Voice Sculptor:
A tool for improving public speaking abilities

Analizador de Voz:

Una herramienta para aprender a hablar en público

UNIVERSIDAD COMPLUTENSE MADRID

Trabajo de fin de grado del Grado en Ingeniería Informática

Facultad de Informática

Autores

Guillermo Ovejero Sánchez

Miguel Ferreras Chumillas

Directores

Borja Manero Iglesias

Jaime Sánchez Hernandez

Curso 2020–2021

Analizador de Voz

Una herramienta para aprender a hablar en público

Memoria que se presenta para el Trabajo de Fin de Grado

Guillermo Ovejero y Miguel Ferreras

Dirigido por Borja Manero Iglesias[1] y Jaime Sánchez Hernandez[2]

[1]**Departamento de Ingeniería del Software e Inteligencia Artificial**

[2]**Departamento de Sistemas Informáticos y Computación**

Facultad de Informática

Universidad Complutense de Madrid

Madrid, 2021

Abstract

Speech is the main way we have to communicate to other people what we want, feel or think. However, and despite the importance that this has to society, there is a large part of the population without enough preparation to speak in public or with fear of doing so.

There are different ways to improve a speech, be it the structure, the message, or the vocabulary used. This work focuses on the sound qualities of the voice, since they can be measured and analyzed by software.

The outcome of this project is a tool capable of extracting measurable parameters from the voice - pitch, intensity, speed and pauses - quantifying and drawing relations between these metrics and providing real time data.

To demonstrate the applicability of our software we have carried out two experiments based on the analysis of the speech of two types of speakers, on one hand, professionals used to public speaking and, on the other, oratory students. In both cases, our tool has proved to be effective in precisely extracting voice properties and variations that determine the quality of the speech.

Keywords

Audio, Real Time, Analysis, Voice, Praat, Parselmouth

Resumen

El discurso es el medio principal del que disponemos para comunicar a otras personas lo que queremos, sentimos o pensamos. Sin embargo, y a pesar de su importancia en la sociedad, hay una gran parte de la población sin la preparación suficiente para hablar en público o con miedo a hacerlo.

Existen diferentes formas de mejorar un discurso, ya sea la estructura, el mensaje o el vocabulario utilizado. Este trabajo se centra en las cualidades sonoras de la voz, ya que se pueden medir y analizar a través de software.

El resultado del proyecto es una herramienta capaz de extraer parámetros medibles de la voz - tono, intensidad, velocidad y pausas - cuantificando y trazando relaciones entre estas métricas y proporcionando los datos en tiempo real. Estos datos se presentan en una interfaz de usuario que los visualiza en un formato legible, dirigido a profesionales del discurso.

Para demostrar la aplicabilidad de nuestro software, hemos realizado dos experimentos basados en el análisis del discurso de dos tipos de oradores: por un lado profesionales acostumbrados a hablar en público y por otro estudiantes de oratoria. En ambos casos, nuestra herramienta se ha demostrado eficaz para extraer de forma precisa las propiedades de la voz y las variaciones que determinan la calidad del discurso.

Palabras clave

Audio, Tiempo Real, Analisis, Voz, Praat, Parselmouth

Glossary

amplitude the maximum displacement or distance moved by a point on a vibrating body or wave measured from its equilibrium position. It is equal to one-half the length of the vibration path. The amplitude of sound is experienced as the loudness of sound.

decibels or dB for short is a unit to measure the intensity of a sound or the power level of an electrical signal by comparing it with a given level on a logarithmic scale.

formants each of several prominent bands of frequency that determine the phonetic quality of a vowel.

frequency is the rate per second of a vibration constituting a sound wave, frequency is represented by cycles per seconds and it's measured in Hertz (Hz).

intensity the variation of the energy flux produced by the acoustic perturbation.

paralanguage is the non-lexical component of communication by speech, for example intonation, pitch and speed of speaking, hesitation noises, gesture, and facial expression.

paralinguistic the study of the paralanguage.

pitch is the quality of a sound governed by the frequencies producing it; the degree of highness or lowness of a tone that can be perceived by humans .

sound pressure level is the ratio of the absolute sound pressure against a reference

level of sound in the air.

speech rate is the speed at which a person speaks, it can be measured in words per minute or syllables per second.

syllable is a unit of pronunciation having one vowel sound, with or without surrounding consonants, forming the whole or a part of a word.

timbre is the perceived quality of a sound that makes distinctive a particular voice or instrument.

tone a musical or vocal sound with reference to its pitch, quality, and strength.

vocal cue any meaningful variation in the sound of the voice during talk. These include: vocal qualifiers (rate, rhythm, duration, pitch, tone, articulation, loudness, pauses); vocalizations; and vocal characterizers (laughing, crying, yawning, coughing, and so on).

wav waveform Audio file format.

Sobre TEFLONX

Teflon X(cc0 1.0(documentación) MIT(código))es una plantilla de LaTeX creada por David Pacios Izquierdo con fecha de Enero de 2018. Con atribuciones de uso CC0.

Esta plantilla fue desarrollada para facilitar la creación de documentación profesional para Trabajos de Fin de Grado, Trabajos de Fin de Máster o Doctorados. La versión usada es la X

V:X Overleaf V2 with XeLaTeX, margin 1in, bib

Contacto

Autor: David Pacios Izquiero

Correo: dpacios@ucm.es

ASCII: ascii@ucm.es

Despacho 110 - Facultad de Informática

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Public speaking is an essential soft skill in both personal and professional life. Nowadays, from small meetings to weekly reports or presentations at big conferences, people need to communicate with each other. There are always times when it will be necessary to speak in public. Being able to do it in the best possible way is a skill that will help to success. In addition, being able to communicate in a confident and effective manner gives visibility to promote careers and is an advantage in recruitment processes [10, 52].

The interpersonal skills acquired from oral presentations are also useful outside work, giving the person the opportunity to actively participate in the community by taking the role of a leader or just by helping him to voice his ideas [23].

Delivering an effective speech involves more than choosing the right words. Being able to empathize with your audience using your voice and body language is essential to get the message along [44]. That is why paralinguistic has become very important in the understanding of the speech.

Paralanguage is a wide term that can accept many definitions. Trager divide it in voice set, voice quality and vocalization [43].

Voice set refers to those characteristics of the voice inherent to the speaker because of his age, sex, etc. These qualities affect to timbre, natural pitch height, or volume of the voice. Laver and Trudgill [21] characterize these voice features as “informative” but not

"communicative".

Voice qualities include speech variables which characterize how a speaker's voice adapts to situational factors. Trager classifies them as follows:

- Pitch range and control (spread or narrowed, as in monotone speech)
- Vocal lip control (from hoarseness to openness)
- Glottis control (sharp or smooth transitions)
- Articulatory control (forceful vs. relaxed speech)
- Rhythm control (smooth or jerky)
- Resonance (from resonant to thin)
- Tempo (increasing or decreasing)

There are vocalization noises that are specifically identifiable and do not belong to the general background characteristics of the speech, such as crying or laughing [49].

A standard speaker can heavily influence the confidence he is showing by knowing how to use the voice qualities to his advantage. Confidence can usually be expressed by increasing the loudness of voice and using a fast rate of speech, along with infrequent, short pauses. Under some conditions, a higher pitch and some energy fluctuations can also help this purpose [39]. Conveying confidence is important because, during oral communication, listeners try to decode vocal cues [47] to determine how the speaker is feeling, and how sure he is about his statements. Therefore, being able to show confidence will unconsciously make listeners give more credibility to what they hear [17].

A speaker who shows self-confidence is more likely to convince his audience, being in many cases the paralanguage, and not really the content of the message, what makes another person listen and support an idea [45].

In this TFG we are focusing on the importance of voice paralanguage for the success of the speech, as voice is the most empathic tool humans have [18] and paralanguage heavily determine how persuasive a speech can be [45].

1.1 Motivation

Today, practically all of us need to speak in public several times throughout our life. We are always talking and telling stories to people of different ages and all experts agree that if we do not give a good speech, listeners will stop paying attention to what we say to them.

In schools, public speaking is often relegated to the background and writing receives more attention, being much more studied and practiced. Only in very few cases, schools actively teach techniques that improve our abilities to speak in public. This is the reason why, in many cases, people are afraid to give an oral presentation, but feel comfortable writing an essay.

Our main motivation for carrying out this project has been to help people improve their speech and understand why they usually fail to engage with their audience. The objective is to propose an educational alternative to the lack of study on this form of communication, while we take the opportunity to deepen our knowledge in the fields of public speaking and audio in general.

1.2 Goals

The project main goal is to develop a tool capable of measuring the qualities of the voice during a speech, providing feedback to the speaker in real-time.

Sub-objectives of the project:

- Study the qualities of the voice, to include those that could be measured with algorithms and to be able of analyzing the voice main features. (pitch [30], intensity [14], speech rate, pauses).
- Apply the tool to measure changes in the voice of a group of students after an oratory course.
- Be able to detect typical errors in a speaker's paralanguage.
- Create a report with all the mentioned measurements.

1.3 Document structure

This document is divided into 5 chapters. The first chapter is dedicated to introducing the topic and presenting the motivation and context of the project.

In the second chapter, we explain the different audio features we can find in a speech and we review some of the existing applications and tools for audio analysis.

The third chapter deals with the process we went through to create the application, showing as well the tools we used.

In the fourth chapter, we detail the experimentation carried out with the tool we developed, from the preparation of the experiments to the results they produced.

The fifth chapter shows the conclusions we have reached after the completion of the project and refers to future work that could be done in our application to improve it.

The sixth and last chapter depicts our individual contributions to the project.

Finally, the appendix covers an installation method and a user guide for the application.

Chapter 2

State of the art

Here will be discussed different software and tools that are relevant for this project.

2.1 Software to analyze speech

In this section it will be talked about similar software that tries to achieve one or more of our goals. Another point that will be discussed are the similarities and differences between what has been achieved and this software.

2.1.1 LikeSo App

LikeSo is a mobile app designed to analyze a speech and give a feedback for improving it. It can analyze speeches up to 30 minutes long and the report it gives is focused on the filler words and pace of speech (see Figure 2.1). It also stores all your reports and grades, to show your progress. This app is only available for iOS [22].

Figure 2.1: Like So App

This app doesn't feature a real time option and the free option is very limited. It ships a feature for giving a score to a talk but the documentation on why that score is given is not clear and don't take in consideration the number of pauses or duration of them. Our app gives more information about the speech but doesn't give a score to the user.

2.1.2 Orai App

Orai is a mobile application similar to the previous one, but with added features. It gives the user feedback about the filler words an pace, but in addition takes into account the intensity of the voice, the conciseness of the speech which is calculated in part from the number of words that are repeated. On top of that allows to record a video to analyze your facial expression while speaking [29], something out of scope for our project.

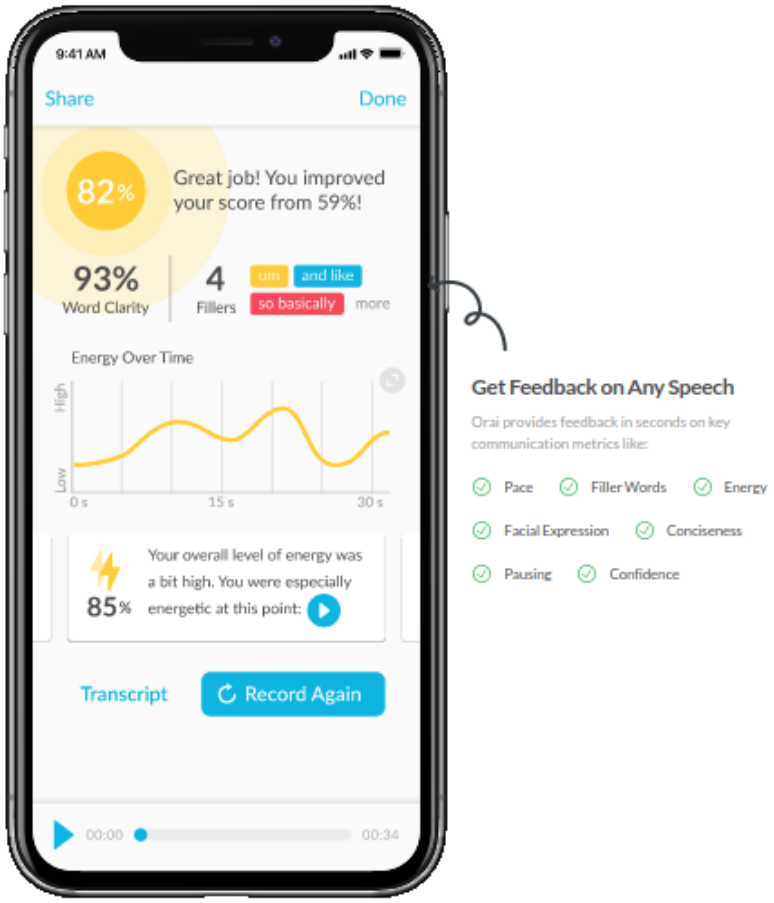


Figure 2.2: Orai App

This app is more complete in terms of the different metrics it gives. But as well as the others lacks the support for a real time analysis of the voice. The report it gives (see Figure 2.2) is very similar of what can be achieve with our app but more stylized and with better explanations and what is good or bad and why. Additionally it comes with a filler word recognizer.

2.1.3 Voice Analyst App

This application is the only one in this section that can give a feedback in real time [41]. It shows the pitch level and volume of the user while speaking, and some metrics related to them, as the maximum, minimum, average and range (see Figure 2.3).

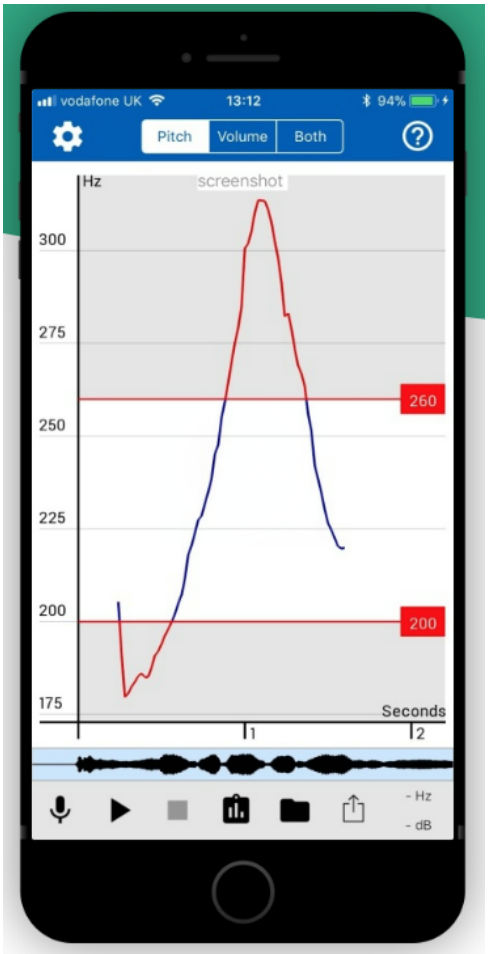


Figure 2.3: Caption

This app comes also with a commercial license. The app only measures pitch and intensity. According to their web page, it's more focused on medical purposes rather than oratory. These app is similar to what we achieved but with less metrics to give.

