



# **BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA**

---

---

## **FACULTAD DE CIENCIAS DE LA COMPUTACIÓN**

**Integración de tecnologías para detectar  
muletillas en el idioma español**

**TESIS**

**PARA OBTENER EL GRADO DE  
LICENCIADA EN INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN**

**PRESENTA:**

**VANESSA ROBLES MARGARITA**

**ASESOR:**

**Dr. PEDRO BELLO LÓPEZ**

**Puebla, Pue. Noviembre, 26, 2024**

## DEDICATORIA

A mi bello ángel que siempre me guía, a ti mi hermosa hermanita Jennifer que donde quiera que te encuentres, sé que me mandas tu bendición y amor cada día.

## AGRADECIMIENTOS

Agradezco a mi mamá Mónica Margarita Hernández y a mi papá Guillermo Robles Cano por todo el apoyo brindado a través de todo este tiempo y los consejos hacia mí. Por su paciencia y amor incondicional ante todos los obstáculos de la vida y por sus palabras de aliento cada día. Este logro también es de ustedes pues sin ustedes no lo habría logrado.

A mis hermanos con quienes he compartido momentos tristes y alegres, pero siempre nos tenemos los unos a los otros en todo momento.

A mi amoroso esposo Enrique José Ruíz, papi de mis niños, por su apoyo constante y su motivación para alcanzar este objetivo.

A mis hermosos hijitos Hayley, Allyson, Froy y Bella que son mi motivación día a día.

Agradezco sinceramente a mi profesor y asesor al Dr. Pedro Bello López, que ha confiado en mí al sugerirme este tema de tesis y me ha apoyado en la realización de esta, a través de sus conocimientos me ha guiado y asesorado, quien me motivó a culminar este gran paso importante para mi vida profesional.

A mi compañero y amigo de profesión al ICC Adrián Maceda Zayas que me dio ese pequeño pero gran empujoncito para culminar este peldaño.

Agradezco a Dios por todo lo dado en este hermoso sendero llamado vida.

## CONTENIDO

Dedicatoria.....	2
Agradecimientos.....	3
Resumen.....	5
Capítulo 1. Introducción.....	7
1.1 Objetivos generales y específicos.....	7
1.2 Objetivos específicos.....	8
1.3 Metodología usada en la investigación.....	8
1.4 Antecedentes del Proyecto.....	9
1.4 Trabajos relacionados.....	11
Capítulo 2. Marco conceptual.....	13
2.1 Muletillas.....	13
2.2 Procesamiento Digital de señales.....	17
2.3 Procesamiento de audio.....	18
2.4 Reconocimiento de voz.....	19
2.5 Transcripción de un audio o video a texto.....	20
2.6 Texto a voz.....	21
2.7 Procesamiento por frecuencias.....	22
2.8 Transcripción de archivos (SRT).....	23
2.9 Desarrollo de aplicaciones web.....	24
Capítulo 3. Diseño y pruebas del sistema.....	31
3.1 Trabajando con archivos SRT.....	31
3.2 Aplicación web para el cálculo de frecuencias.....	34
3.3 Herramientas para el desarrollo de la aplicación.....	34
3.3 Reincorporación del archivo srt al video.....	40
Conclusiones.....	42
Bibliografía.....	43

En la presente tesis se muestra una amplia investigación sobre el tema de las muletillas, de acuerdo con la Real Academia Española (RAE) la define como: VOZ O FRASE QUE SE REPITE MUCHO POR HÁBITO. Dicho en otras palabras, la muletilla o estribillo es un vicio del lenguaje porque utiliza palabras, frases o expresiones innecesarias de manera constante. La muletilla es una palabra o sonido que utilizamos de manera frecuente cuando hablamos, por lo general la usamos para ganar tiempo cuando se nos ha olvidado la idea de lo que queremos expresar y en muchos casos ya es una mala costumbre al hablar, creemos, sin razón, que así hablamos y que así es como la gente habla normalmente, sin embargo las muletillas no son necesarias en el idioma ni para comunicarnos correctamente, de hecho una persona que utiliza demasiadas muletillas puede quitar importancia a su discurso y hacer que las personas se enfoquen más en la muletilla que en lo que está queriendo expresar.

