Name: Jaskaran Singh

Roll No: 102103296

Group: 4CO11

Data Analysis

The code provides an overview of how an audio dataset of spoken words is analyzed, partitioned, and visualized, offering insights into the characteristics of the dataset and ensuring its proper preparation for tasks like model training and evaluation.

The code describes the analysis and partitioning of an audio dataset in which the audio files are .wav format, likely containing spoken words. The steps include:

Basic Information Collection: The dataset is located in a specified directory, and all audio files within that directory are analyzed. The dataset is stored in a directory (dataset_path), which the script explores using os.walk() to find all .wav files.For each file, it extracts basic properties such as word label (from folder names), speaker ID (from filenames), and audio metadata (duration, sample rate, channels) using the wave library.It collects these properties into a list, which is later converted into a Pandas DataFrame for easier analysis.

For each audio file, information like the word label (based on the directory structure), speaker ID (from the filename), and audio properties (duration, sample rate, number of channels) are extracted.

Descriptive Statistics: After collecting the data, a summary of the dataset is produced. The *describe_data* () function provides an overview of the dataset, including:

. This includes:

- The number of unique words (labels) spoken in the dataset.
- The number of unique speakers.
- Count of audio files per word, a breakdown of the number of files per word.
- Audio duration statistics i.e. descriptive statistics about the duration of the audio files.
- Distribution of the sample rates and audio channels (mono or stereo).

Saving the Data: The collected data and statistics are saved into a CSV file for future use and further analysis. The dataset statistics are saved to a CSV file named '*speech_commands_dataset_statistics.csv'* for future use.

Data Visualization: Several key visualizations are created:

- The distribution of the number of files for each word in the dataset, showing how frequently each word appears.

- A histogram showing the range of durations of the audio files, illustrating the spread of file lengths.
- A bar chart displaying the number of files recorded by each speaker, with a focus on the top 20 speakers.
- A bar chart showing the average duration of audio files for each word label, offering insights into whether some words have consistently longer or shorter recordings.

The script explores a speech dataset, collects and analyzes basic audio statistics, partitions the data into training/validation/testing sets, and visualizes important aspects such as word and speaker distributions, along with audio durations.

```
Number of unique words: 36
Number of unique speakers: 2624

Word Counts:
 word_label
zero                4052
five                4052
yes                 4044
seven               3998
no                  3941
nine                3934
down                3917
one                 3890
two                 3880
go                  3880
stop                3872
six                 3860
on                  3845
left                3801
eight               3787
right               3778
off                 3745
four                3728
three               3727
up                  3723
dog                 2128
wow                 2123
house               2113
marvin              2100
bird                2064
```

```
happy                2054
cat                  2031
sheila               2022
bed                  2014
tree                 1759
backward             1664
visual               1592
follow               1579
learn                1575
forward              1557
_background_noise_      6
Name: count, dtype: int64
```

```
Audio Duration Statistics:
 count    105835.000000
mean          0.984649
std           0.508240
min           0.213312
25%           1.000000
50%           1.000000
75%           1.000000
max          95.183125
Name: duration, dtype: float64
```

```
Sample Rate Statistics:
 count    105835.0
mean      16000.0
std           0.0
min       16000.0
25%       16000.0
50%       16000.0
75%       16000.0
max       16000.0
Name: sample_rate, dtype: float64

Channel Distribution:
 num_channels
1    105835
Name: count, dtype: int64
```

