Audio Classification Using Deep Learning

DYARI MOHAMMED SHARIF

dma140h@cs.soran.edu.iq

Introduction

Sound recognition may be one of our senses that has kept humans alive. Acoustics play a vital part in everyday life, from recognizing a danger nearby to being motivated by music, from bands of human voices to the singing of a sparrow. Sound classification is one of the most often used applications in Audio Deep Learning. It requires mastering the ability to recognize sounds and predict which group they belong to. This type of problem may be applied to a number of circumstances, such as classifying music clips to determine the genre of music or classifying short utterances by a group of speakers to determine the speaker based on the voice (Kim, et al., 2020).

It's crucial to categorize the audio sources, and it's already widely used for a variety of issues. Deep Learning (DL) has become one of the most prominent methods for tackling a variety of challenges, including this one. There are several methods to define music genres in harmony. In the audio domain, DL has showed exceptional performance by effectively recognizing multiple target class patterns in time-series data, with the environment also being important since batches of other sounds, commonly known as noise, can be generated, interfering with the genuine data (Chandu, et al., 2020; Al-Emadi, et al., 2019; Raza, et al., 2019).

DL has lately been perhaps the most popular technique for solving many problems in our lives due to its precision and the improvement of processing equipment like the CPUs and/or GPUs. Many DL architectures have grown and been used in domains including speech recognition, natural language processing, and a range of classification problems in recent decades, where they have consistently outperformed previous techniques (Callaway, 2020; Garcia-Garcia.A., et al., 2017).

In this work, we implement a Multi-layer Perceptron (MLP) model using Signal Processing (SP) techniques to identify environmental sounds from the samples.

1. Dataset

The project's goal, as previously stated, is to classify environmental voices in a natural setting. To accomplish so, we used the fifteenth version of a License free publicly available database, called Environmental Sound Classification 50 which is raw audio classification of environmental sounds (Moreaux.Marc, n.d.). The dataset is owned by Marc Moreaux and was last updated on 2018-10-26 and was created on 2017-10-17. It is available at: https://www.kaggle.com/mmoreaux/environmental-sound-classification-50. As of June 27, 2021, the activity stats are: 32.2k views, 3286 downloads, 0.1 download per view ratio and 17 total unique contributors. The dataset consists of 50 16KHz WAV files for 50 distinct classes. 40 audio samples of 5 seconds each are assigned to each of the classes. All of these audio recordings were concatenated by class to create 50 wave files totalling 3 minutes and 20 seconds. The author also provides a .csv file in which there are four columns: filename, category, filename-test and category-test. The dataset had already divided the samples into test and train data, for the sake of not getting varying results of accuracy if computed.

2. Data Preparation

As for most deep learning problems, we will follow these steps, and this one is not exceptional.

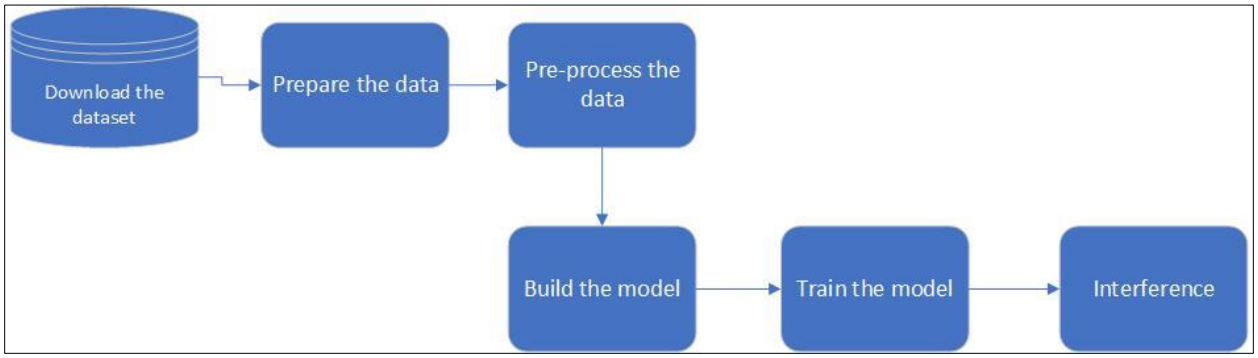

Fig. 1 The workflow of Deep Learning

This problem's data will be straightforward:

- The audio file paths (X) are the features.

- The class names are represented by the target labels (y).

We utilize the metadata file from the dataset directly because it already has this information. The metadata was provided along with the samples when we downloaded them. Each audio file's metadata provides information about it. We read it using the Pandas library in python because it's a CSV file. The information then is used to prepare the features and label data.