Por tanto, se presenta el diseño de un sistema que integra diferentes tecnologías para detectar muletillas en el idioma español, lo que servirá a los oradores buscar formas de poder reducir o eliminar dichas muletillas. Para la detección de muletillas se utilizan archivos SRT (Secure Reliable Transport) que son provistos por voz o videos y que pueden manipularse con la tecnología Google2SRT, posteriormente se utiliza un sistema propio usando tecnología web que realiza una revisión del archivo SRT y obtiene las frecuencias de las palabras que son comparadas con un repositorio de las muletillas más comunes en el idioma español y marcamos las palabras que podrían ser consideradas como muletillas, al finalizar usamos la tecnología VLC media player para integrar los cambios al archivo SRT y unirlo con el video con lo que al reproducir el video se ven marcadas de color las palabras consideradas muletillas en el idioma español.

Este proyecto de investigación de tesis está dividida en 4 capítulos:

Capítulo 1. Introducción: En esta sección se describe el problema, el objetivo general, específico del proyecto, la metodología de investigación utilizada y se plantea el modelo general para resolver el problema planteado, además se incluyen los trabajos relacionados del tema en cuestión.

Capítulo 2. Marco conceptual: En este apartado se presentan conceptos relacionados con las tecnologías para detectar muletillas en el idioma español, así como información básica de las herramientas que se utilizarán.

Capítulo 3. Desarrollo del sistema: En esta sección se describe la forma en que se integraron las diferentes tecnologías para detectar las muletillas en el idioma español, así como pantallas de muestra del funcionamiento del sistema completo.

Conclusión: Por último, se presenta el conocimiento obtenido sobre el sistema web de integración de tecnologías para detectar muletillas en el idioma español, el cual sería una excelente propuesta para un uso adecuado de detectar y así eliminar las muletillas, además se proponen algunos trabajos futuros.

Finalmente se incluye la bibliografía base para el desarrollo de este proyecto de tesis.

El procesamiento de audio y video [1] es ampliamente utilizado en diversos ámbitos de la ciencia que puede ser aplicada a la sociedad, ya sea para detectar voz, identificar personas o como en este caso para determinar si una persona recurre de manera frecuente al uso excesivo de palabras denominadas muletillas [2].

Las muletillas [3] representan un uso excesivo del lenguaje se trata de palabras o frases cortas que se repiten una y otra vez al hablar y que generalmente son palabras que no son necesarias o están demás al hablar.

Para el análisis de los audios [4] y videos es necesario conocer métodos y/o herramientas para realizar lo que se conoce como transcripción, la transcripción está representada por un archivo denominado SRT [5] que contiene una secuencia de texto en un determinado minuto, segundo o milisegundo. De esta forma para la detección de las muletillas se extrae este archivo SRT y se procesa el texto plano a través del análisis de las frecuencias de las palabras. Para determinar si existen muletillas se tiene un diccionario de las muletillas más comunes en el idioma español.

Por lo tanto, en este trabajo se introducen los conceptos básicos para procesar un audio o video para obtener su transcripción (SRT) que posteriormente es procesada para obtener las muletillas más comunes en el idioma español. Se utilizan una serie de herramientas para lograr descargar un audio o video, obtener su transcripción, realizar un análisis de frecuencias de la aparición de palabras y compararlas con un diccionario para así modificar el archivo SRT marcando las muletillas encontradas.

### 1.1 OBJETIVOS GENERALES Y ESPECÍFICOS

Integrar un conjunto de tecnologías y sus herramientas asociadas en la detección de muletillas en el idioma español en audio y video a través de una aplicación que esté disponible a través de internet.

## 1.2 OBJETIVOS ESPECÍFICOS

- Investigar las herramientas para realizar la transcripción de audio y video en texto.
- Revisar las diferentes herramientas para poderlas integrar en el reconocimiento de muletillas.
- Diseñar el sistema en web para reconocer muletillas en español.
- Implementar la aplicación que integra las herramientas y el sistema web para reconocer muletillas.
- Realizar las pruebas necesarias para determinar el correcto funcionamiento de la aplicación de detección de muletillas.