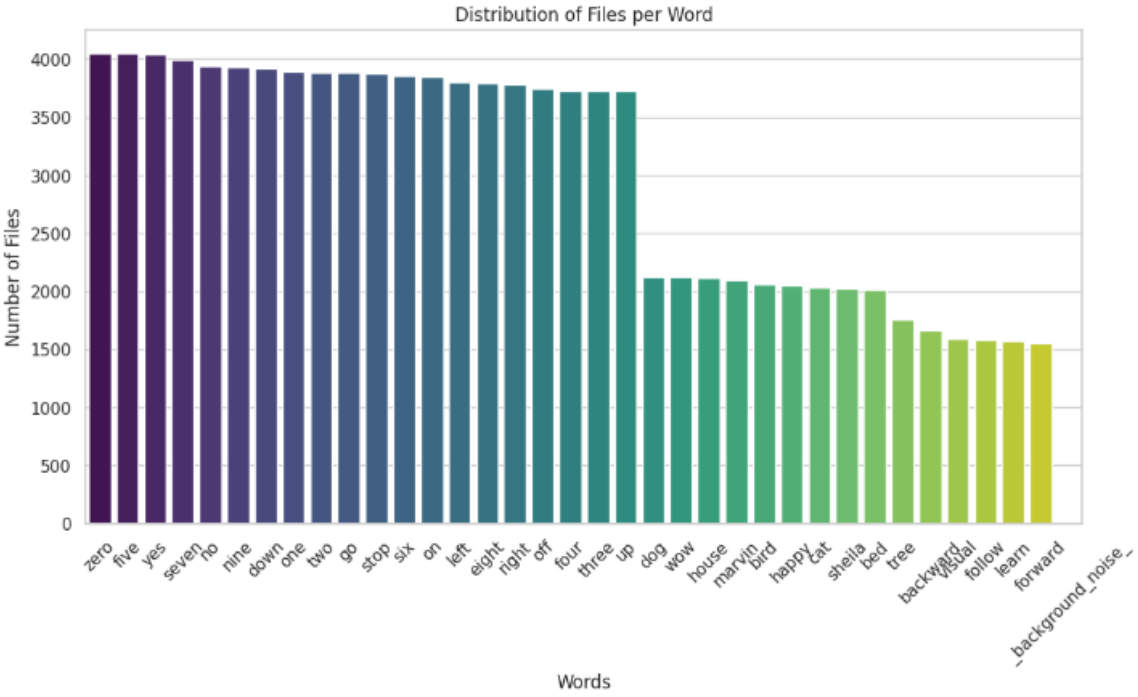


Figure 1: Distribution of Files per Word

Figure 1 represents the bar plot showing the number of files available for each word label

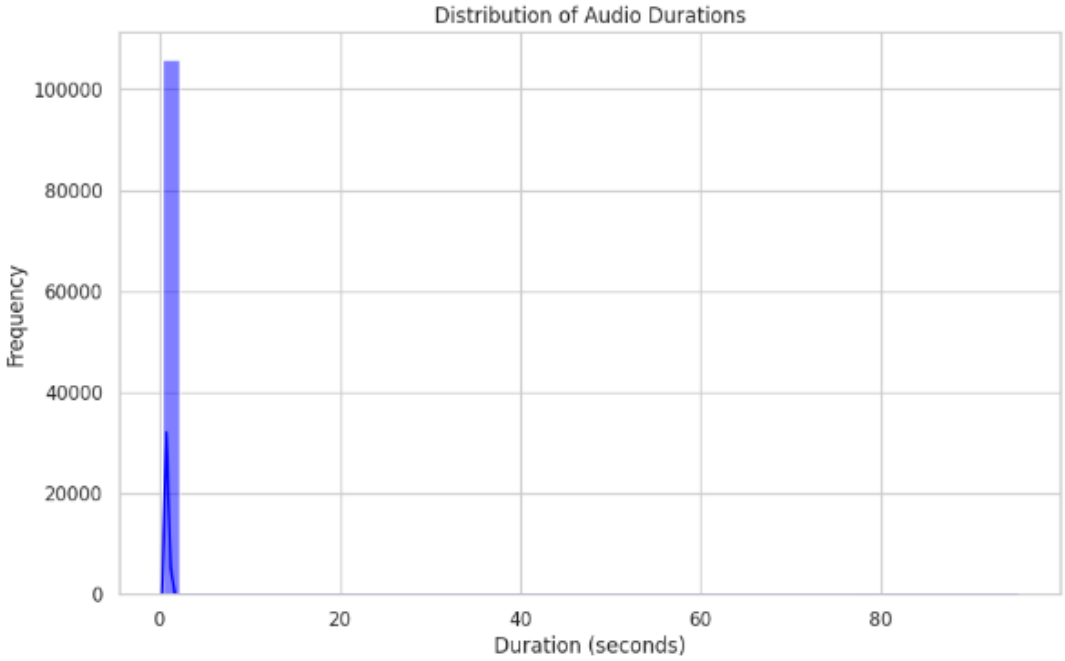


Figure 2: Distribution of Audio Durations

Figure 2 represents histogram showing the range of audio durations.