2.2 Audio and speech

In this section we will introduce different audio features from speech that will be useful up ahead. The ones described in the video from Emma Rodero [37] are the most important for speech. This features also have in common that can be measured with computer software and aren't subjective.

2.2.1 Intensity

When we refer to intensity, we refer at the measurements or changes in sound pressure levels measured in decibels (dB SPL) [30, 33] as this is the way Praat measures this feature.

A definition given by SCENIHR is [38]:

> One parameter of the acoustic (sound) wave which is generally used to assess sound exposure to humans is the sound pressure level expressed in μPa or Pa. Human ear audible sound pressure levels range from 20 μPa (hearing threshold) till 20 Pa (pain threshold), resulting in the scale 1:10,000,000. Since using such a large scale is not practical, a logarithmic scale in decibels (dB) was introduced which is also in agreement with physiological and psychological hearing sensations.

As we can see in Figure 2.4, the sound intensity that Praat shows is given in dB (shown below in green), this level of dB correlates with the amplitude [5] of the sound wave (shown above with both channels of the waveform), the broader the signal the higher the intensity captured is.

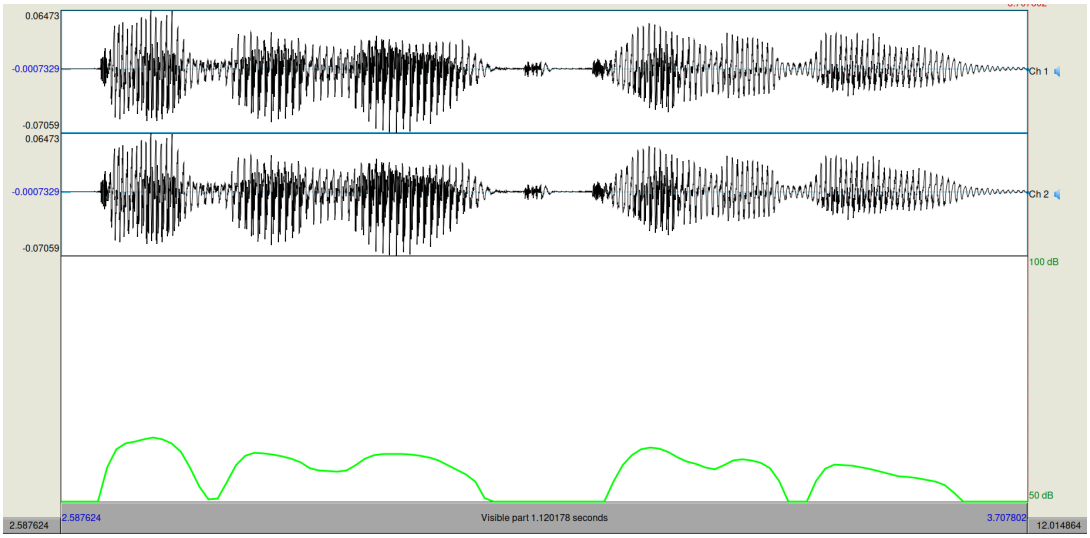


Figure 2.4: Intensity extracted from Praat

2.2.2 Pitch

Pitch is one of the main properties of sound that can be perceived. It is the sound quality most related to frequency which it's measured in Hertz (Hz). 1 Hertz equals 1 cycle per second. On average people can hear frequencies between 20 and 20.000 Hz [42]. Humans, on average have a Fundamental frequency (f_0) (also called tone) around 100 to 120 Hz for men and around 200 to 220 Hz for women [12].

Pitch as defined in Britanica [6]:

> "Pitch, in speech, the relative highness or lowness of a tone as perceived by the ear, which depends on the number of vibrations per second produced by the vocal cords."

Pitch and frequency are both related to each other but not in a linear way as we can see in Figure 2.5 (frequency plotted in logarithmic scale) frequencies between 20 and 1000 Hz increases faster than for values greater than 1000 Hz. Pitch is the feature that makes humans perceive sound as 'high' or 'low' [50].

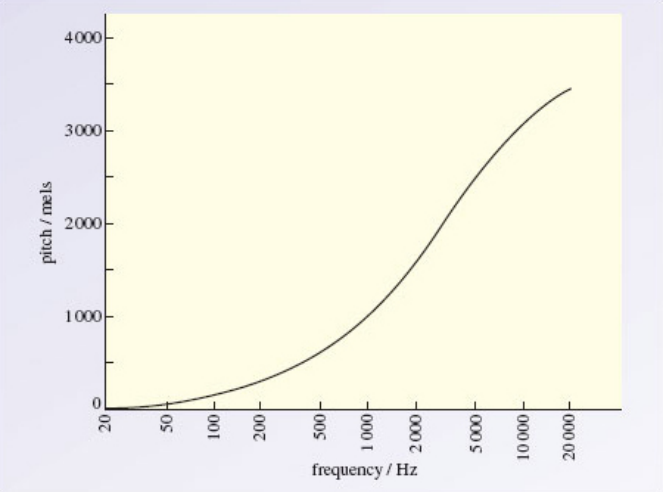


Figure 2.5: Pitch vs Frequency [42]

The meaning of 'high' and 'low' is relative to the times per second the wave oscillates (see Figure 2.6). The more slow it vibrates, the 'lower' it will sound.

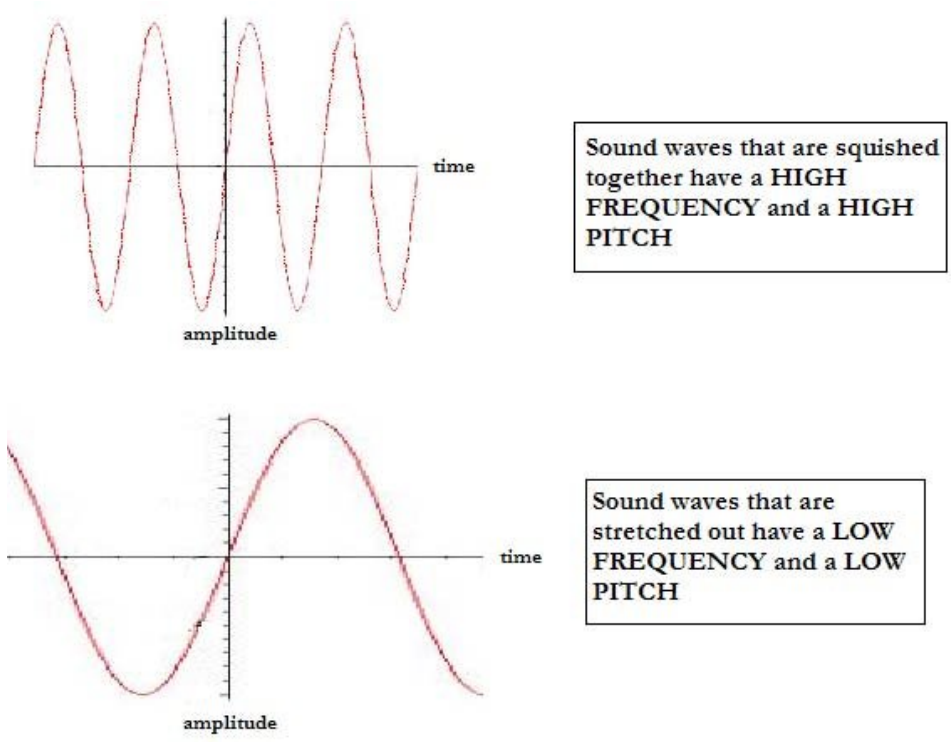


Figure 2.6: Pitch comparison

From now on, we will use the terms pitch or tone interchangeably, as meaning to say fundamental frequency (f_0). For what regards to this project given that the metric that most interest us is the range of fundamental frequencies the speaker uses in a speech and not the absolute value it has.

2.2.3 Harmonics

Harmonics are a series of frequencies derived from the fundamental frequency. An harmonic is a positive integer multiple of f_0. This fundamental frequency is the loudest one and also the one with the lowest frequency [48]

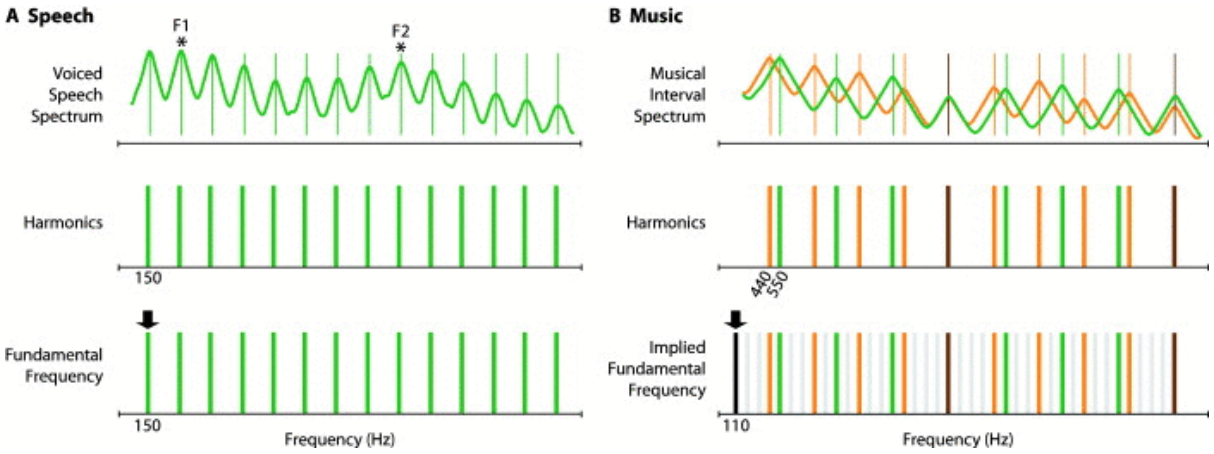


Figure 2.7: Harmonics (Daniel Bowling, 2010)

In both of the examples from the Figure 2.7 we see a sound spectrum and below that the harmonics that form that sound. Marked with a black arrow can be seen the fundamental frequency that starts at 150Hz (in case of Figure A), next harmonics by definition are a multiple of the fundamental, so next harmonics would continue 300Hz, 450Hz, 600Hz and so on. In addition it can be seen the first two harmonic peaks (F1 and F2).

2.2.4 Formants

A definition of formant [30] would be the one in the Praat guide [51]:

> A formant is a concentration of acoustic energy around a particular frequency in the speech wave. There are several formants, each at a different frequency, roughly one in each 1000Hz band. Or, to put it differently, formants occur at roughly 1000Hz intervals. Each formant corresponds to a resonance in the vocal tract.

Formants come from the vocal tract and depending on where and how the vocal tract resonates (tongue, larynx) the vibration produced by these will be different. In result of that it will make a different frequency for the formants [48]. Distinguishable formants usually range from F_1 to F_4, but some vowels also expand up to F_6, these formants, as the quote above points out, goes in roughly 1000Hz bands.

As we can see in Figure 2.8, there are six distinguishable formants bands across the audio,

the ones pointed out are F_1. Above this dark band there are several formants above, each represented by a dark band.

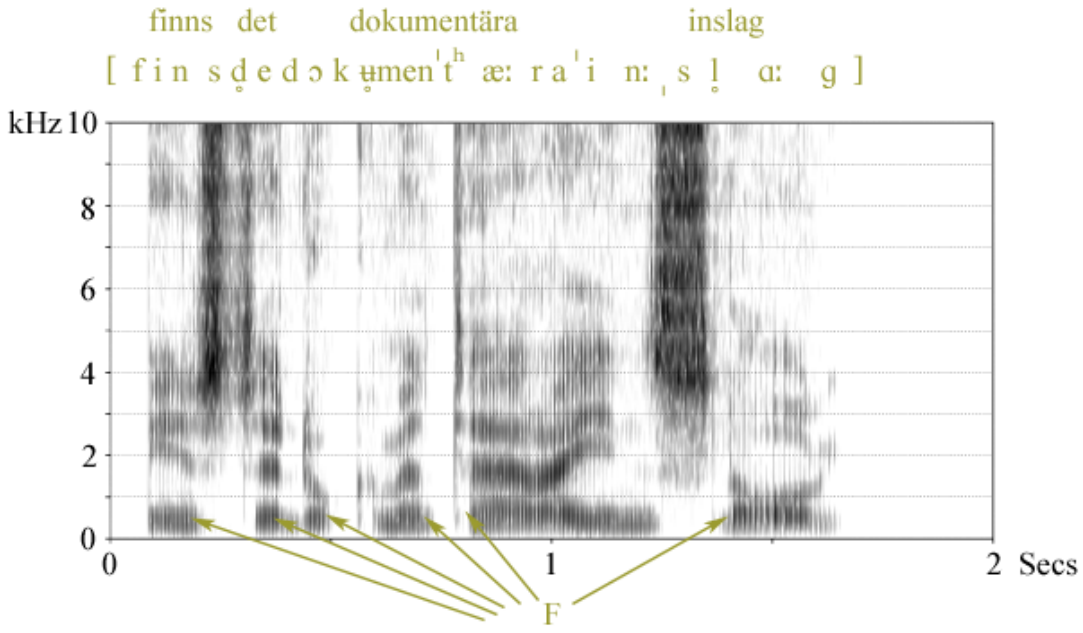


Figure 2.8: Spectrogram of formants (Praat)

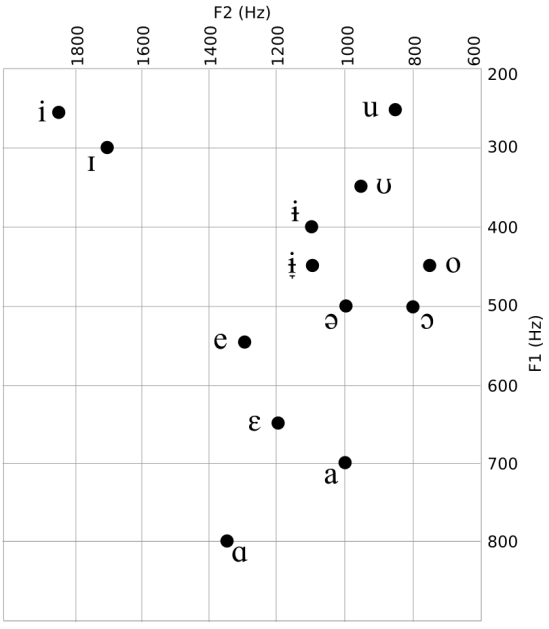


Figure 2.9: Wales vowels formants

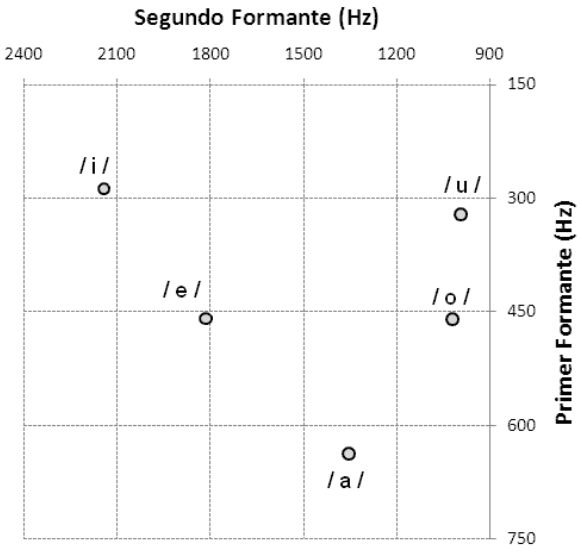


Figure 2.10: Spanish vowels formants

Differences between vowels formants between spoken languages are common [3] mainly

due to differences in prosodic characteristics [7, 26] also called suprasegmental. Some of these characteristics are stress, tone, or rhythm.