## 1.3 METODOLOGÍA USADA EN LA INVESTIGACIÓN

Los pasos necesarios para el desarrollo de este trabajo de tesis se describen puntualmente:

- Investigar el estatus actual del procesamiento de audio, video, transcripción y análisis de texto, revisar las bases de datos Scopus y ACM para conocer el avance en el desarrollo del procesamiento de la transcripción (archivos SRT).
- Análisis del sistema para el desarrollo de la aplicación. Integrar diferentes herramientas junto con PHP [6, 7] para crear una aplicación web.
- Herramientas para utilizar, se deben considerar diversas tecnologías y herramientas para la extracción del archivo SRT y VLC media player para integrar los cambios al archivo SRT y unirlo con el video.
- Análisis de resultados: se deben realizar diferentes pruebas para poder integrar las herramientas.

Para la obtención de las muletillas se consideró una lista de muletillas, como se muestran en la Tabla 1.1 y se calcularon las frecuencias de las palabras, para lo cual se eliminaron espacios y caracteres especiales. El método de frecuencias es el método estadístico más general para analizar palabras, se almacenan en un arreglo simple y se ordenan de mayor a menor [8]. El trazado del número de veces (frecuencia de aparición) en que se repite cada una de las diferentes palabras de un texto se denomina histograma.

**Tabla 1.** Principales palabras consideradas como muletillas

<i>Que</i>	<i>Obvio</i>	<i>Y dice</i>	<i>Tu muy bien</i>
preguntas	Aja	te lo juro	o sea
dudas	Sabes	es de que	yo creo que
estee	Bien	estas mal	es claro
hee	Aaa	Ora	Si
pues	Okey	hay aja	lo que viene siendo
igual	Vale	ha no bueno	si o no
pero	Verda	como se llama	si lo ven
ira	Verdad	de acuerdo	está claro

#### 1.4 ANTECEDENTES DEL PROYECTO

El formato de archivo externo basado en texto más popular que se usa hoy en día es, con mucho, el SRT (SubRip Text). SRT es ampliamente compatible con una variedad de reproductores y programas de creación de subtítulos. Esencialmente, un documento SRT contiene las entradas de texto que se mostrarán y el momento de esa presentación. No hay soporte para comentarios. Algunos reproductores multimedia reconocen los comandos de formato de texto (negrita, cursiva, subrayado y color de fuente) ingresados con etiquetas HTML [9].

Una persona cuya forma de hablar impide la adecuada manifestación de sus ideas, padece una perturbación de la palabra que le incapacita para emplearla con propiedad, ya sea por vocabulario reducido, ignorancia o apatía, pues aunque constituya una mera etiqueta para nombrar objetos, acciones, sentimientos, emociones, cualidades, experiencias y conceptos, cuando se utiliza reiteradamente sin explicación alguna, el hábito hace que deje de advertir ese tedioso defecto, pero no que pase inadvertido para los demás [3].

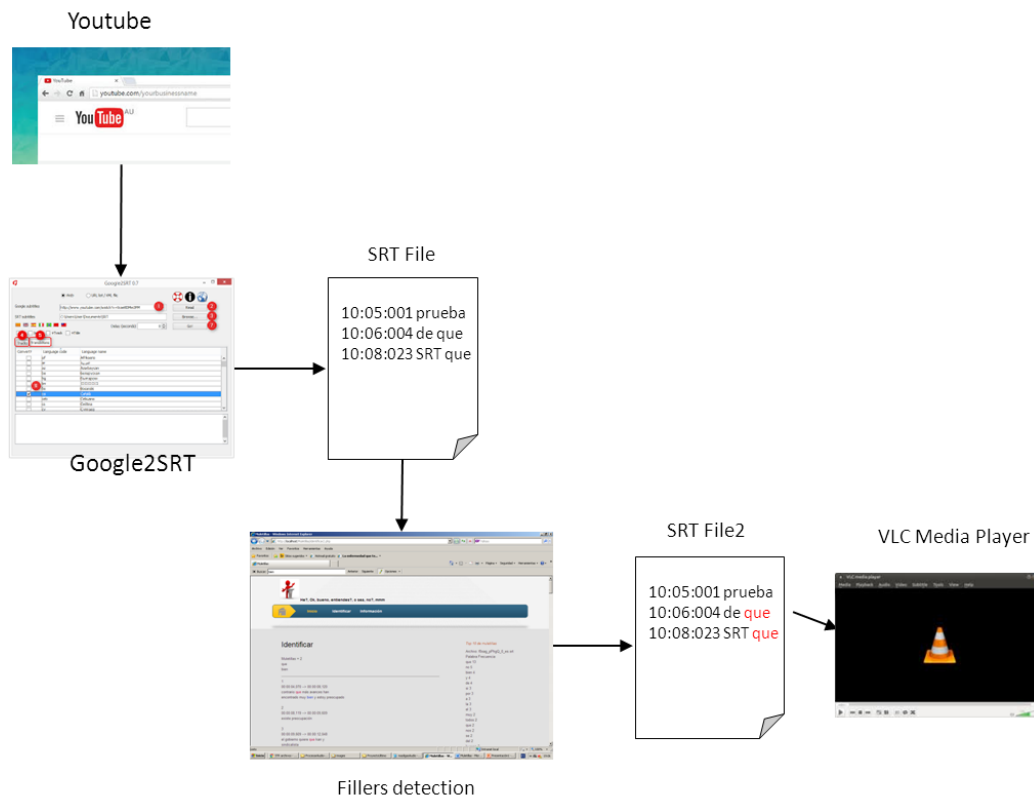
En el procesamiento de audio se pueden identificar cuatro problemas generales:

- Reconocimiento de voz

- Identificación de voz
- Transcripción de un audio o video a texto
- Texto a voz

Los reconocedores de voz se pueden clasificar en varias dimensiones diferentes. Algunos necesitan ser entrenados para la voz de un usuario, mientras que otros son "independientes del hablante". Algunos permiten que los usuarios hablen continuamente, mientras que otros obligan a los usuarios a hablar con pausas entre cada palabra. Finalmente, algunos reconocedores de voz requieren el uso de una gramática de reglas, otros permiten solo frases fijas para comando y control, y otros se basan puramente en modelos estadísticos [10].

El modelo propuesto para este proyecto de tesis se muestra en la Figura 1.1, donde se integran las diversas tecnologías propuestas para el desarrollo del sistema, resaltando el procesamiento de archivos SRT.



**Figura 1.1.** Modelo del proceso de detección de muletillas

#### 1.4 TRABAJOS RELACIONADOS

El estado del arte en tecnología de voz ha progresado hasta el punto en que ahora es práctico para los diseñadores considerar la integración de entrada y salida de voz en sus aplicaciones. En este proyecto [9], formalizan y validan un marco de trabajo de sincronización de subtítulos en dos fases.

El beneficio sobre los enfoques actuales radica en el uso de anotaciones de huellas dactilares de audio generadas a partir de la señal de audio base como anclas de sincronización de segundo nivel. De esta manera, permiten que el reproductor multimedia corrija dinámicamente durante la reproducción los casos más comunes de desalineación de sincronización de subtítulos que comprometen la experiencia de visualización de los usuarios. En [10], un tutorial se centra en técnicas para diseñar interfaces de voz. Los temas cubiertos incluyen una introducción a la entrada y salida de voz, una discusión sobre los problemas de diseño de la interfaz de usuario de voz y una exploración de las formas de involucrar a los usuarios en el proceso de diseño.

En [11] se muestra que los sistemas incrementales se perciben como más reactivos, naturales y más fáciles de usar que los sistemas no incrementales. Sin embargo, el trabajo previo sobre NLG incremental no ha empleado los avances recientes en la optimización estadística mediante el aprendizaje automático. Este documento [11] combina los dos enfoques y muestra cómo las operaciones de actualización, revocación y purga que se utilizan normalmente en los enfoques incrementales pueden implementarse como transiciones de estado en un proceso de decisión de Markov. Diseñan un modelo de NLG incremental que genera resultados basados en interpretaciones de micro-turno de las expresiones del usuario y es capaz de optimizar sus decisiones utilizando el aprendizaje automático estadístico. Los resultados presentados muestran que el agente aprende a evitar largos tiempos de espera, rellenos y autocorrecciones, al reordenar el contenido en función de su confianza.

En [12], para facilitar una comunicación fluida a través de un sistema de videoconferencia, proponen un método novedoso, "Visual Filler", que llena los espacios temporales en los turnos causados por la existencia de retrasos. Visual Filler superpone un estímulo visual artificial que tiene una función similar a la de los sonidos de relleno en una pantalla con las imágenes de los participantes. Y han evaluado la efectividad de un Visual Filler para reducir la

falta de naturalidad de tomar turnos en una situación de diálogo diádico simulado con un retraso.