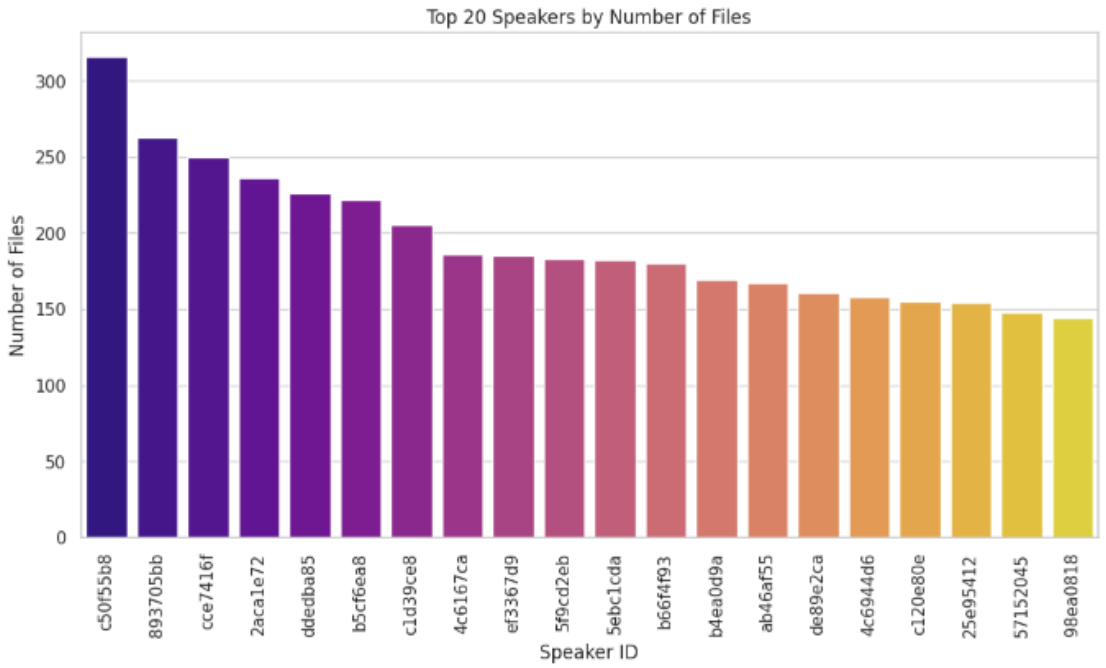


Figure 3: Distribution of Files per Speaker:

Figure 3: represents the bar plot showing the top 20 speakers based on the number of files they recorded.

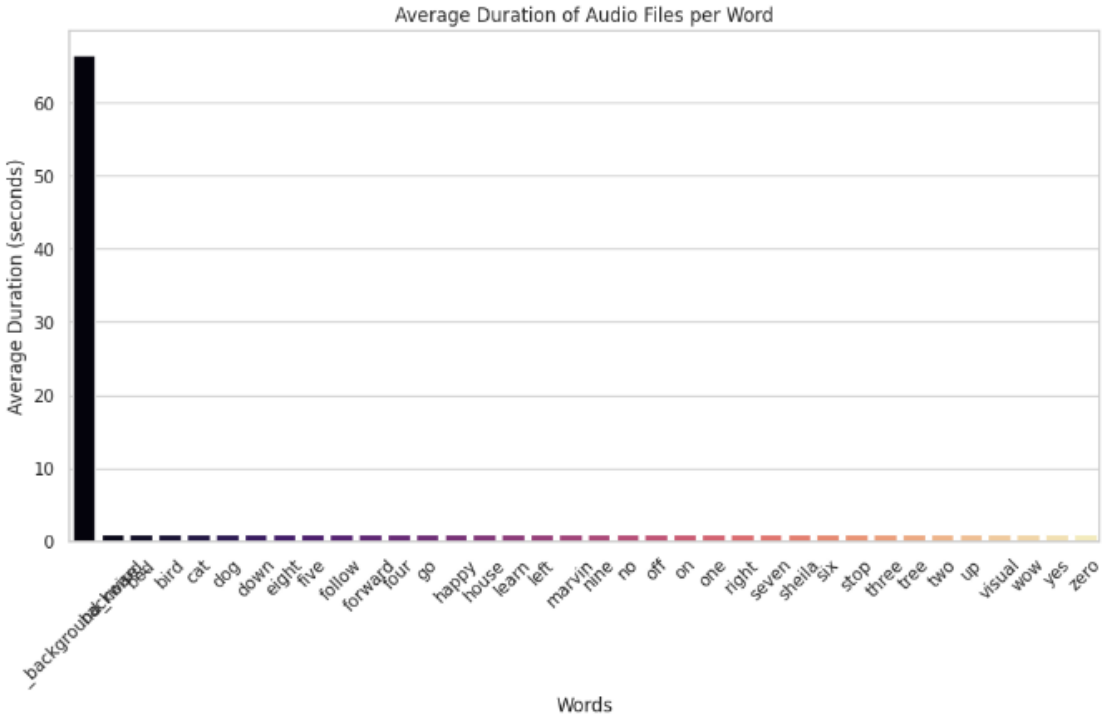


Figure 4: Average Duration of Audio Files per Word

Figure 4 represents the bar plot showing the average duration of audio files for each word.