2.2.5 Timbre

Timbre as defined in the Merriam Webster dictionary [25]:

> the quality given to a sound by its overtones: such as
> **a :** the resonance by which the ear recognizes and identifies a voiced speech sound
> **b :** the quality of tone distinctive of a particular singing voice or musical instrument

Timbre is also one of the four essential qualities of voiced sound. Timbre can't be measured because it's composed of different sound features (formants, harmonics, tone). Timbre or sound quality is not the same from one instrument to another, the sound emitted from a guitar at 440Hz and from a trumpet playing the same note is different and we can perceive these changes.This principle also applies to the human voice and is how we can differentiate one voice from another [20].

2.3 Software

This section is dedicated to reviewing some of the software tools, applications and libraries that exist in the market, and that have been useful in the creation of this TFG or have been used as a model.

2.3.1 Tools to analyze speech attributes

Praat and Parselmouth

Praat [4] is a software used for the analysis of the voice. It allows to record voice or import prerecorded audio files, and analyze different features of it such as:
spectral analysis (spectrograms), pitch analysis, formant analysis, intensity analysis and also graphical representations.

Praat also have more complex features for analyzing different voice characteristics. Some

of them can be automatize for more complex analysis of the voice. As Praat library can be accessed by making calls directly to the code, Parselmouth, a Python library currently under development, aims to provide a complete and *pythonic* interface to the internal Praat code. This library accesses directly to Praat's C\C++ code, which means it gets the exact same results with its functions and also with great performance since it calls C code [16, 31].

Praat was used in first stages of development and research to give us better understanding of how audio features were measured and what results to expect when using Parselmouth in Python. For every feature that we aimed to get, Parselmouth provided a way to get it via Python objects, additionally there are some scripts that were originally written in Praat code, that were translated to Python for ease of use.

The example in the Figure 2.11 shows an example from a radio locution, there can be seen different data points.

In red, the formants analysis each line representing from the bottom-up the formants F_1 to F_4.

In green we have the intensity measured in decibels of the portion of the sound, as it can be seen the intensity correlates with the amplitude of the wave, when there is no vibration, the intensity drops.

Finally in blue we got the pitch (zoomed in in Figure 2.12), measured in hertz, this line also correlates with the wave, the higher the pitch, the wave is more compressed.

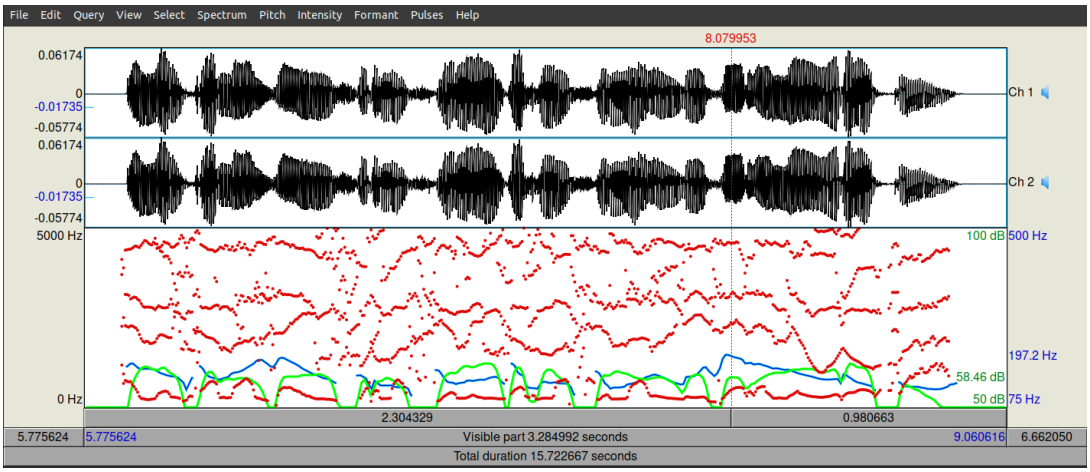


Figure 2.11: Praat Example

If we measure the period a wave takes to complete, we can get its frequency.

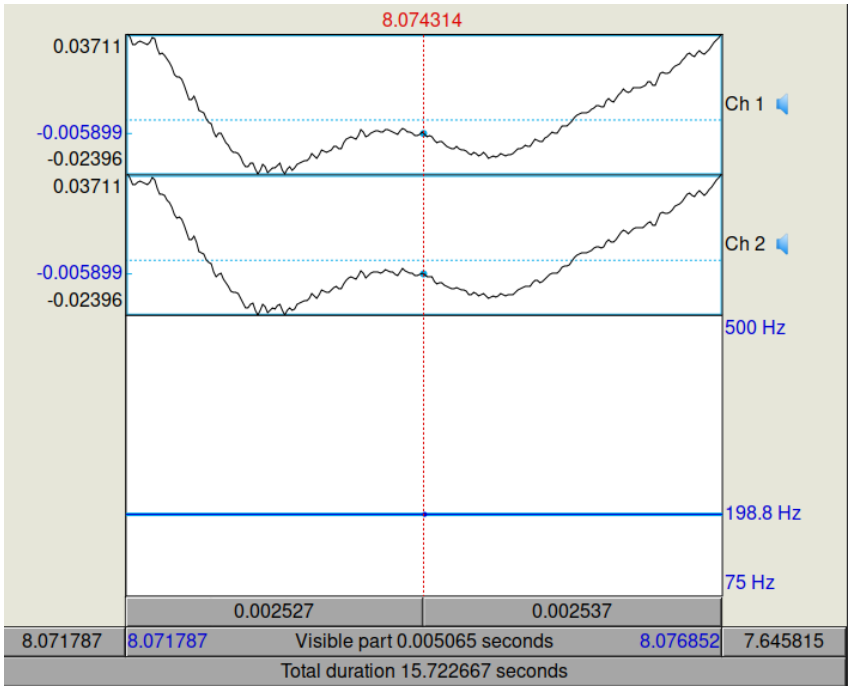


Figure 2.12: Praat Pitch

Praat also has a programming language called Praat Scripts [35], which could be extended to add more functionality with C or C++. These scripts can be called outside the context of Praat to integrate with another tool. A Python library (Parselmouth) ports these Praat functionalities. This library has been helpful to extract the audio features for posterior transformations and analysis.

To extract information about syllables [24], pauses and speech rates we used a script that retrieves it by using Praat function calls. This script was transcribed to Python in 2019 by David Feinberg [13, 27]. This script has been modified by us to extract the information in the format we wanted.

Sonic Visualizer and Vamp Aubio Plugin

Sonic Visualiser [2, 40] is a free, open-source application for Windows, Linux and Mac, designed as an audio visualization tool. It offers a configurable detailed visualisation, analysis, and annotation of audio recordings. This application also comes with a variety of plugins that could be installed with Vamp, being Aubio one of those, that allows Sonic Visualizer to extract information about pitch, intensity for interactive audio anal-

ysis.

As we can see in Figure 2.13, the application shows in a clear way information about the frequencies that are being captured, obtaining in some case at the beginning of a sentence an harmonic (marked in green) of the fundamental frequency.

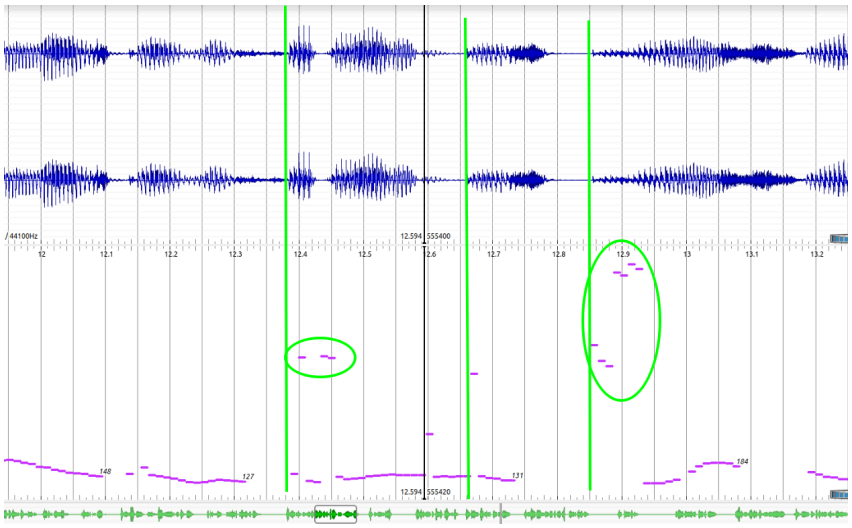


Figure 2.13: Using Sonic Visualizer to capture Harmonics

PortAudio and PyAudio

These tools follow a similar philosophy as Praat and Parselmouth. PortAudio being a library written purely in C and C++ for efficiency and portability and PyAudio being an interface of PortAudio for Python programmers.

> "PortAudio is a free, cross-platform, open-source, audio I/O library. It lets you write simple audio programs in C or C++ that will compile and run on many platforms including Windows, Macintosh OS X, and Unix (OSS/ALSA). It is intended to promote the exchange of audio software between developers on different platforms." [8]

> "PyAudio provides Python bindings for PortAudio, the cross-platform audio I/O library. With PyAudio, you can easily use Python to play and record audio on a variety of platforms." [32]

The main reason to use PyAudio, was that was compatible with different OS and had a clear interface and examples for our use case. There are other PortAudio bindings for

Python that work perfectly fine like 'sounddevice'.

NumPy and SciPy

NumPy and SciPy [28, 46] are both Python libraries for scientific computing. NumPy provides a multidimensional array object, various derived objects. It also provides 'routines for fast operations on arrays, including mathematical, logical, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations and much more' Making it very useful while working with audio arrays.

NumPy was used to obtain the numerical arrays from the sound objects from Parselmouth so it can be manipulated easily. Applying different transformations with NumPy itself and to make it serializable in a binary format so it could be easily transported between different processes. In regards to SciPy, it was used to apply some transformations and filters to NumPy so at the time of visualization, curves wasn't so harsh due to outliers in the processing of the audio.

2.4 Web Frameworks

2.4.1 Flask

Flask is a 'micro' Web Server Gateway Interface (WSGI) framework used for deploying web apps. It comes with a template engine called *'Jinja2'* which is used to pass parameters to the content and make the HTML dynamic when its rendered in the web browser. Flask comes with the minimal to work, but could be extended using a lot of plugins the community and the Flask team has developed, in case the app evolved and the requisites grow, this Flask app could be easily modified and extended to reach those new functionalities. For now our app will have only the core of Flask as no need for extra plugins was required (login, send emails, database storage).

2.4.2 Dash

Dash is a Python framework built on top of two JavaScript libraries, *Plotly.js* and *React.js* and a Python web framework, Flask. Dash features an enterprise version and an open source version of the library which comes with less functionality and support. Dash also

comes with some limitations due to the lack of good documentation there are some edge cases that are difficult to obtain. Dash abstract most of the code for building a web app and creating an interactive figure, so its easy and quickly to iterate and try different options. It also features a hot reload feature that reloads the app without the need to stop it. Dash was used in the develop of both reports, in the live report used not only as a visualization library, but also as a server to update the data of those visualizations.

Chapter 3

Project Implementation

In this chapter will be covered the project development and implementation of the program.

3.1 Programming language

When selecting a programming language to develop this application we picked up Python. The main reason to pick Python over anything else (R, Julia, C#) was that even there are other programming languages used for scientific purposes and with libraries to manipulate audio, Python is in fact the one with the larger community and with more documentation and libraries available for our use case. We also were familiar with Python so it was also another key point in favor of it. The last factor was that Python allows us to try different things quickly and to change and adapt to new functionalities that we wanted to add to the application.

3.2 Program structure

The structure of the code was divided into core functionalities:

- Live Report: application that serves a web server with Dash for real time analysis.
- Deferred Report: application that runs a Flask server to serve the webpage.

- Utils: having code in common with some other functionalities.
- Analysis: Jupyter Notebooks files used to try different things in a visual way.

In Figure 3.1 we can see 3 main components for the deferred report. The python component which process the audio and returns the information to the Flask server so it can render the HTML and serve it to the client.

Figure 3.1: Deferred Report Architecture

The architecture for the live report (see in Figure 3.2) is a bit different from the other one. First, the communication doesn't happen between the Flask server and the python process that records and process the audio. But rather independently, Python saves the information to storage and then the Flask server obtains the last information that was saved so it could update the view for the client.

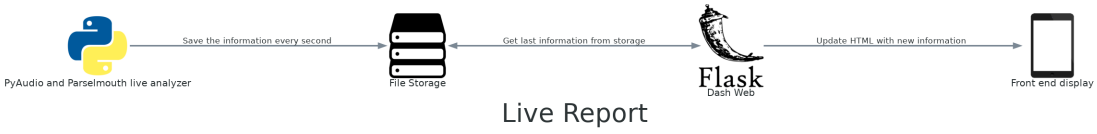


Figure 3.2: Live Report Architecture

3.3 Program development

In this section will be discussed the program development and implementation of it.

For a first draft of the program we considered a good starting point the metrics that Emma Rodero highlights in the TED talk, “Persuade con tu voz. Estrategias para sonar creíble” [37].
The four metrics Emma speaks about in the talk are the following:

a). Intensity

b). Pitch

c). Speed Rate

d). Timbre

As timbre was a difficult thing to measure and capture we substituted this metric with the number of pauses, another important metric in oratory [19]. This pauses were measured using a Praat script which will be discussed later.

The visualizations in Figure 3.3 were made in an interactive Jupyter Notebook from a record of audio of 8 seconds total. The data on these 3 graphs, was obtained with Parselmouth functions to obtain this particular data (amplitude, intensity, frequency).