En el proyecto definido en [8] evalúa el beneficio de eliminar los rellenos (p. ej., ya sabes, me gusta) al principio del análisis del habla conversacional. Los estudios de legibilidad han demostrado que las disfluencias (rellenos y reparaciones del habla) se pueden eliminar de las transcripciones sin comprometer el significado, y se ha demostrado que eliminar las reparaciones antes del análisis mejora su precisión. Luego explore si esta estrategia de eliminación temprana también es beneficiosa con respecto a los rellenos. Los experimentos informados miden el efecto de la eliminación temprana en condiciones de entrenamiento del analizador dentro y fuera del dominio utilizando un analizador de última generación. Si bien se encuentra que la eliminación temprana produce solo un beneficio modesto para el análisis en el dominio, se logra una mejora significativa para la adaptación fuera del dominio. Esto sugiere un papel potencialmente más amplio para el modelado de disfluencia en la adaptación de herramientas basadas en texto para procesar el habla.

A continuación, se describen los principales conceptos teóricos utilizados para el desarrollo de este trabajo de tesis.

### 2.1 MULETILLAS

También conocida como coletilla [8], latiguillo, bordón, bordoncillo o ripio es una palabra o expresión que se repite o intercala en el discurso con excesiva frecuencia, como si se tratara de un tic de la persona que habla o escribe. Estas frases o palabras en otras situaciones tienen un significado concreto, sin embargo, al usarlas como muletillas no significan nada, es decir, que podrían eliminarse y el discurso seguiría teniendo el mismo significado. Entonces una muletilla es solo eso, una pequeña muleta, un apoyo en el discurso que se repite mucho por hábito o incluso puede ser debido a los nervios de la persona que en su momento las utilice. Por ejemplo, en el siguiente texto tenemos varias muletillas.

*“Pues, la verdad es que, cuando te pones a pensar, muchas veces al hablar, no nos damos cuenta de que, pues, estamos usando muletillas todo el tiempo. Es decir, las metemos casi sin querer para, pues, rellenar esos espacios cuando estamos pensando en lo que vamos a decir. Y, pues, lo curioso es que, aunque a veces pueden parecer innecesarias, estas palabras o frases nos ayudan a mantener el ritmo de la conversación, ¿no? Entonces, pues, se vuelven parte de nuestra manera de expresarnos sin que nos demos cuenta”.*

Este tipo de expresiones son comunes en la conversación informal y a veces se usan para ganar tiempo o aclarar ideas mientras se piensa qué decir.

Existen diferentes tipos de muletillas [3]:

- **Conjuntivas:** Se usan para llenar los espacios que necesitan pausas.  
Por ejemplo: En lugar de decir “me llamo Vanessa, soy de Puebla, pero durante el invierno me gusta viajar a San Diego”, se escucha así “me llamo Vanessa y mmmmm, soy de Puebla, pero durante el invierno mmmmm me gusta viajar a mmmmm San Diego”.

- **Dramáticas:** son las usadas para hacer énfasis.  
Por ejemplo: El famoso “literal” es el más común.
- **De pasarela:** Se usan al comienzo de las oraciones para iniciar con una idea.  
Por ejemplo: “como”, “este”, “o sea”, “en teoría”, “así pues”, “así mismo”, “amm”.
- **De revisión:** Son aquellas palabras que son usadas para revisar la información con la audiencia. No hay problema con usarlas porque son de apoyo, solamente no hacer uso de ellas de manera repetitiva.  
Por ejemplo: “¿verdad?” “¿Ok?” “¿me entiendes?” “¿cierto?”.
- **El timo:** Cuya principal característica es su carácter repetitivo en varias fases de un mismo tema y se refieren a locuciones.  
Por ejemplo: Válgame Dios.
- **El latiguillo:** Caracterizado por su naturaleza temporal de modo similar al recorrido que traza una moda.  
Por ejemplo: Díganos.
- **La empuñadura:** Es necesaria para encauzar y conducir determinados coloquios.  
Por ejemplo: El “Érase una vez de los cuentos”.

### **Problemas con las muletillas**

Al utilizar las muletillas [13], estas restan un mensaje claro, es decir, le quitan importancia a lo que queremos transmitir y es por ello que se debe considerar ser más claros y concisos y por eso es importante eliminarlas.