	A	B	C	D	E	F	G	H
1	file_path	word_label	speaker_id	duration	sample_rate	num_channels	num_frames	set
2	/kaggle/input/jaskaran-speech/no/4c6944d6_nohash_3.wav	no	4c6944d6	1	16000	1	16000	training
3	/kaggle/input/jaskaran-speech/no/4e99c1b7_nohash_3.wav	no	4e99c1b7	1	16000	1	16000	training
4	/kaggle/input/jaskaran-speech/no/97f4c236_nohash_3.wav	no	97f4c236	1	16000	1	16000	testing
5	/kaggle/input/jaskaran-speech/no/cb2929ce_nohash_3.wav	no	cb2929ce	1	16000	1	16000	training
6	/kaggle/input/jaskaran-speech/no/ca4d5368_nohash_3.wav	no	ca4d5368	1	16000	1	16000	testing
7	/kaggle/input/jaskaran-speech/no/ad63d93c_nohash_1.wav	no	ad63d93c	1	16000	1	16000	validation
8	/kaggle/input/jaskaran-speech/no/aeb99b1c_nohash_1.wav	no	aeb99b1c	1	16000	1	16000	training
9	/kaggle/input/jaskaran-speech/no/89947bd7_nohash_4.wav	no	89947bd7	1	16000	1	16000	training
10	/kaggle/input/jaskaran-speech/no/c5a1e46c_nohash_3.wav	no	c5a1e46c	1	16000	1	16000	training
11	/kaggle/input/jaskaran-speech/no/0a2b400e_nohash_0.wav	no	0a2b400e	1	16000	1	16000	training
12	/kaggle/input/jaskaran-speech/no/45692b02_nohash_1.wav	no	45692b02	1	16000	1	16000	validation
13	/kaggle/input/jaskaran-speech/no/9d32f10a_nohash_0.wav	no	9d32f10a	1	16000	1	16000	validation
14	/kaggle/input/jaskaran-speech/no/14c2b13d_nohash_0.wav	no	14c2b13d	0.810625	16000	1	12970	training
15	/kaggle/input/jaskaran-speech/no/01648c51_nohash_1.wav	no	01648c51	1	16000	1	16000	training
16	/kaggle/input/jaskaran-speech/no/e0315cf6_nohash_1.wav	no	e0315cf6	1	16000	1	16000	training
17	/kaggle/input/jaskaran-speech/no/28497c5b_nohash_1.wav	no	28497c5b	1	16000	1	16000	testing
18	/kaggle/input/jaskaran-speech/no/c22d3f18_nohash_0.wav	no	c22d3f18	1	16000	1	16000	testing
19	/kaggle/input/jaskaran-speech/no/a1533da4_nohash_0.wav	no	a1533da4	1	16000	1	16000	testing
20	/kaggle/input/jaskaran-speech/no/e1469561_nohash_0.wav	no	e1469561	1	16000	1	16000	testing
21	/kaggle/input/jaskaran-speech/no/db24628d_nohash_3.wav	no	db24628d	1	16000	1	16000	testing
22	/kaggle/input/jaskaran-speech/no/b06c19b0_nohash_1.wav	no	b06c19b0	1	16000	1	16000	training
23	/kaggle/input/jaskaran-speech/no/997867e7_nohash_0.wav	no	9.98E+12	0.882375	16000	1	14118	training

Figure 5: Descriptive Statistics

Figure 5 shows the snapshot of dataframe having number of unique words (labels), number of unique speakers, count of audio files per word, a breakdown of the number of files per word,

audio duration statistics i.e. descriptive statistics about the duration of the audio files, and distribution of the sample rates and audio channels (mono or stereo).

Main Code

The code using the *'SpeechCommands'* dataset containing 35 spoken commands from different speakers, with each audio file lasting approximately one second. The process demonstrates how to load, process, and classify these audio commands using the PyTorch and Torchaudio libraries.