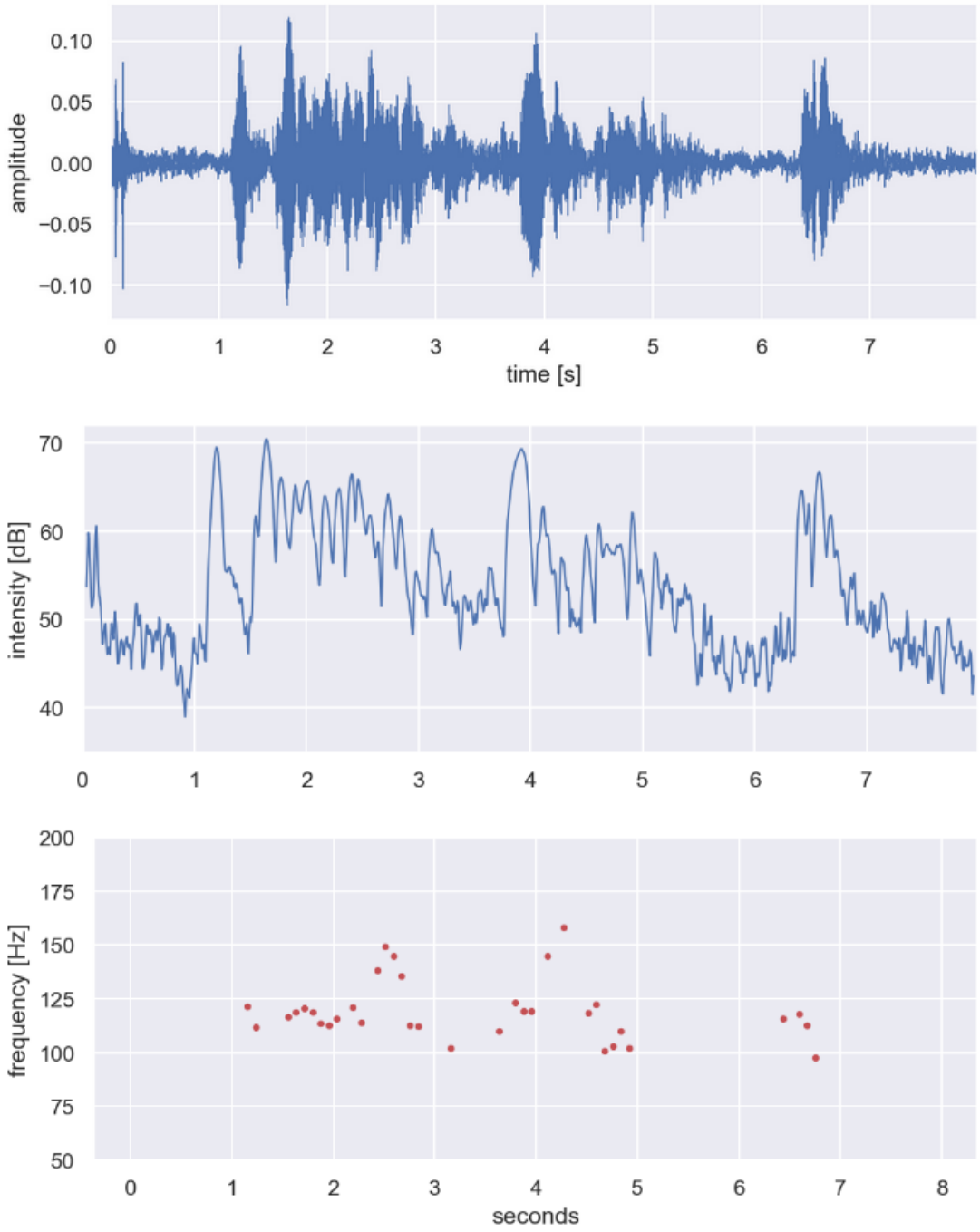


Figure 3.3: Wave amplitude, Intensity and pitch from a 8 seconds record sample

To see how much efficient Parselmouth calls are we also measured the calls to functions from Parselmouth in Figures 3.4, 3.5

```
In[1] %timeit snd = parselmouth.Sound("holamesa.wav")
170 µs ± 3.22 µs per loop (mean ± std. dev. of 7 runs, 10000 loops each)

In [2]: snd.duration
Out[2]: 1.6370068027210884

In [3]: %timeit snd.to_pitch()
9.26 ms ± 94.3 µs per loop (mean ± std. dev. of 7 runs, 100 loops each)

In [4]: %timeit snd.to_intensity()
2.08 ms ± 5.26 µs per loop (mean ± std. dev. of 7 runs, 100 loops each)
```

Figure 3.4: Performance of calls ‘to_intensity‘ and ‘to_pitch‘ from an audio with a duration of 1.63s

For different audios of different length we can see that the calls to the library for the function ‘to_pitch’ goes up in time linearly (see Figure 3.5) with an algorithmic complexity of $O(n)$ given that n is the number of bytes in an audio file.

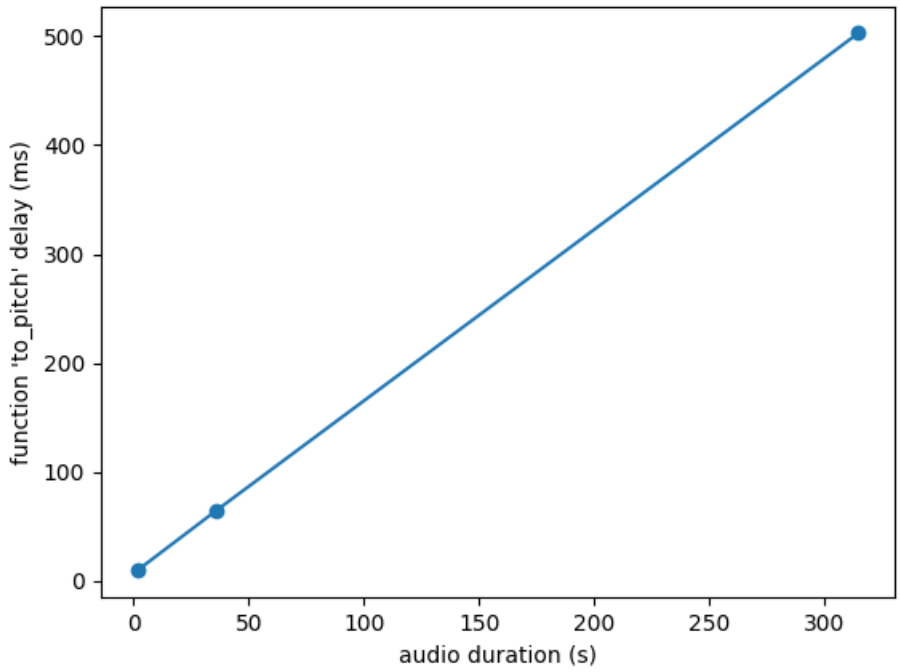


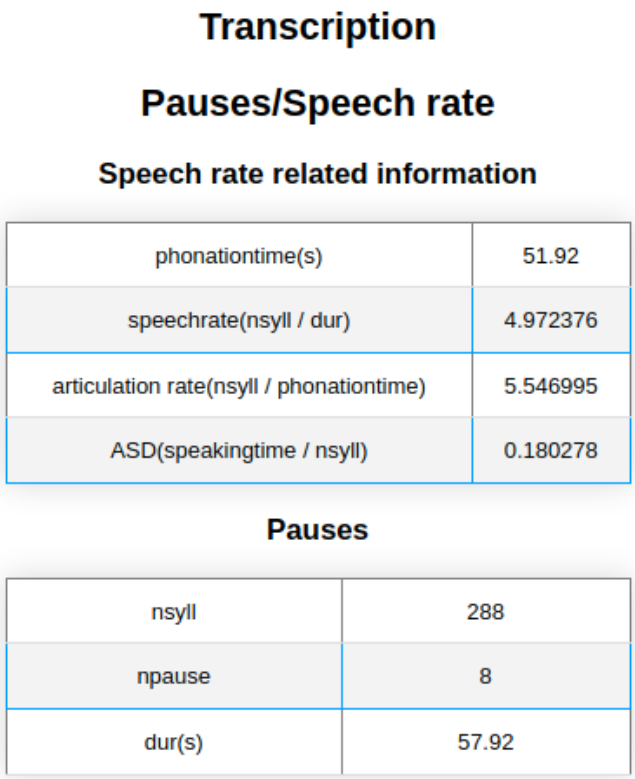
Figure 3.5: Performance of calls ‘to_pitch‘ with audios of different duration

3.4 Deferred report

Although the main goal was to get a real time tool that could give feedback to the user, a Deferred report of the speech was also developed, this report could fill another purpose for the user of the app which could give some feedback once the speech is done to get a retrospective view of it, as you can hear again some portions and get a deeper understanding of what went wrong. This report given the audio file it will print the statistics and graphs of the audio the user passed as parameters to the application.

This report is created in a web server that returns a HTML page that could be highly customized and served via the internet, it was done this way because this could be consumed as a free Software as a Service (SaaS) on the internet so the user doesn't even have to install anything on their computer and just uploading an audio file to get the report of the speech in a few seconds.

This report was built on top of the Flask web server framework with a template in HTML that would be rendered using the template engine that comes with Flask, Jinja2. This rendered HTML prints some tables from a JSON returned from the 'Syllable Nuclei' scripts (see Figure 3.6) and also plot two graphs from the data contained in NumPy arrays. These plots are interactive in a way that pitch and intensity could be zoomed in and out and also panned through the whole length of the audio.



Transcription

Pauses/Speech rate

Speech rate related information

phonationtime(s)	51.92
speechrate(nsyll / dur)	4.972376
articulation rate(nsyll / phonationtime)	5.546995
ASD(speakingtime / nsyll)	0.180278

Pauses

nsyll	288
npause	8
dur(s)	57.92

Figure 3.6: Table with audio information

As a static graph wasn't easy to get insights from it, a more appropriate approach was to plot this graphs in an interactive way, for this we make use of Dash graphing capabilities to plot on the web the intensity and pitch. As Dash is built on top of Flask it haves an easy integration with the Flask web server.

Figures 3.7 and 3.8 are made interactive with Dash, as it has Javascript libraries for dynamic plotting, so in the first figure (3.7) the whole audio intensity could be seen, but to see more clear some portions of audio it could be selected and zoomed in like in Figure 3.8

Figure 3.7: Intensity plot (Dash)

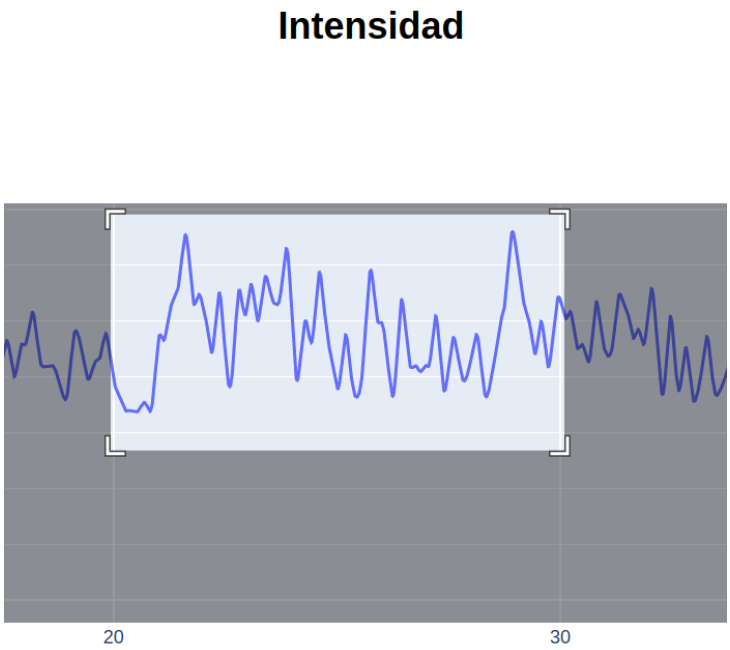


Figure 3.8: Intensity plot, selecting a specific time window to visualize it clearly

Another important feature was to be able to playback the speech in the same report. This was made in the template including an HTML tag for audio that is capable of playing audio in the web browser.

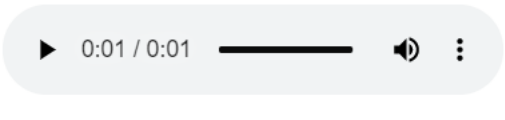


Figure 3.9: Audio Controller

3.4.1 Transcription

Exists several methods of audio transcription (also called *Speech to Text*). Here we will list some of them and discuss the use of one over the others.

As different methods exist we will focus on the ones that uses machine learning models to predict the text, as nowadays these methods give the most accurate results. We distinguished them between two different types. Cloud based or in-house.
An existing python library that have Speech to Text models or using a cloud service from AWS, Azure or IBM. Since we wanted to get results that were close to reality but also fast and with parameters that could be configured we opted to use the cloud companies since these usually have better trained models for Speech Recognition for a variety of different languages, libraries with preloaded models usually come with a single model that has been trained with mainly English data, but since we were analyzing Spanish speakers that won't be as precise as a Spanish trained model. This cloud service from 'IBM Watson Speech to Text' also offers a free tier to try with 500 minutes per month of Speech to Text computations. The IBM Speech to Text also comes with different parameters to tune the service. One of this parameters calculates a timestamp for each word for better audio labeling.

In Figure 3.10 we can see a full example of what IBM offers in this service. It can detect multiple speakers in an audio and add word timings.

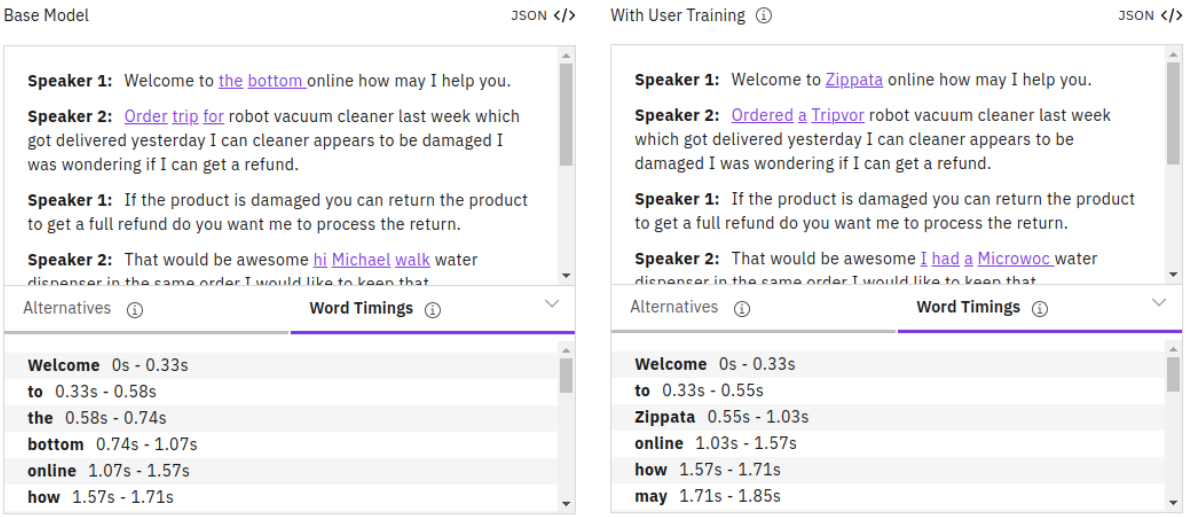


Figure 3.10: IBM Speech to Text Demo

3.5 Live audio streaming

In the previous sections all the audio analysis were realized with static audio files previously recorded, this couldn't be used in a real time scenario were we want to see changes in the discourse at the moment they occurred.

For this purpose we develop a script that opens a stream of audio, obtaining the sound in bytes from an input device (microphone), this input device could be changed to be any different audio device from the computer, a thing that was useful later on in the experimentation part as we wanted to playback the different audios to analyze to the application without losing any quality. The rest of the live report is based on this script that records audio in an asynchronous way so that the process does not block and saves it to a circular buffer that hold up a specific number of seconds of audio information in it.

3.6 Live report

For the live report we used real-time audio analysis and extract statistics of the voice, To later display this information dynamically in a web page, to develop this we make use

of the audio streaming script for obtaining the information in the last N seconds. The number of seconds could be changed to any number, but a buffer of around 10 to 15 seconds works better for performance and visualization of the information. For the part of extracting information we also used Parselmouth as it is the best and fastest option to obtain all this metrics. Next will be discussed the form of communication between both process and the configuration to output data in real time to the browser.

3.6.1 Inter process communication

Here will be discussed the development of the communication between the server (audio streaming) and the client (web dashboard).

