan to explore the use of other high-level models to further improve classification accuracy. Additionally, we aim to develop a system capable of directly processing live audio input.