Laboratory 7 Report - A Speech Recognition Task

Francesco Capuano
Polytechnic of Turin
Turin, Italy
francesco.capuano@studenti.polito.it

***Abstract*—This is a report for the task of speech recognition on the *free spoken digit dataset* assigned in Laboratory 7.**

***Index Terms*—Speech recognition, Fourier Transformations, Spectrogram, Signal Analysis.**

I. The Dataset

The provided data are organized in two different collections. In the "dev" folder the data are represented as a collection of audios which are named in the format *recording-id_number-pronounced.wav*, hence a collection of *labelled* data in which each recording is associated with the corresponding spoken digit. In the "eval" folder the data are instead presented with no label, thus leading to the conclusion of using these recording as a test.

II. The Data

The documentation related to the provided recordings clearly state that the sampling of the audio is done with a frequency of 8kHz. Being these recording related to human voice, whose frequency's band extends until approximately 3000Hz, from the Nyquist Sampling Theorem, it is more than clear that this sampling rate is actually such that the original signal is almost perfectly sampled.

A. Data Processing

The recordings are translated into arrays via using the *scipy.io* method related to waves processing. The `wavfile` method allows to obtain the signal represented as an array of numbers whose length is equal to the value

$$n = t \cdot f_s$$

Where t represents the duration of the audio file and f_s represents the sampling rate used in the read of the audio file. From the fore defined equation is then more than clear that the length of the array crucially depends on the duration of the audio, which is a parameter that obviously varies according to the speaker (one could pronounce the same number faster than an another one), to the number (some number takes less time to be pronounced than another one) and generally to conditions of the recording (i.e. the fact that the recording did not get stopped properly). However, even in this context of high variability, some assumptions can be made on the whole data.

It is clear that if $\bar{\mu}$ and $\bar{s}$ represents the mean duration and the standard deviation of the audios of *all the speaker for all the spoken digits*, then all the recordings which have a duration

$$d^{\star} : d^{\star} \leq \bar{\mu} - 3\bar{s} \vee d^{\star} \geq \bar{\mu} + 3\bar{s}$$

could be classified as outliers with a sufficiently high degree of confidence, hence removing them should improve the quality of the remaining data.

B. Model Selection

The audio signal could be represented with some type of quantitative representation. Finding the proper type of features that can identify a particular signal is a crucial step towards the selection of a predictive model.

Given the structure of the data, the chosen model is a *Random Forest*.

C. Feature Extraction

In order to train the Random Forest properly the training set should contain records with the same number of features. Because the number of elements in the representation in the time domain of the signal differ from one signal to another (this often holds even with signals which shares the same label) it is clearly not possible to use a dataset of arrays representative of the signal in the time domain as training set. Same reasoning apply to the Fourier transformation of the signal in the time domain.

In order to use both the information contained in time and in the frequency representation of the signal, it is possible to resort to the usage of the *spectrogram* of the signal, i.e. to the type of representation which depicts the change of frequency with respect to the time.

This spectrogram can be visualized as a matrix, with shape varying from signal to signal, but upon which it is possible to execute some operations to both reduce its dimensions and make its shape homogeneous among signals.

This procedure for sure takes on, at first, reducing the spectrogram as a grid of N cells. Then, a new summarized version of the spectrogram, A of shape $N \times M$ is is such that

$$A_{i,j} = f(S_{i,j})$$

Where f is a properly selected function applied upon the element in the original S. This f, along with the values of N and M, could be considered as a first hyperparamether of the model.

a) Choosing statistical measures of each cell : the f function could be chosen to be the function that, given a grid of $a_1 \cdot a_2$ values, computes the mean and the standard deviation of these $a_1 \cdot a_2$ values.

b) Choosing metrics related to the measures of each cell: the f function could be chosen to be the function that, given a grid of $a_1 \cdot a_2$ values, returns the maximum and the minimum between the these $a_1 \cdot a_2$ values.

III. The model

The models that have been tested to perform such a task were

- Random Forest Model
- SVM Model

Overall, in the sake of possibly obtaining the highest value for the metric in applying the Model to the "eval" folder, the two models have been tested. Random Forest has been shown to work indisputably better than SVM, hence a deeper phase of hyperparamether tuning has been conducted on the Random Forest solely. The hyperparamethers tested were:

- The number $2 \cdot N \times M$ of elements in the array that will be the single data point in the training dataset.
- The function f applied to each cell in the spectrogram that returns each element in the datapoint.
- The number N_{est} of trees in the Random Forest
- The criterium for the split of the Decision Tree in the Forest.

After having trained and tested a Random Forest for each one of the possible combination of the previous hyperparamether the optimal set of hyperparamethers has been shown to be:

f	mean and standard deviation
(N, M)	(8,4)
N_{est}	500
splitting criterium	*gini*

IV. Performance assessment

With the above specified set of hyperparamethers the F1 score on the test set is 93.97%

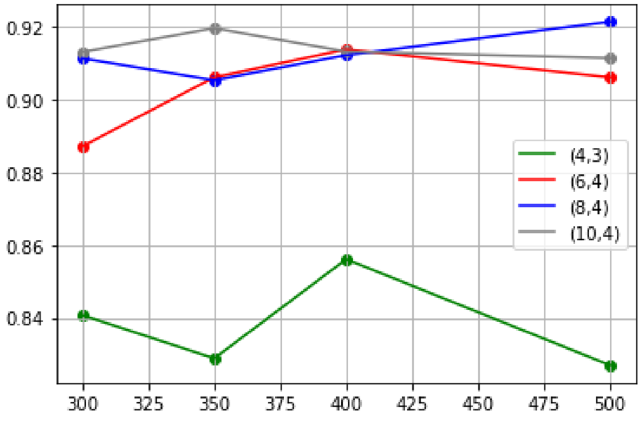

Fig. 1. F1-score values as a function of N_{est}