To make inter process communication between the two process, a straightforward strategy was used, due to the limitations of performance being in the rendering of the information and not being capable of updating the view faster than once per second, we opted for a simpler option that allows for better debugging and understanding of the code. That is to communicate with files, in binary and JSON file formats. Also in this way there isn't a need to synchronize both process since one will write data and the client will get the newest data available at that time, in case some information is lost during this process the application would still work as it will update with the last information that was received. As the server could work forever appending new data, a mechanism to erase old and unused data from the file its used, so the hard drive doesn't fill up with garbage.

3.6.2 Dash Components

The Dash components that make up the report are of two different types:

- Graph figures
- Indicator figure

The component used to display the intensity and pitch was the Graph figure.
This graph updates with the audio information received, it contains information about intensity and pitch over time. Both of them are displayed using the same format and at the same time, that is achieved by sharing the same callback to update both graphs.

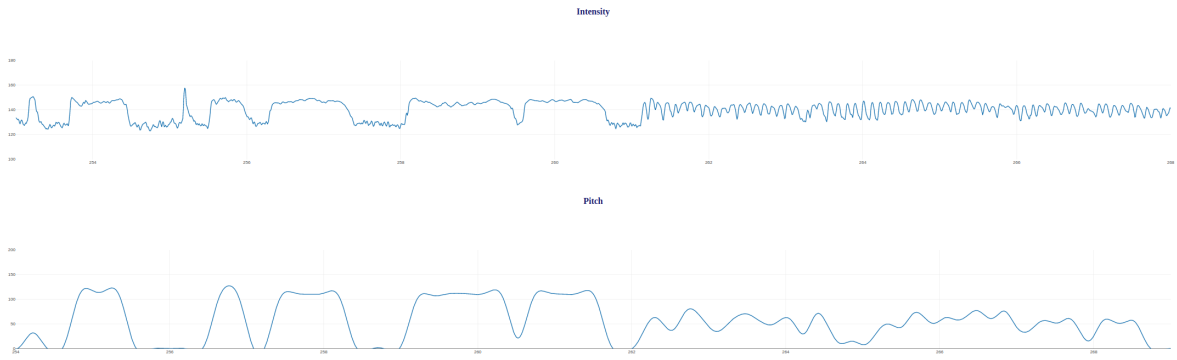


Figure 3.11: Intensity and pitch interactive plots

To display information about pauses, syllables and speech rate, we used the Indicator component. In the case of speech rate this component was customized to display a gauge (see Figure 3.12) in the other ones a number with just the number being indicated on.

In the Figure 3.12, we can see different visual points that stand up, first there is an approximated indicator of what a good speech rate should be (drawed as a range in green between the values 3.5 and 5), this is for the user to have a sense of what is good or bad in terms of speech rate.

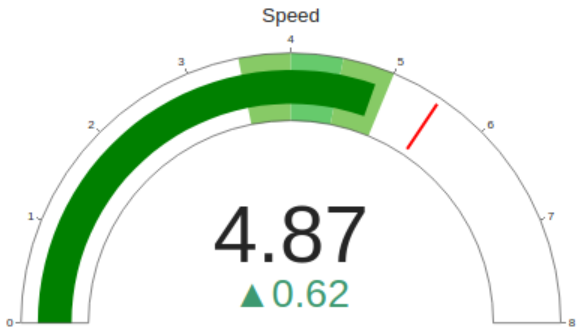


Figure 3.12: Dash gauge of speed, measured in syllables per seconds

Both of this types of figures update dynamically and react to the changes in the data is being displayed. To update this figures the server updates it via dash callbacks. These

callbacks have some limitations due the nature of the application. Dash callbacks done in the server side can only be updated at a minimum of 1 frame per second, the server could spit information much faster but due to this limitation the updates could be made once per second, which is a reasonable rate for this application. These callbacks returns the figures that can be customized with code and are displayed with help of the JavaScript libraries that Dash uses underneath.

Chapter 4

Experimentation

To see the performance of our application with real data and see its capabilities when analyzing speeches, we have conducted two experiments.

The first one consists of the analysis of recorded speeches from famous people recognized as great speakers. With this experiment, starting from objectively good discourses that we have extracted from videos, we want to see if our tool is able to recognize the speech quality and to provide data to confirm this premise. If this is the case, we also want to see how these speeches fit the canon of what would be the "perfect speech"

In the second experiment, we recorded and analyzed a set of speeches from a group of students before and after an oratory course. With this data we want to see if there is a change in the use of vocal cues by students, and if our application is able to reflect it.

4.1 Analyzing great speakers

4.1.1 Methodology

Participants

To conduct this experiment we used speeches from renowned experts in the art of public speaking.

When selecting participants, the first person chosen was Emma Rodero due to her importance to our project and her research on public speaking. The rest of the participants were recommended by our tutors.

The following speeches were chosen from the corresponding participants.

Speaker	**Speech**
Emma Rodero	Persuade con tu voz. Estrategias para sonar creíble [37]
Blanca Portillo	El teatro nos enseña que equivocarse no es un drama [34]
Mario Alonso Puig	En todo ser humano hay grandeza [1]
Edgar Cabanas	Las claves para vender la felicidad [9]
Hernan Casciari	Mi hija quiere entender el sistema financiero [11]

Table 4.1: Selected participants for the first experiment

Emma Rodero is a recognized expert in the field of public speaking, with a Ph.D. in communication and psychology, and two master's degrees in Pathology of Voice and Psychology of Cognition.

Blanca Portillo is a theater and film actress with several awards in her career. Thanks to this she is very used to modulate her voice to transmit whatever she wants to an audience.

Mario Alonso Puig is a doctor currently dedicated to research and teaching. He has also been giving courses, conferences and coaching in both national and international companies for more than 20 years.

Edgar Cabanas is a researcher and teacher in the field of psychology, who has given and participated in numerous conferences, seminars and dissemination podcasts.

Hernan Casciari is an Argentine writer who since 2012 has been reading his stories on radio, television and theater.

Experimental design

For this first experiment we wanted to try the accuracy of the real time analysis of our application. To achieve it, we used a computer with Linux Mint 20.2 Cinnamon as the operating system installed. We downloaded a tool called Pavucontrol to change the input that our computer received, and with the audio output as the new input device, we ran our application and started visualizing the videos one by one.

For the analysis, we chose two random minutes from each speech. In each one, the first fifteen seconds were dedicated to filling the buffer of our application and the next 45 to record the variation of the data in real time.

Once the results were recorded, we compared them with the standard of what would be a good speech. To obtain this standard, we relied on the research of Emma Rodero, where she defines the ideal parameters as follows [37]:

- Intensity: High, without shouting, around 40 to 50 dB
- Pitch: A low pitch, this is relative because by nature humans have different pitches (women normally has higher pitched voice than men)
- Speed: A relative good speed would be around 160 to 180 wpm. Which translates to around 4 to 4.5 syllables per second (since in Spanish a word has a mean of around 1.7 syllables [15]).

Another important factor for the effectiveness of the speech is the number and duration of the pauses, since pauses give time to the audience to assimilate what has just been said. However, if the pauses consist on a filler word or they are too long, they can distract the audience from the speech development. In a study from E. Rodero [36], a good number of pauses would be around 10 per minute, yet this study was conducted with English speakers during a radio bulletin. We would usually see more pauses per minute in a common speech and we should also take into account the differences in languages. Thus, Spanish speakers tend to speak faster and to pause more during speech.

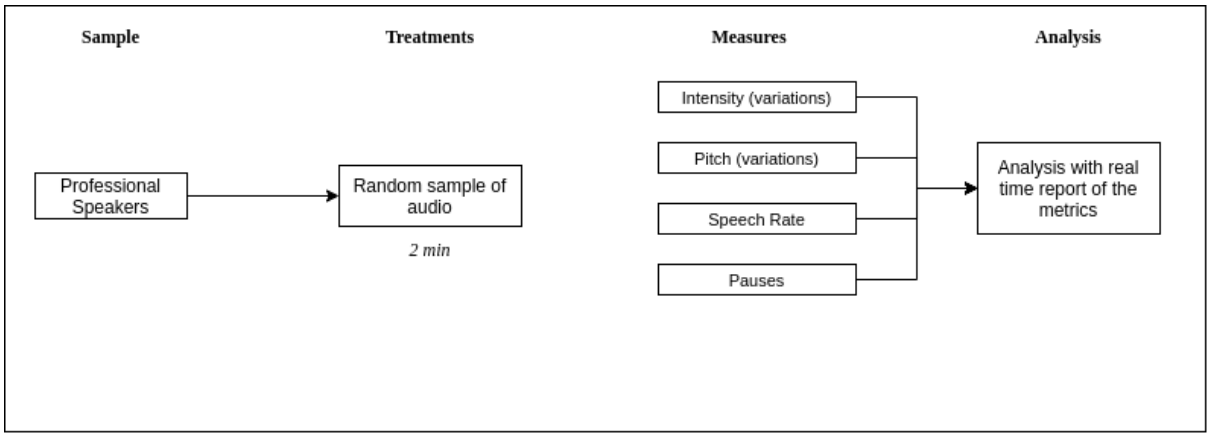


Figure 4.1: Experiment Design of 'Anylyzing great speakers'

4.1.2 Results

Emma Rodero

Speech Video: Persuade con tu voz. Estrategias para sonar creíble
The following minutes from the video were selected to be analyzed: 2:00 - 3:00 and 6:00-7:00

During the two minutes recorded, Emma Rodero maintains a constant speed, which drops only when she relies on the visual support she has. Thus, the speech rate remains constantly between four and four and a half points, only decreasing to three and a half when there is a longer pause. Besides, throughout the talk, the speaker relies very little on pauses, keeping them between one and three every 15 seconds.

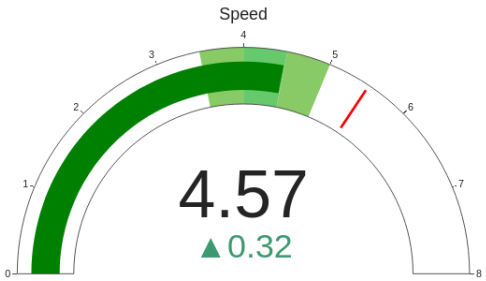


Figure 4.2: Emma Rodero speed sample

When we look at the pitch we can see a perfect use of pauses to finish the sentences, as shown in the figure below, in addition to a very wide range of values.

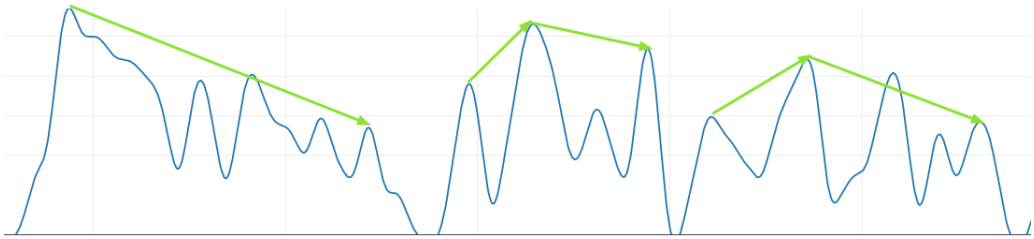


Figure 4.3: Emma Rodero pitch sample from the first minute

In the second minute, Rodero speaks about the pitch. There is a very characteristic part in which, to represent how different pitchs affect the speech, she changes her own while talking, first too high, and then too low. The following figure depicts the pitch shown in the application during this part.

In the image you can clearly see three stages, the one on the left corresponding to when the pitch begins to increase, the one in the middle representing the part with the lowest pitch, and the one on the right the speaker's normal voice.

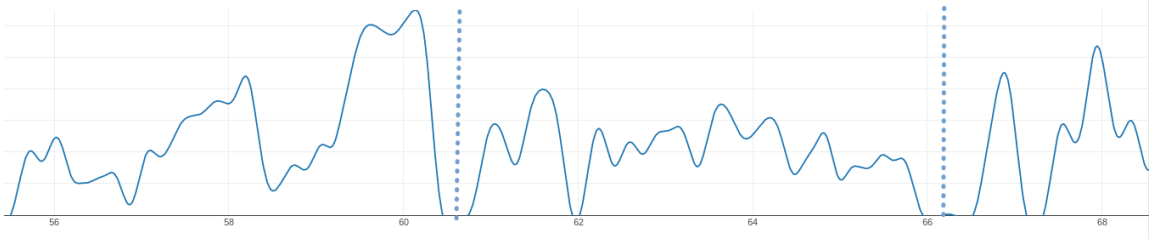


Figure 4.4: Emma Rodero pitch sample from the second minute

As the data shows, and as expected, this speech fits perfectly into the standards, maintaining a constant and adequate speed, with not too many pauses and a perfect use of the pitch.

Regarding the intensity, we can see that the results shown by our application are not well calibrated, since according to the data obtained the intensity exceeds 200db at some point, something that clearly does not happen. On the other hand, and despite the fact that the absolute value is not correct, the variation can correspond to reality.

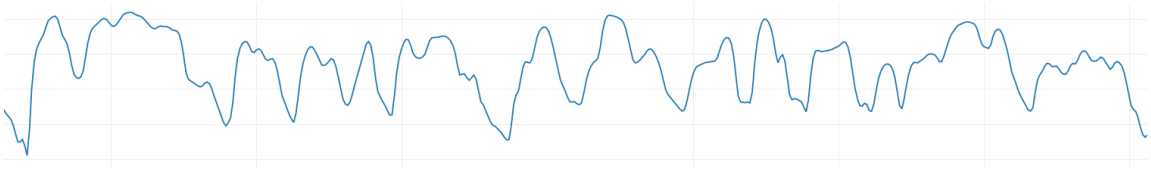


Figure 4.5: Emma Rodero intensity sample

Blanca Portillo

Speech Video: El teatro nos enseña que equivocarse no es un drama

The following minutes from the video were selected to be analyzed: 1:00 - 2:00 and 3:00-4:00

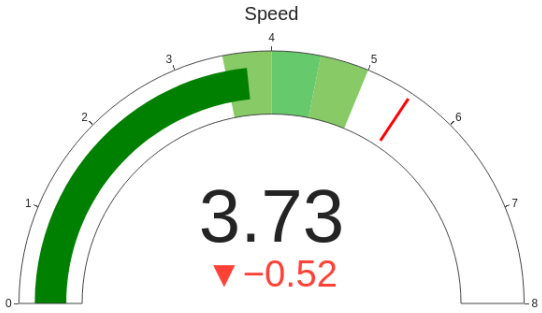


Figure 4.6: Blanca Portillo speech rate sample

If we listen to this speech we can see that the speaker talks calmly and is not afraid to pause. This is reflected in our application, which during the two minutes analyzed shows a number of pauses that varies between three and six, and a speech rate between three and four. If we compare the speed with the canon, it is slightly slower than it should be. However, when talking about anecdotes and personal experiences, the speaker prioritizes a calm and slow tone that makes the message clear, compared to a faster one that sounds more confident.

In the following figures we can see samples of the pitch extracted from the two minutes analyzed.