2.1 Feature Extraction

Computer-assisted learning ML takes raw data and extracts features to produce a rich representation of the material. To draw conclusions, we must learn the fundamental information without the noise (if it is done correctly). Turning to voice recognition, our goal is to identify the optimal sequence to match the audio. Mel-frequency cepstral coefficients (MFCC), which include 39 features, are one common audio feature extraction approach. The number of features is low enough that we are forced to memorize the auditory information. The amplitude of frequencies is governed by 12 factors. It gives us a sufficient number of frequency channels for audio analysis.

The flow of obtaining the MFCC characteristics is shown below, in Fig. 2.

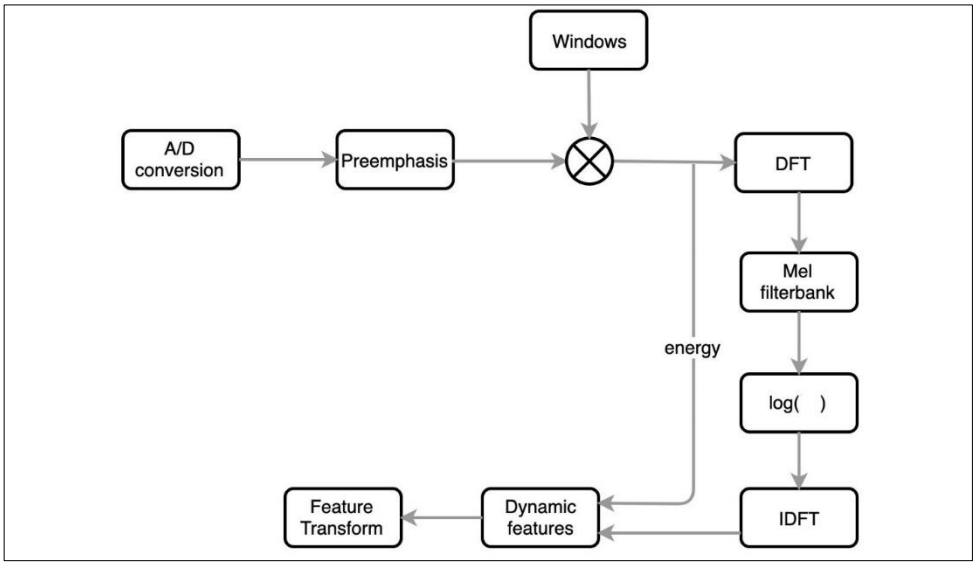

Fig. 2 Extracting the MFCC features workflow

The following are the main goals:

- Remove the pitch information by removing the vocal fold excitation (F0).
- Make the extracted characteristics stand on their own.
- Adapt to how people perceive sound volume and frequency.
- Capture the phone's dynamics (the context).

2.1.1 Mel-frequency cepstral coefficients (MFCC)

We will discuss each and every step of it, at a time.

A/D conversion: A/D conversion digitizes the material by sampling the audio clips and transforming the analog signal into discrete space. Typically, sampling frequencies of 8 or 16 kHz are utilized. Here, in our project, a sampling rate of 16kHz are utilized.

Pre-emphasis: The quantity of energy in the high frequencies is increased via pre-emphasis. Lower frequencies have greater energy than higher frequencies for voiced parts like vowels. This is referred to as spectral tilt, and it is linked to the glottal source (how vocal folds produce sound). Increasing the high-frequency energy makes formant information more accessible to the acoustic model. This enhances the accuracy of phone detection.

Windowing: Signals are divided into sliding frames via windowing. We can't, however, just cut it off at the frame's edge. The abrupt drop in amplitude will result in a lot of noise, which will be seen at high frequencies. The amplitude should progressively decrease at the frame's border to slice the audio.

Discrete Fourier Transform (DFT): Following that, we use DFT to extract frequency domain information.

Mel filterbank: Our hearing perception differs from the equipment readings. The perceived loudness of people varies depending on frequency. In addition, when frequency increases, perceived frequency resolution diminishes. Humans are less sensitive to higher frequencies. The Mel scale translates the recorded frequency to the frequency we experience in the context of frequency resolution.

Log: A power spectrum is produced by Mel filterbank. When the energy level is high, humans are less sensitive to tiny energy fluctuations than when the energy level is low. It's logarithmic, in reality. The log will be extracted from the Mel filterbank's output in the following phase. This also

lowers the number of acoustic variations that aren't useful for speech recognition. Following that, we must address two additional needs. First, we must remove the F0 information (pitch) and ensure that the retrieved characteristics are independent of each other.