It starts by verifying the availability of a CUDA-compatible GPU, which enhances the speed of training and testing the neural network. Following this, the *SpeechCommands* dataset is introduced, containing audio commands such as "yes," "no," "up," "down," and others. Using torchaudio, the audio data is loaded as tensors, and a custom class is created to divide the dataset into training, validation, and testing subsets. The training set excludes validation and testing data, and the audio waveforms are visualized for better understanding of their structure.

Further data formatting is addressed. The audio is downsampled to 8000 Hz, which reduces the size and increases the processing speed without significant loss of classification power. Each spoken word (command) is encoded as an index from the list of labels. A collate function is implemented to pad and batch the data, making it suitable for feeding into the neural network during training.

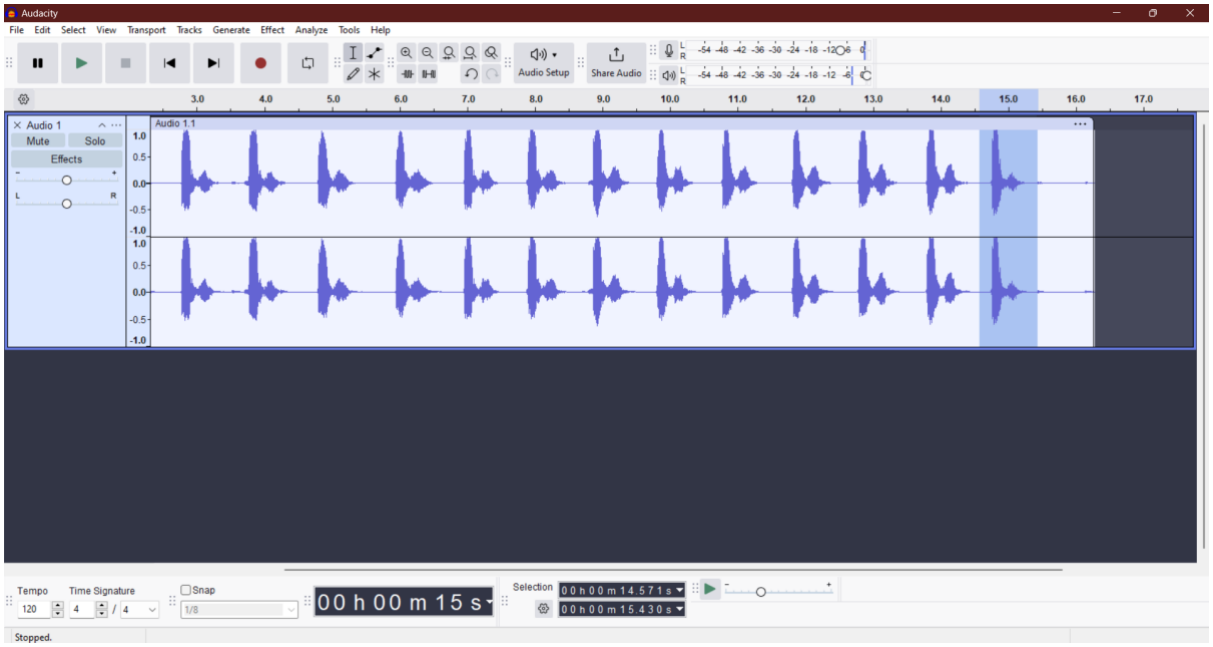
A convolutional neural network (CNN), inspired by the *M5 network architecture*, is then defined to process the raw audio data. Although CNNs are typically used for image processing, they can be applied to audio data since raw audio waveforms can be treated similarly to images. The model consists of several convolutional layers with batch normalization and pooling layers. The *Adam* optimizer is used along with a learning rate scheduler to reduce the learning rate during the training process.

The process then defines functions for both training and testing the network. During training, the model is placed in training mode, and the loss is calculated using negative log-likelihood. After each epoch of training, the model is switched to evaluation mode, where its accuracy on the test set is measured. These results are displayed, and the model's accuracy is expected to surpass 85% after sufficient training epochs.

To make predictions on new audio samples, a function is provided. Users can record their own commands and use the trained model to classify the audio in real-time. The final section mentions additional preprocessing methods such as using Mel-frequency cepstral coefficients (MFCC), which are commonly employed in speech recognition to reduce dataset size while retaining important features for classification.

Custom Data Creation

Custom dataset is made for all 35 classes. For each class 15 samples were recorded due to time constraints. The data was recorded using the software audacity.



The required word was spoken 15 times with the time for speaking being 1 second (shown in above figure). After that each waveform was clipped into individual .wav files with the name (16 bit hash of my roll number (102103296) _ index of waveform. wav).