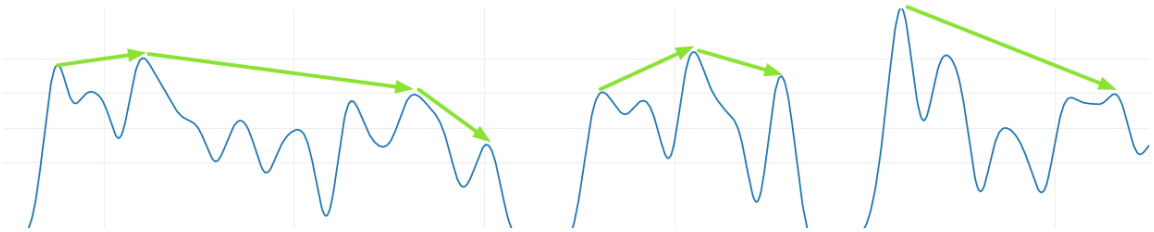


Figure 4.7: Blanca Portillo pitch sample from the first minute

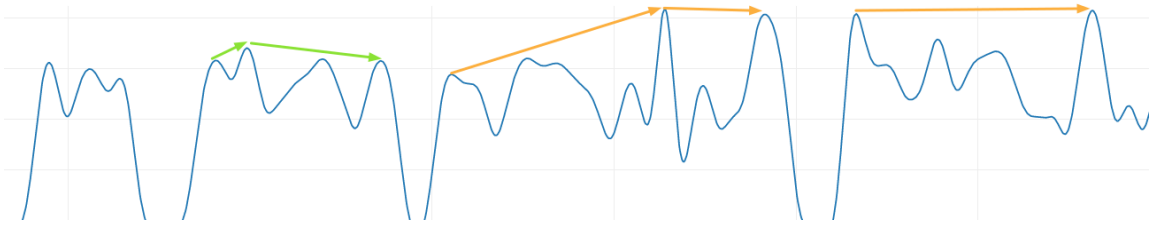


Figure 4.8: Blanca Portillo pitch sample from the second minute

As the data shows us, the first minute meets the standards when it comes to dealing with pauses, ending each sentence in a low pitch. However, in the second minute, we can see how the speech begins to deviate from the standard and the sentences start to end in the same pitch in which they begin or even in a higher one.

At the beginning of the second minute, when Portillo is recalling an anecdote, one of her sentences ends on a much higher pitch than the one she started with, making the listener feel that the sentence, followed by an uncomfortable pause, is not over. When we look at our application, this is perfectly reflected as shown in the following figure.

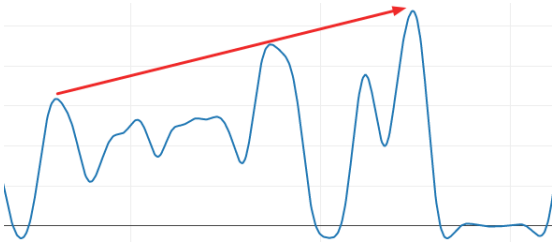


Figure 4.9: Blanca Portillo high pitch at the end of a sentence

Mario Alonso Puig

Speech Video: En todo ser humano hay grandeza

The following minutes from the video were selected to be analyzed: 19:00 - 20:00 and

38:00-39:00

In general, during the speech, Alonso Puig maintains an adequate speed although this varies greatly depending on what he is telling, he also introduces a large number of pauses that increase the variation. He has an average speech rate of around three and four and a half, and his pauses have values between two and six although these vary greatly.

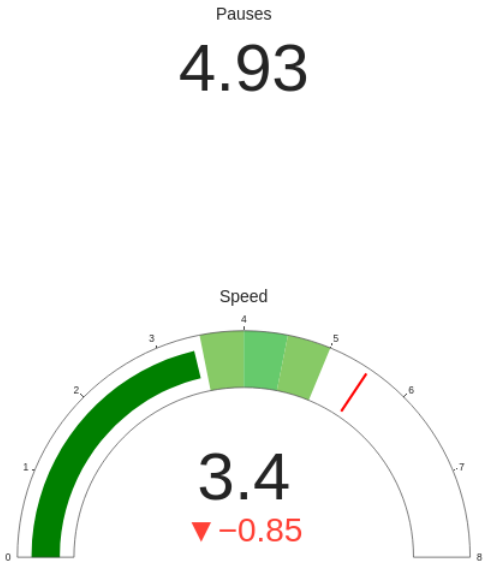


Figure 4.10: Mario Alonso Speed sample

If we look at the pitch, in the first figure we can see a curious technique that is quite far from the canon. It occurs on several occasions during the speech, and consists of using a large number of short-medium pauses preceded by sentences that end in a high pitch. On the contrary, in the second figure we can see that the speaker also adapts his speech to the standards, using less pauses and finishing the sentences in a low pitch.

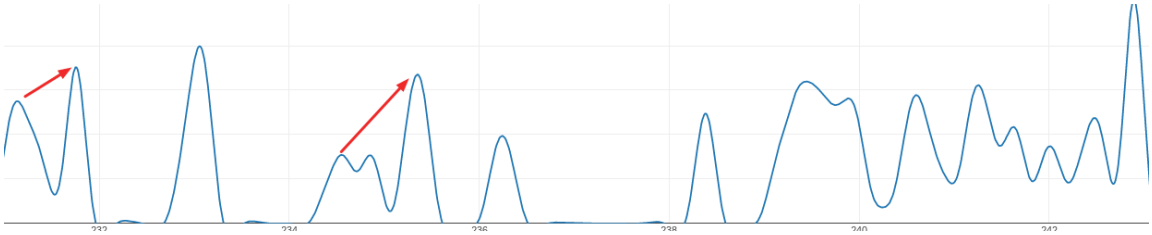


Figure 4.11: Mario Alonso pitch sample from the first minute

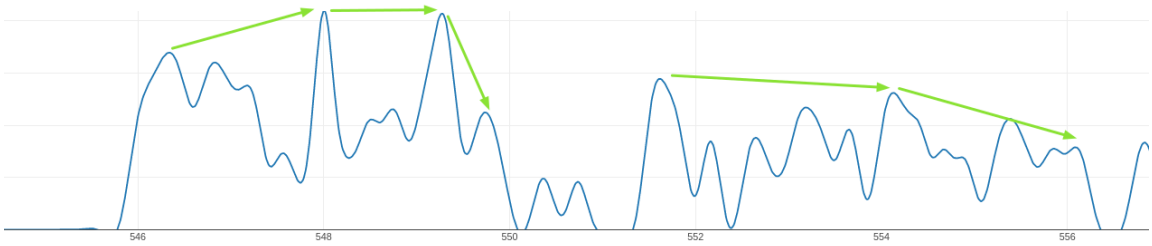


Figure 4.12: Mario Alonso pitch sample from the second minute

Edgar Cabanas

Speech Video: Las claves para vender la felicidad

The following minutes from the video were selected to be analyzed: 4:00 - 5:00 and 15:00-16:00

In its form, this is a very similar speech to the one of Mario Alonso Puig. During the minutes analyzed, the average speech rate is highly variable and it fluctuates between three and a half and four and a half points due to the large use of pauses. The recorded pauses also vary between two and seven.

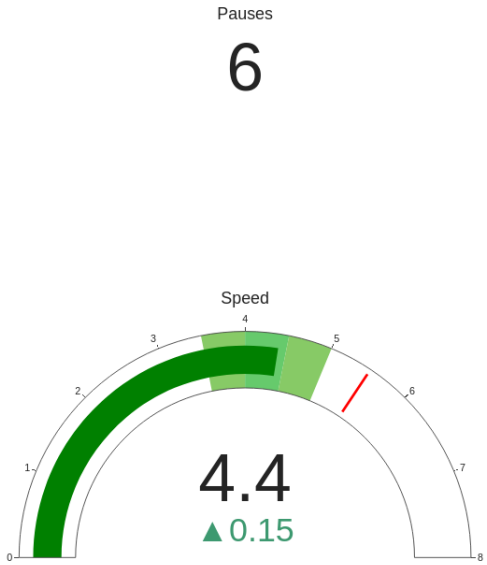


Figure 4.13: Edgar Cabanas speed sample

When we look at the pitch, the similarities with Puig's speech are clear. Cabanas also

uses a large number of pauses – shorter, though - finishing the previous sentence in a high pitch. And, as Puig, he does not use this technique during the whole speech but, as we can see in both the first and second figures, most of the time he adapts to the standards.

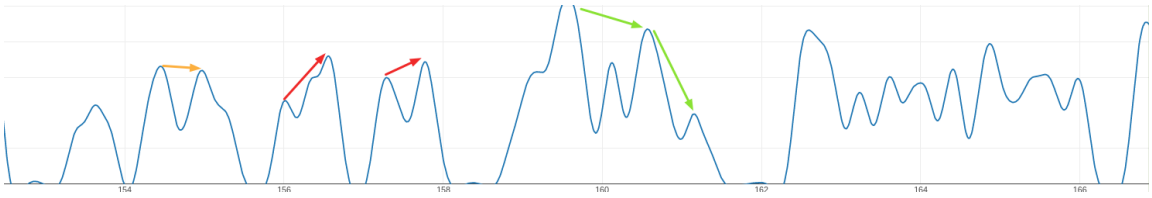


Figure 4.14: Edgar Cabanas pitch sample from the first minute

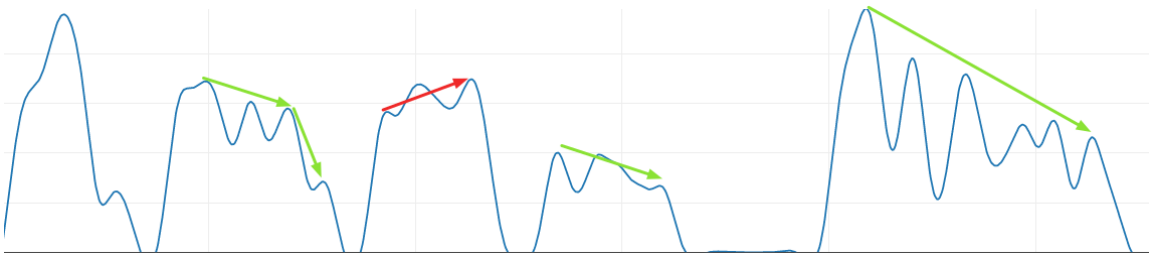


Figure 4.15: Edgar Cabanas pitch sample from the second minute

In the figure below we are able to see the real time intensity our application show us. During the two minutes analyzed there were no mayor changes in the speaker intensity, who maintained a regular voice level.

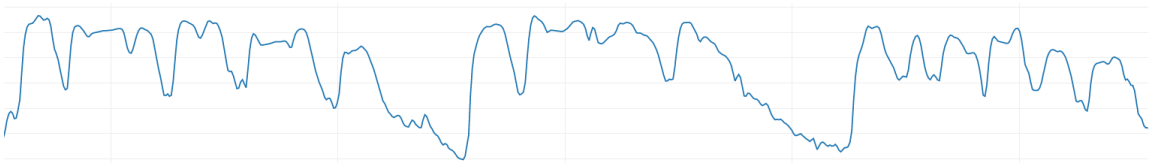


Figure 4.16: Edgar Cabanas intensity sample

Hernan Casciari

Speech Video: Mi hija quiere entender el sistema financiero

The following minutes from the video were selected to be analyzed: 4:00 - 5:00 and 13:00-14:00

In this speech Casari uses a slow rhythm, with a speech rate that varies between three and four point five due to the use of pauses. The latter are frequently used to emphasize the sentences.

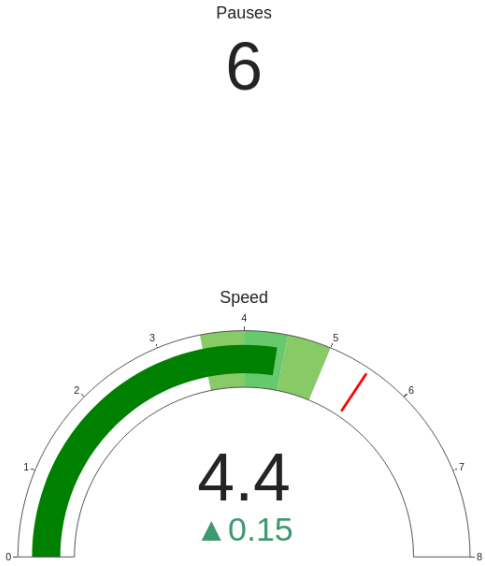


Figure 4.17: Hernan Casciari speed sample

In the analysis we can see how the speech does not fully adapt to the standards. The speaker often leaves a high pitch before a short-medium pause to get the listener's attention and then he finishes the concept with a sentence ended in a low pitch. This can be clearly heard in the video and is represented in the figures below.

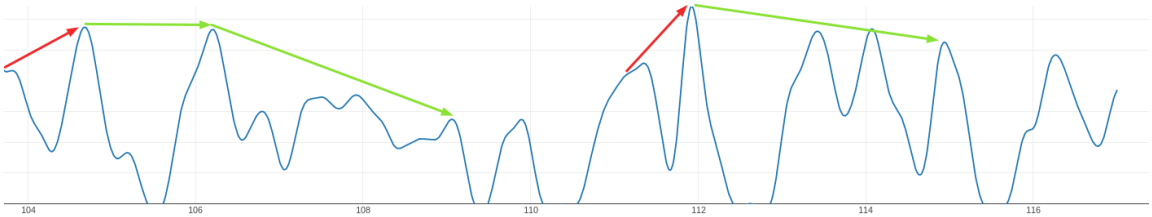


Figure 4.18: Hernan Casciari pitch sample from the first minute

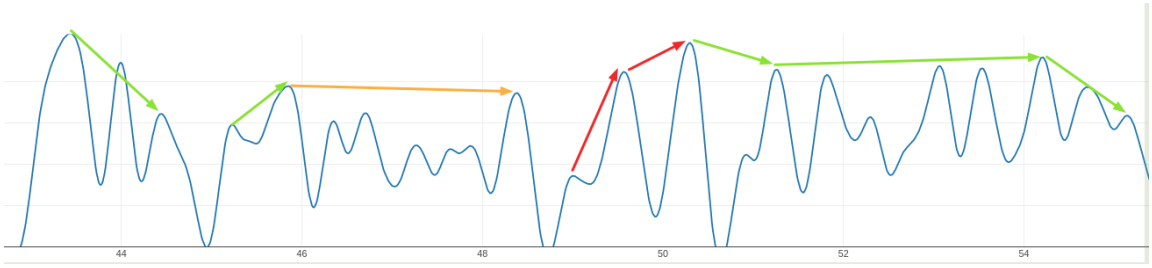


Figure 4.19: Hernan Casciari pitch sample from the second minute

4.2 Analyzing students from an oratory course

4.2.1 Methodology

Participants

In this experiment we had the participation of eleven students from different degrees within the Complutense University of Madrid, nine women and two men. All of them were young people under the age of thirty.

Experimental design

For this experiment we had access to a three-day oratory course at the Complutense University of Madrid, where we recorded two speeches from each participant.

The first recordings correspond to the first speech given by the students at the beginning of the course on a given topic. The topic was about the reason why they considered themselves good candidates for a certain position at the university administration. To prepare this topic, the students had ten minutes.

In order not to influence the speech or make the participants nervous we were not allowed to attend the event, so the recordings were given to us by the course professor. The recordings were made of both video and audio, being the audio the one collected with the camera's microphone.

The second recording session corresponded to the final speeches of the course's students. For these speeches the students chose their topics and whether or not they wanted a

graphic support. In addition, they had a full day to prepare themselves.

