

Data Science Task

Gender and Speaker classification

Part 1. Gender Classification from audio files

1. Please find and download part of the LibriSpeech [here](#). Find some time to read the documentation of the database. Download the dev-clean corpus, which is relatively small in size. This database contains directories with ids. Each id represents one speaker and contains audio files associated with that speaker. You will also find a file that indicates the gender of each speaker.
2. In case you have computational issues, please feel free to select a subset of the dataset. However, try to keep at least 18 speakers (9 male and 9 female). The next step would be to extract features for the audio files. The most common type of features are the MFCC features. Feel free to use existing libraries in order to extract MFCC features for all the audio files of your dataset.
3. Once you have the MFCC features extracted, perform some data analysis to better understand the data you are working with (features distribution, outliers detection, normalization techniques) and try to correlate them to the gender values.
4. After the previous step, you can start playing with various Machine Learning Algorithms. We would like you to perform a gender classification task. For each audio file you create a gender label. Build train and test (unseen while training) sets. **Be careful when you split your dataset in train and test.** Train a classifier on the train set and predict the labels of the test set. Measure accuracy and report performance. Try to experiment with at least 4 classifiers. Use two of them as baselines (one naive and one more sophisticated), a neural network and a deep learning model, such as a CNN. Once again, feel free to use any libraries that are convenient for you.

Part 2. Speaker Classification from audio files

1. Now that you have completed the first task, let's try something new. Read the wav2vec2.0 [explanation](#) (if you are brave you can find and read the [paper](#)) and try to understand its basic and high-level concepts.
2. Download the wav2vec2.0 base model (No Finetuning) from this [repo](#), use it as [features extractor](#) to create features from audio signals. Apply the wav2vec model to the audio files you used in the previous tasks. We suggest you apply the model not on the entire file audio, but on 2 seconds subsequent chunks (though increasing the amount of samples).
3. We ask you to do two types of analysis over a speaker classification task:
 - a. A quantitative analysis, using prediction models (starting from the features you just extracted). As before, use objective metrics (such as precision, recall and

accuracy) to evaluate your results. Be careful when you split your dataset in train and test, it can be tricky.

- b. A qualitative analysis, using algorithms of dimensionality reduction and clustering and visualizing the results.

Deliverables

1. It would be great to use the Jupyter Notebook for this assignment. Please try to show us your coding skills. Clean code with comments and correct programming principles are much appreciated.
2. The assignments should be pushed on GitHub. Please share the link of the repository with us when you are done with the assignment.
3. For the aforementioned tasks, analyze and interpret the results. Don't focus only on numbers. Try to explain why an algorithm performs better/worse than the others. The important part for us is to see the way you think and deal with a project like that. Don't be afraid of interpreting the results and making mistakes. We are not going to judge that negatively.