Domain Background

Because of not dealing on a speech recognition problems, I decided to strengthen my skills in such domain and work on a speech classification example that cover my need for working in speech recognition tasks. In my project, I would like to build a radio website that play a radio station and provide additional information below on the same page about the speaker that is currently speaking on the radio.

I decided to focus this project in building a demo project that covers a simpler problem than the final solution so I will just focus on building a speech classifier solution between two speakers and I will increase it's difficulty incrementally "Future Work".

Problem Statement

The problem needed to be solved in my project was to classify who is currently speaking on a certain radio station in order to be able to display more data about the speaker such as name, age, previous work, photos, etc. I think that it's not an easy problem to classify huge number of speech that could be possible in a radio station, but the radio station that I am targeting is currently having a few number of speakers so I think the problem will not require much processing requirements in terms of hardware capabilities.

The scope that I am targeting and I will implement in the capstone project is just classify simple conversations and assign each to their speaker so I will work on the best ways to classify speech data, the best libraries such as Libarosa library, and all the way towards classifying a speech.

Datasets and Inputs

The data that I will be using is the Free Spoken Digit Dataset (FSDD) that mentioned here: https://github.com/Jakobovski/free-spoken-digit-dataset and it simply some audio/speech dataset consisting of recordings of spoken digits in wav files at 8kHz. The recordings are trimmed so that they have near minimal silence at the beginnings and ends.

FSDD is an open dataset, which means it will grow over time as data is contributed. In order to enable reproducibility and accurate citation the dataset is versioned using Zenodo DOI as well as git tags.

The wav files that is ready made in the dataset above is perfect to be fed in the library that I choose for the project which is librosa so there is no need for extra modification for the dataset other than possible scaling or something related to the classification task.

Solution Statement

The solution which I will focus on the capstone is a simple classification problem for a speech dataset. The dataset is simple “about 4 speakers”, but I will use it as a demo for building the structure of the complete radio station solution “Future Work”. I will use the dataset provided above to create a suitable classifier that able to successfully classify speakers “assign audio clips to it’s speaker”.

Benchmark Model

The complete solution will be a website that targets a sports radio station and display some information about the current speaker. The solution will simple make a wav file out of the currently speaking speaker and feed this to the classifier which should be able to classify the speaker based on it’s stored dataset. “There is a given info that the number of speakers on that certain radio station in limited so it will be easy to make a dataset of all the speakers in that radio station”.

After correctly classifying the speaker, I will display some information about him/her that I see that will be useful for the visitor of the website such as the name, age, previous work, etc.

Evaluation Metrics

The success of my capstone is to build a classifier that could easily classify the 4 only speakers that present in the above dataset. I will begin by sampling the wave files which is simply representing it in terms of numbers so it could be used in a numpy array or the rest of the libraries that I will use on the project. As the Nyquist–Shannon sampling theorem showed that if our sampling rate is high enough, we are able to capture all the information in the signal and even fully recover it.

I will use MLP network to build my model and for the evaluation metrics I will use simple MLP settings such as 'categorical_crossentropy' for a loss function, ‘adam’ as an optimizer and ‘accuracy’ as a metices.

Project Design

The workflow of the project will be loading the data mentioned above, access it, feed the wave recordings for each speaker, build an MLP model that could simply use parts of the recordings and the assigned speakers and form it’s training and testing sets, and finally embed the solution in a final complete solution that will be used in a radio station website to classify the currently speaking person in a certain time during that day, then use this information to display info about the speaker.