This session was divided into two days and we were able to attend both. For the recordings, we got a lapel microphone connected to the camera that allowed us to get a better audio.

Even though the microphone used was different from the first batch of recordings to the second, the audio quality was good enough to compare them. Even so, while analyzing, we gave less importance to the intensity and the number of pauses because the background noise of the first recordings could slightly adulterate the results.

Finally, listening to the recordings, we saw that one of the speeches of the second batch, due to its goal, contained a lot of noise, which made it difficult to analyze. After the analysis was done, no valid data came out from this recording, so we decided not to use it as part of the experiment.

The experiment was carried out at the end with 10 participants, 8 women and 2 men. It consisted on using the deferred part of our application on the collected recordings to get all the possible data. We also created a script that calculated the averages of this data and put together in tables the first and second recording of each participant.

Separating men from women, we used data to obtain a global view of the change in participants' use of vocal cues after their oratory course.

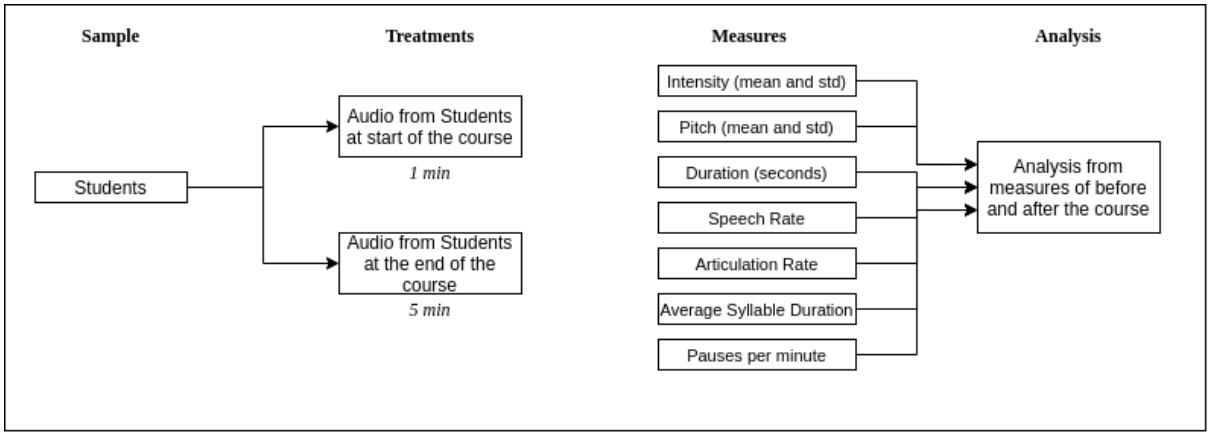


Figure 4.20: Experiment Design from 'Analyzing students from an oratory course'

4.2.2 Results

Women

Student number 1

	Before course	After course	Difference
Mean intensity	73.39	65.06	-8.33
Std intensity	4.76	11.78	7.02
Mean pitch(Hz)	177.22	166.3	-10.92
Std pitch(Hz)	113.06	141.2	28.14
Duration(seconds)	61.59	314.62	253.03
Number of pauses per minute	0.0	19.26	19.26
Speech rate(syllabus/duration)	5.29	4.25	-1.04
Articulation rate(syllabus/phonation time)	5.29	5.88	0.59
ASD(average syllabe duration)	0.19	0.17	-0.02

Table 4.2: Student number 1

Student number 2

	Before course	After course	Difference
Mean intensity	72.45	67.35	-5.1
Std intensity	5.24	13.45	8.21
Mean pitch(Hz)	139.32	130.57	-8.75
Std pitch(Hz)	109.73	123.21	13.48
Duration(seconds)	63.34	300.06	236.72
Number of pauses per minute	0.0	20.4	20.4
Speech rate(syllabus/duration)	5.04	3.32	-1.72
Articulation rate(syllabus/phonation time)	5.04	4.99	-0.05
ASD(average syllabe duration)	0.2	0.2	0

Table 4.3: Student number 2

Student number 3

	Before course	After course	Difference
Mean intensity	66.73	59.92	-6.81
Std intensity	6.5	12.44	5.94
Mean pitch(Hz)	151.03	105.67	-45.36
Std pitch(Hz)	119.18	118.12	-1.06
Duration(seconds)	60.2	292.77	232.57
Number of pauses per minute	0.0	21.93	21.93
Speech rate(syllabus/duration)	5.4	3.49	-1.91
Articulation rate(syllabus/phonation time)	5.4	5.56	0.16
ASD(average syllabe duration)	0.19	0.18	-0.01

Table 4.4: Student number 3

Student number 4

	Before course	After course	Difference
Mean intensity	71.86	66.18	-5.68
Std intensity	6.42	14.12	7.7
Mean pitch(Hz)	170.84	136.47	-34.37
Std pitch(Hz)	97.59	128.32	30.73
Duration(seconds)	63.64	374.4	310.76
Number of pauses per minute	1.89	12.66	10.77
Speech rate(syllabus/duration)	5.17	4.09	-1.08
Articulation rate(syllabus/phonation time)	5.33	5.95	0.62
ASD(average syllabe duration)	0.19	0.17	-0.02

Table 4.5: Student number 4

Student number 5

	Before course	After course	Difference
Mean intensity	69.35	66.05	-3.3
Std intensity	7.64	14.16	6.52
Mean pitch(Hz)	175	135.04	-39.96
Std pitch(Hz)	114.51	122.44	7.93
Duration(seconds)	37.82	227.78	189.96
Number of pauses per minute	4.76	18.18	13.42
Speech rate(syllabus/duration)	5.34	3.49	-1.85
Articulation rate(syllabus/phonation time)	5.66	4.92	-0.74
ASD(average syllabe duration)	0.18	0.2	0.02

Table 4.6: Student number 5

Student number 6

	Before course	After course	Difference
Mean intensity	69.55	69.22	-0.33
Std intensity	6.32	12.11	5.79
Mean pitch(Hz)	173.29	154.25	-19.04
Std pitch(Hz)	87.94	114.15	26.21
Duration(seconds)	61.65	361.92	300.27
Number of pauses per minute	0.97	11.6	10.63
Speech rate(syllabus/duration)	5.26	4.44	-0.82
Articulation rate(syllabus/phonation time)	5.31	5.37	0.06
ASD(average syllabe duration)	0.19	0.19	0

Table 4.7: Student number 6

Student number 7

	Before course	After course	Difference
Mean intensity	63.19	57.09	-6.1
Std intensity	6.65	14.26	7.61
Mean pitch(Hz)	117.11	93.09	-24.02
Std pitch(Hz)	127.21	129.75	2.54
Duration(seconds)	46.1	321.89	275.79
Number of pauses per minute	0.0	20.5	20.5
Speech rate(syllabus/duration)	3.58	2.52	-1.06
Articulation rate(syllabus/phonation time)	3.58	5.41	1.83
ASD(average syllabe duration)	0.28	0.18	-0.1

Table 4.8: Student number 7

Student number 8

	Before course	After course	Difference
Mean intensity	70.66	65.3	-5.36
Std intensity	6.8	13.59	6.79
Mean pitch(Hz)	168.23	131	-37.23
Std pitch(Hz)	100.07	133.69	33.62
Duration(seconds)	60.69	302.66	241.97
Number of pauses per minute	2.97	23.79	20.82
Speech rate(syllabus/duration)	4.96	3.54	-1.42
Articulation rate(syllabus/phonation time)	5.13	5.38	0.25
ASD(average syllabe duration)	0.19	0.19	0

Table 4.9: Student number 8

Men

Student number 9

	Before course	After course	Difference
Mean intensity	69.12	63.41	-5.71
Std intensity	3.76	11.04	7.28
Mean pitch(Hz)	52.94	66.93	13.99
Std pitch(Hz)	64.11	72.7	8.59
Duration(seconds)	54.4	316.7	262.3
Number of pauses per minute	0.0	23.68	23.68
Speech rate(syllabus/duration)	4.3	4.16	-0.14
Articulation rate(syllabus/phonation time)	4.3	5.77	1.47
ASD(average syllabe duration)	0.23	0.17	-0.06

Table 4.10: Student number 9

Student number 10

	Before course	After course	Difference
Mean intensity	69.52	63.81	-5.71
Std intensity	4.16	11.47	7.31
Mean pitch(Hz)	45.23	50.57	5.34
Std pitch(Hz)	58.45	61.17	2.72
Duration(seconds)	45.4	269.38	223.98
Number of pauses per minute	0.0	23.61	23.61
Speech rate(syllabus/duration)	3.81	3.75	-0.06
Articulation rate(syllabus/phonation time)	3.81	3.75	-0.06
ASD(average syllabe duration)	0.26	0.18	-0.08

Table 4.11: Student number 10

Although we will not consider intensity as a very relevant parameter, there is a general tendency to slightly decrease it in the second speech, except for student number 6 who

maintained the same intensity in both recordings. There is also an overall increase in standard deviation, indicating a wider use of volume levels by students.

Looking at the main pitch we can see a big difference between men and women. Men start from a pitch between two and three times lower than women and both slightly increase it from the first to the second talk. On the contrary, women decrease it between twenty and forty hertz approximately.

This difference is not found when we look at the standard deviation. In general, in this parameter we can observe an increase from the first speech to the second. This increase has highly varied values, going from two and a half to thirty-three hertz. The only exception is student number three. However, if we look at the total value and not at the comparative one, the value he achieves is very similar to the one of other students in the second speech.

This general increase implies that in the three days of the course the students learned to widen the pitch range they used, which allowed their speeches to sound less monotonous.

In the following parameters we can see the duration - which we only use to calculate the averages of other parameters -, the ASD, that barely changes, and the speed, represented by the articulation and speech rate. Due to the low precision of the pauses, this last factor leaves us with unclear data.

Chapter 5

Conclusions and future work

5.1 Conclusions

In this project we studied how paralanguage influences the art of public speaking to create an application that provides orators with real time and deferred feedback on the quality of their speech.

The experiments carried out demonstrate how our application is capable of showing enough data to understand the shape of a speech in real time.

In the first experiment, we observed how voice data collected from 'good' speeches share multiple common traits. Some of this traits can also be linked to the paralanguage standards proposed by Emma Rodero, which define the characteristics for an effective message delivery when speaking in public in Spanish language.

We could identify the following traits between professional speakers:

- They all use a faster speed when trying to transmit a emotion, and a slower with longer pauses when they want the audience to remember a specific message.
- Even though they use different strategies for the use of the pitch to confront pauses, they all resort to the standards many times during the speech.

The second experiment demonstrates how our application can provide valuable data to

track progress in public speaking. We compared voice recordings from students in an oratory course, both before starting and after concluding the course. We could observe the improvements students made just by looking at the data. This improvements correspond mainly to a wider and a more appropriated use of their voice intensity and pitch.

The experiments were carried out with certain limitations. In the first, due to the material used, we did not get a good input for certain parameters, what gave us unreal values in intensity making it not useful for the analyses. In the second experiment the background noise of the first batch of recordings altered the data relative to speeds and pauses giving us no valid data to compare.

5.2 Future Work

Our application has been able to extract from a good input source accurate metrics about the intensity, pitch, speed and pauses in a speech. The future work will consist on the following.

- Adding the script we did to analyze the reports in the experiment, to immediately get metrics like mean and standard deviation of the pitch and the intensity.
- Integrating in the real time report the transcription, focusing on the filler words and the number of them. The filler words are a good indication of whether a speech is good or not, so adding them will improve our application.
- Linking the transcription with the metrics graphs in order to better understand how certain values are related to certain sentences along the speech.
- Dividing pauses in different types, short, medium or long, depending on the duration, because each one should be confronted in a different way. Adding these categories and making our tool able to show how a person uses the different pauses can give very valuable data for the analysis.
- Create faster and smooth updates. In dash also exist client side callbacks which can be executed in intervals of less than 100 milliseconds but needed the data to be stored in the client. Implementing this will improve the real time analysis by

showing information without any delay.

- Modify the real time application interface to allow to see all the metrics at the same time, to not having to repeat the analysis to focus on different metrics.

- Repeat the experiments with a greater and more varied number of participants, and a material that allows our application to obtain all the parameters in a valid way.

We believe the tool we have designed and the data it shows can be very useful in different fields of research, applicable to multiple industries. In psychology, for instance, this application could help to understand how certain stimuli affects a person's speech. In oratory, the output data provided by the app could track a person's progress in learning public speaking skills or it could be used to get information about the quality of a speech in comparison to a certain standard. These are just some of the possibilities we have thought about but, in the future, this application could be used to develop other projects in numerous fields.

Chapter 6

Individual work

6.1 Guillermo Ovejero

I started researching for tools that could give us information about audio and more specifically the voice, there are a lot of commercial tools that make this job possible, some are open source ones, and other are paid apps. We immediately discarded the paid apps and stick with the ones that were free to use. Most of this tools also feature a lot of complex transformations that can be applied to audio, this could be an interesting feature to preprocess some of the audio files to reduce noises. But since it was going to be with real time audio. I didn't bother to research about these complex transformation that are computationally heavy and require a lot of time to finish.
The two main tools I investigated were:

- Praat
- SonicVisualizer (w/ Aubio Plugin)

These tools were used mainly to visualize data, but they have a wide range of possibilities.

The next thing I started developing was a way to obtain formants from an audio with the purpose of analysing a feature of timbre and also as a first approach of measuring speed, since if vowels could be counted, it will be easy to get the number of syllables. I recorded

myself saying the vowels and develop a script that checked, based on information about first and second formant ranges, what vowel was being pronounced. At the end we don't end up using this for the final project, but it was an interesting thing to develop and learn about.

In the opportunity we have to attend some of the Borja's oratory classes there was a lot of task to do to record the students fast and without interrupting them. As there was multiple task to do. I was in charge of assuring the camera and the microphone were recording at all times and there was no noise in the microphone due to friction with the clothes, also because I carried my laptop over there I was also in charge of saving the files once the class was over for later processing and removal of the image, so I later could upload them to have access to them when needed. These recordings were useful later on when the application was finished and the experimentation phase started.

The next thing was to build a simple web app with flask that given an audio file rendered a HTML with the metrics. The first initial version was very simple and barely have any styling with CSS. This version was iterated over to add more complex tables, style and the interactive graphs.
One of the features that was also included in the deferred report was a transcription of the text. Miguel researched the different option that were available to make this feature. He told me about the IBM one and decided to stick with it. Then I started testing with the SDK IBM has available and implemented the transcription in the report.

Finally the developing of the live report was divided in two parts, the communication process and the dashboard. I started trying with different ways to communicate with the dashboard. The first attempts using a message broker tool were very complex for the project. They are perfect tools for messaging with high performance, as in this context we didn't need that performance we tried simpler ways to communicate with process, the message broker tools I used were Redis, a In-memory database with capabilities to be used as a message broker. And RabbitMQ, a fully functional message broker that comes with more configuration and advanced options to set the brokers than Redis.
As we searched for other ways to communicate between processes I develop a couple of proof of concepts for communicating two python processes with sockets, one sending audio data and the other receiving the data from the socket port. This idea was maintained for

a bit because it worked for every OS and also was fast enough. This solution was good but a bit complex for debugging. And given the dash limitation of updates at a minimum of 1 frame per second wasn't the best option either. In the side of the dashboard development I introduced some small features like the use of the speed gauge and a play/stop button to make the interface more user friendly. Also related with the applications I made some documentation so that other developers could improve the application and a user manual, for users that would like to try these app but don't necessarily have a background in computer science.