Cepstrum — IDFT: The form of the vocal tract is controlled by our articulations. The vocal fold vibrations are combined with the filter formed by our articulations in the source-filter model. The structure of the vocal tract will suppress or amplify the glottal source waveform at different frequencies.

Cepstrum is made up of the first four letters of the word "spectrum" reversed. The Cepstral, which separates the glottal source from the filter, is the next stage.

Dynamic features: There are 39 features in MFCC. Context and dynamic information are crucial in pronunciation. The formant transitions can be used to identify articulations such as stop closures and releases. The context information for a phone is provided by characterizing feature changes over time.

Cepstral mean normalization: Following that, we can do feature normalization. We use the mean to normalize the features. Over all the frames in a single utterance, the mean is determined using the feature value j.

3. Experiments

Using Pandas, we read the metadata which includes the information for the data that we are going to populate. Then, we use a MLP to classify 50 classes. We extract features from each audio sample using MFCC features and populate them to the model. After using various parameters for our mode, we have come up with the following structure while we were trying to reduce the complexity. Input layer has 40 neurons, which holds the 39 coefficients of MFCC of audio samples. Furthermore, we have three more hidden layers, each with 400 neurons and dropout being 0.5. The activation functions of all layers are sigmoid except the output layer which is SoftMax. The output layer has 50 neurons because we want to classify 50 kinds of sound. The accuracy we achieved was 99%. Moreover, we used batch size of 32 for all out experiments. As mentioned earlier, our problem's data will be the features of the audio file paths (X) and the target labels of the class names (y). The following is the snippet code we used for the structure of MLP network. We have come with models that had %99 accuracy with less epochs but with 1000, the accuracy was even come close 100%. The following tables are our experiment. Furthermore, we used two different activation functions but the sigmoid function performed better.

Table 1 the details of the various parameters we used for our experiment, in this table we used sigmoid function in all layers except the output layer.

Activation Function and batch size	**Number of Epochs**	**Number of neurons per layer**	**Accuracy**
Sigmoid, 32	50	400	%59
	100		%77
	150		%84
	200		%88
	250		%92
	300		%95
	350		%97
	400		%97
	450		%98
	500		%98
	600		%98

	700		%99
	800		%99
	900		%99
	1000		%99

Table 2 the details of the various parameters we used for our experiment, in this table we used relu function in all layers except the output layer.

Activation Function and batch size	**Number of Epochs**	**Number of neurons per layer**	**Accuracy**
relu, 32	50	400	%50
	100		%68
	150		%80
	200		%84
	250		%85
	300		%86
	350		%89
	400		%87
	450		87%
	500		87%
	600		%90
	700		%89
	800		%90
	900		%91
	1000		%90

4. Conclusion

One of the most commonly utilized applications of Audio Deep Learning is audio categorization. It's crucial to categorize audio sources, which is already extensively utilized for a number of problems, and Deep Learning has emerged as one of the most popular approaches for handling a variety of problems, including this one. Even though all the data samples we fed the model width, were of the same size, the model can take care of various lengths. We tried some more data that had different lengths in the testing phase, the results were positive. This project can be expanded to include more classes.

References

Al-Emadi, S., Al-Ali, A., Mohammed, A. & Al-Ali, A., 2019. *Audio Based Drone Detection and Identification using Deep Learning.* s.l., ResearchGate: International Wireless Communications & Mobile Computing Conference (IWCMC 2019).

Callaway, E., 2020. 'It will change everything': DeepMind's AI makes gigantic leap in solving protein structures. *nature,* 30 November.

Chandu, B. et al., 2020. *Automated Bird Species Identification using Audio.* Amaravati, India, IEEE.

Garcia-Garcia.A., et al., 2017. A Review on Deep Learning Techniques. *arxiv.org.*

Kim, H., Karabash, D., Korotkov, M. & Chen, T., 2020. *Detecting Sounds with Deep Learning.* [Online]
Available at: https://towardsdatascience.com/detecting-sounds-with-deep-learning-ed9a41909da0

Moreaux.Marc, n.d. *Kaggle.* [Online]
Available at: https://www.kaggle.com/mmoreaux/environmental-sound-classification-50?select=esc50.csv
[Accessed 2007].

Raza, A. et al., 2019. Heartbeat Sound Signal Classification Using Deep Learning. *MDPI: Sensors,* 19(21).