6.2 Miguel Ferreras

When the project started, my main task was to do research on the topic, learning about audio, vocal cues, paralanguage, and the effects of all these factors on speech. This research helped us better understand both the characteristics of speech and the metrics we wanted to measure with our app.

During this time, I was also analyzing existing applications that were close to the idea we wanted to develop, in order to see what they did, how they worked and what type of errors we should avoid.

Once we got a preliminary design for our app, in addition to further research I tested different tools and programming languages to obtain the data we wanted for the voice.

The following image corresponds to a small program that I created in Unity, which showed in real time the intensity of the sound captured through a microphone.

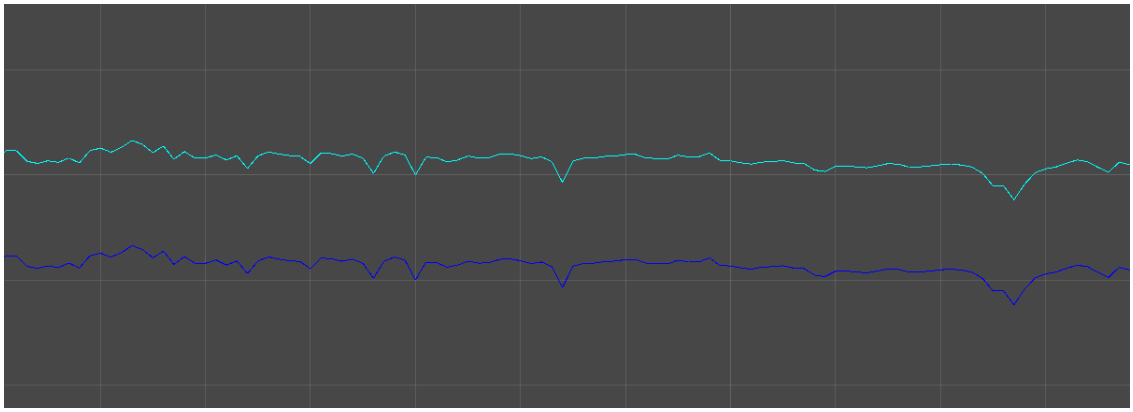


Figure 6.1: Measure of sound intensity in Unity

After initial research, and once we decided we would use Python as a programming language, I started working with Jupyter Notebook to create scripts capable of extracting basic metrics from a recorded audio. To this end, I searched the internet for audio samples that we could use to see the accuracy of our analysis. However, it was not easy to find suitable samples with only one voice and no background sound. To solve this problem, I created a small piece of code, which allowed me, thanks to the 'pyaudio' and 'wave' libraries, to record my own voice for as many seconds as I wanted and save it in a '.wav' file. This made the following process much easier, as I was able to create my own samples, allowing me to modulate my voice and get the tone I wanted.

By joining these scripts with the work done by Guillermo, we created the basic structure of our application. Then I started looking for ways to get the program work in real time as well.

Halfway through the project, and thanks to Borja Manero, we had the opportunity to attend two sessions of a public speaking course to record the students' speeches. For these recordings, I was in charge of reviewing the material and going to each session early to prepare and test it, making sure there were no problems when recording.

After the course ended, we continued with the development of our app, dividing the work of figuring out how to introduce the missing metrics. During that time, I discovered the IBM Watson tool that Guillermo used to implement the transcription of an audio in our app while I was in charge of finding a way to measure and analyze the pauses of a speech.

With the app core finished, I focused on the search for tools that would allow us to create an interface capable of incorporating data in real time, and I discovered Plotly and Dash, which we have used for the final interface.

Finally, once the application was completed, I was in charge to collect the necessary data for the two planned experiments, taking the metrics from both the YouTube videos and the audios recorded during the public speaking course we attended.

Appendix

User Manual

Installation

Windows

Preferred method to install Python is via Chocolatey, a Windows package manager. But could also be done with Anaconda

Install Python 3.8

Install Python 3.8 with preferred method

Install Dependencies

```
py -m ensurepip --upgrade
pip install wheel
pip install PyAudio-0.2.11-cp38-cp38-win_amd64
pip install -r requirements.txt
```

Unix

```
sudo apt-get install python3.8
pip install PyAudio-0.2.11-cp38-cp38-win_amd64
pip install -r requirements.txt
```

Use Guide

Deferred Report

Enter to the "deferred_report" folder and run the following command:

```
python report\_deferred.py
```

Then access to localhost in port 5000:

http://127.0.0.1:5000/

Upload a file in .wav or .mp3 format and submit it.

This will take you to another route like this:
http://127.0.0.1:5000/report?file=my_uploaded_speech.wav

Stream Analyzer

Enter to the folder "stream_analyzer" and run the following command:

```
python main.py
```

Then access to localhost in port 5050:
http://127.0.0.1:5050/

Click on the *'play'* button to start recording and start analyzing your speech.
When done hit *'stop'* and the record of that session will be saved in the *'output'* folder.

The app will pick your default microphone. If it doesn't recognize your microphone there is a script in 'utils/get_input_devices.py' that would list the different input devices.

Configuration

To change settings for the stream analyzer app enter modify the '.env' file from the 'stream_analyzer' folder and change the following values:

This are the values that work fine by default.

- SECS=15
- SAMPLE_RATE=44100
- FPS=1

Definitions of these parameters

- SECS: Changes the number of seconds the buffer saves.
- SAMPLE RATE: Number of samples obtained in one second. By default most audio apps works with 44.1kHz of sample rate.
- FPS: Frames per seconds the dashboard updates, something less that 1 seconds will not work properly.

References

[1] M. Alonso Puig. (2018). "En todo ser humano hay grandeza," Youtube, [Online]. Available: https://www.youtube.com/watch?v=f69n5VQLIQw.

[2] Aubio, *Aubio, library for audio labelling*. [Online]. Available: https://aubio.org/.

[3] D. M. Behne, "A comparison of the first and second formants of vowels common to English and French," *The Journal of the Acoustical Society of America*, vol. 89, no. 4B, p. 1918, 1991. DOI: 10.1121/1.2029505. [Online]. Available: https://doi.org/10.1121/1.2029505.

[4] P. Boersma and D. Weenink, *Praat: Doing phonetics by computer [Computer program]*, Version 6.1.38, retrieved 2 January 2021 http://www.praat.org/, 2021.

[5] Britannica, T. Editors of Encyclopaedia, "Amplitude," *The Editors of Encyclopaedia Britannica*, 1998. [Online]. Available: https://www.britannica.com/science/amplitude-physics.

[6] ——, "Pitch," *Encyclopedia Britannica*, Jul. 1998. [Online]. Available: https://www.britannica.com/topic/pitch-speech.

[7] ——, "Suprasegmental," *The Editors of Encyclopaedia Britannica*, 2020. [Online]. Available: https://www.britannica.com/topic/suprasegmental.

[8] P. Burk, *Portaudio - an open-source cross-platform audio api*. [Online]. Available: http://www.portaudio.com/.

[9] E. Cabanas. (2019). "Las claves para vender la felicidad," Youtube, [Online]. Available: https://www.youtube.com/watch?v=LWYAUSXbCfI.

[10] D. Carroll, *Skills for academic and career success*. Pearson Australia, 2014.

[11] H. Casciari. (2017). "Mi hija quiere entender el sistema financiero," Youtube, [Online]. Available: https://www.youtube.com/watch?v=HLIJkmy3vy8.

[12] DPA Microphones, *Facts about speech intelligibility*, 2021. [Online]. Available: https://www.dpamicrophones.com/mic-university/facts-about-speech-intelligibility.

[13] D. Feinberg, *Syllable Nuclei Python*, 2019. [Online]. Available: https://github.com/drfeinberg/PraatScripts/blob/master/syllable_nuclei.py.

[14] O. Fleig, M. Iida, and C. Arakawa, "Blade Tip flow and Noise Prediction by Large-Eddy Simulation in Horizontal Axis Wind Turbines," *Engineering Turbulence Modelling and Experiments 6*, pp. 689–698, 2005. DOI: 10.1016/B978-008044544-1/50066-2.

[15] J. A. Gualda Gil, *Densidad de información del español vs el inglés*, 2013. [Online]. Available: https://www.elcastellano.org/densidad-de-informaci%7B%5C'%7Bo%7D%7Dn-del-espa%7B%5C~%7Bn%7D%7Dol-vs-el-ingl%7B%5C'%7Be%7D%7Ds.

[16] Y. Jadoul, B. Thompson, and B. de Boer, "Introducing Parselmouth: A Python interface to Praat," *Journal of Phonetics*, vol. 71, pp. 1–15, 2018. DOI: https://doi.org/10.1016/j.wocn.2018.07.001.

[17] X. Jiang and M. D. Pell, "On how the brain decodes vocal cues about speaker confidence," *Cortex*, vol. 66, pp. 9–34, 2015. DOI: 10.1016/j.cortex.2015.02.002.

[18] M. W. Kraus, "Voice-only communication enhances empathic accuracy.," *American Psychologist*, vol. 72, no. 7, pp. 644–654, 2017. DOI: 10.1037/amp0000147.

[19] H. Lajos, "On the problem of the pauses of speech," *Acta Linguistica Academiae Scientiarum Hungaricae*, vol. 3, no. 1/2, pp. 1–36, 1953, ISSN: 00015946. [Online]. Available: http://www.jstor.org/stable/44309054.

[20] M. Latinus and P. Belin, "Human voice perception," *Current Biology*, vol. 21, no. 4, R143–R145, 2011, ISSN: 09609822. DOI: 10.1016/j.cub.2010.12.033. [Online]. Available: http://dx.doi.org/10.1016/j.cub.2010.12.033.

[21] J. Laver and P. Trudgill, "Phonetic and linguistic markers in speech," in *Phonetic and linguistic markers in speech*, 1979.

[22] LikeSo, *Web page of the app 'likeso'*. [Online]. Available: https://sayitlikeso.com/.

[23] S. Lloyd-Hughes, *How to be brilliant at public speaking: Any audience, any situation*. Prentice Hall Life/Pearson, 2011.

[24] T. McARTHUR, "Syllable," *Concise Oxford Companion to the English Language.. Encyclopedia. com*, Sep. 2021.

[25] Merriam-Webster, *Timbre*, in *Merriam-Webster.com dictionary*. [Online]. Available: https://www.merriam-webster.com/dictionary/timbre (visited on 09/15/2021).

[26] Ng, Raymond W. M. and Lee, Tan and Leung, Cheung-Chi and Ma, Bin and Li, Haizhou, "Spoken language recognition with prosodic features," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 9, pp. 1841–1853, 2013. DOI: 10.1109/TASL.2013.2260157.

[27] Nivja de Jong and T. Wempe, *Praat Script Syllable Nuclei v2*, 2008. [Online]. Available: https://sites.google.com/site/speechrate/Home/praat-script-syllable-nuclei-v2.

[28] NumPy, *Numpy v1.21 manual*. [Online]. Available: https://numpy.org/doc/stable/index.html.

[29] Orai, *Web page of the app 'orai app'*. [Online]. Available: https://www.orai.com/.

[30] *Oxford English dictionary (Online)*. Oxford: Oxford University Press.

[31] Parselmouth, *Praat in python, the pythonic way*. [Online]. Available: https://parselmouth.readthedocs.io/en/stable/.

[32] H. Pham, *Web page of PyAudio*. [Online]. Available: http://people.csail.mit.edu/hubert/pyaudio/.

[33] P. I. Plc, *What is sound pressure level and how is it measured?* Oct. 2019. [Online]. Available: https://pulsarinstruments.com/en/post/sound-pressure-level-and-SPL-meters.

[34] B. Portillo. (2019). "El teatro nos enseña que equivocarse no es un drama," Youtube, [Online]. Available: https://www.youtube.com/watch?v=wZwqSU2Kd60&t=6s.

[35] Praat, *Praat Scripting*, 2020. [Online]. Available: https://www.fon.hum.uva.nl/praat/manual/Scripting.html.

[36] E. Rodero, "A comparative analysis of speech rate and perception in radio bulletins," *Text and Talk*, vol. 32, no. 3, pp. 391–411, 2012, ISSN: 18607330. DOI: 10.1515/text-2012-0019.

[37] ——, (2018). "Persuade con tu voz. Estrategias para sonar creíble," Youtube, [Online]. Available: https://www.youtube.com/watch?v=YlI-e4QJWG0.

[38] Scenihr, "Potential health risks of exposure to noise from personal music players and mobile phones including a music playing function," *Scientific Committee on Emerging an d Newly Identified Health Risks*, no. September, p. 81, 2008. [Online].

Available: http://ec.europa.eu/health/ph_risk/committees/04_scenihr/docs/scenihr_o_018.pdf.

[39] K. R. Scherer, H. London, and J. J. Wolf, "The voice of confidence: Paralinguistic cues and audience evaluation," *Journal of Research in Personality*, vol. 7, no. 1, pp. 31–44, 1973. DOI: 10.1016/0092-6566(73)90030-5.

[40] SonicVisualizer, *Sonic visualiser*. [Online]. Available: https://www.sonicvisualiser.org/.

[41] SpeechTools, *Web page of the app 'voice analyst'*. [Online]. Available: https://speechtools.co/voice-analyst.

[42] The Open University, *The perception of frequency*, 2018. [Online]. Available: https://www.open.edu/openlearn/science-maths-technology/biology/hearing/content-section-11.1.

[43] G. L. Trager, *The field of linguistics*. Norman, Okla: Battenburg Press, 1949.

[44] B. Tucker and K. M. Barton, *Exploring public Speaking: 2nd revision*. University of North Georgia Press, 2016.

[45] A. B. Van Zant and J. Berger, "How the voice persuades.," *Journal of Personality and Social Psychology*, vol. 118, no. 4, pp. 661–682, 2020. DOI: 10.1037/pspi0000193.

[46] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, İ. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: 10.1038/s41592-019-0686-2.

[47] *Vocal cue*. DOI: 10.1093/oi/authority.20110803120135583. [Online]. Available: https://www.oxfordreference.com/view/10.1093/oi/authority.20110803120135583.

[48] VoiceScienceWorks, *Harmonics vs. formants*. [Online]. Available: https://www.voicescienceworks.org/harmonics-vs-formants.html.

[49] N. Winfried, *Handbook of semiotics*. Indiana University Press, 2014.

[50] M. Winters, E. Macintyre, C. Peters, J. Thom, K. Teschke, and H. Davies, "Noise and hearing loss in farming," *Farm and Ranch Safety and Health Association*, Aug.

2005. [Online]. Available: https://www.researchgate.net/publication/29734945_Noise_and_hearing_loss_in_farming.

[51] S. Wood, *What are formants?* 2005. [Online]. Available: http://person2.sol.lu.se/SidneyWood/praate/whatform.html.

[52] S. Živković, "The importance of oral presentations for university students," *Mediterranean Journal of Social Sciences*, vol. 5, no. 19, p. 468, Sep. 2014. [Online]. Available: https://www.richtmann.org/journal/index.php/mjss/article/view/4278.

"Always look on the bright side of life"

Monty Python

Guillermo Ovejero y Miguel Ferreras

2021

Last Update: September 21, 2021

LaTeX lic. LPPL & powered by TeFLON CC-ZERO