Las muletillas [14] suelen ser expresiones innecesarias, comodines o palabras abstractas y breves, generalmente pronombres o jerga, que se utilizan sin cumplir una función primordial en una determinada oración o frase.

### **oraciones con muletillas:**

- ¿Este, que te iba a decir?

- Bueno, ammm, nada, nos vemos mañana.
- No sé, la verdad, mejor me voy ammm para mi casa.
- La cosa es que no sé qué pasa, lo que sí es que se pospone para mañana.
- Lo que te quiero decir es que, si estudiaste el formulario, te irá bien en el examen.

### **Clasificación de las muletillas según su uso:**

**1. Para empezar el discurso:** Para una decisión: bueno, oye, mira, así pues, a ver, este.

Por ejemplo:

-Este, vayamos de compras al mercado.

-Mira, yo prefiero el otro cine.

**2. Para introducir una afirmación:**

La mera verdad, la verdad es que, resulta que, total que, lo que pasa es que, o sea, a decir verdad, ¿sabes lo que pasa?, etc.

Por ejemplo:

La mera verdad es que no sé de qué se trata.

¿Sabes lo que pasa?, que no tengo idea de lo que me tratas de decir.

Resulta que, tengo clases a esa hora, pues no puedo verte entonces.

**3. Para llamar y mantener el interés del interlocutor:**

Esto, wey, ¿qué te iba a decir?, mira, ¿no es cierto?, ¿verdad?, ¿no?, ¿o qué?

Por ejemplo:

-Este libro está muy interesante, ¿verdad?

-Pues sí, ¿no es cierto que está mejor este lugar?

**4. Para controlar el turno de palabra:**

Escucha; escucha un momento; calla, calla.

Por ejemplo:

-Vi algo muy interesante en casa de Luis.

-¡Calla, calla que no sabes lo que tengo que contarte sobre Luis!

## **5. Para ganar tiempo en el discurso para pensar sobre las ideas que se están**

### **exponiendo:**

O sea, es decir, como si dijéramos, diríamos, digamos, a ver, como, como muy, tipo, del tipo de, pues, esto, este.

Por ejemplo:

- ¿Sabes qué? Es de suma importancia conocer a tus compañeros.

Ah, ¿sí?, ¿Cómo por qué? ¿de qué se trata?

Digamos que...tienes que saber sus gustos, tipo de personas y su carácter.

-Este...lo que pasa es que debo irme.

## **6. Para buscar la comprensión y complicidad con el interlocutor:**

¿Entiendes?, ¿sabes?, ¿me explico?, ¿te enteras?, ¿sí?, ¿vale?, ¿ok?, ¿ya?, etc.

Por ejemplo:

-Al entrar giras hacia la derecha y ahí está el módulo, ¿ok?

-Te veo a las 6 en este lugar, ¿me explico?

## **7. Para justificar el discurso:**

Entiéndeme, como el que dice, por decir de alguna manera, digo yo, no es porque yo lo diga, es un decir, por decir algo.

Por ejemplo:

-La verdad es que, no es porque yo lo diga, pero así no se hacen las cosas, no está bien, no es porque yo lo diga, pero así lo veo.

-Después de todo ya es tarde para aprobar... digo yo... debimos entregar el trabajo a tiempo, por decir algo.

## **8. Para dar énfasis (normalmente con intención negativa):**

Vaya (=qué), zhale

Por ejemplo:

- Vaya día tan caluroso... para tomar agua con hielos ... ¡Qué día!...
- Zhale, que mal plan, hay que hacerlo de nuevo.

### **9. Para mostrar desacuerdo:**

Qué quieres que te diga, que puedo decir yo, qué más da, que le puedo hacer, etc.

Por ejemplo:

- Que puedo decir yo, si ya está tomada la decisión.
- Si eso es lo que piensas sobre mí, que le puedo hacer.
- Si ya estás conforme que quieres que te diga.

### **10. Para invitar a la reflexión:**

Imagínate, fíjate, imagina, etc.

Por ejemplo:

- No me gustan los nachos. Imagínate.
- Fíjate, que me encantó el libro.

### **11. Para concluir una idea:**

Bien, ea, pues nada, en fin, bueno, venga pues, órale, etc.

Por ejemplo:

- Bien, vayamos a casa a celebrar.
- Bueno, pues todo está decidido.
- En fin, así damos por terminada la sesión.

## **2.2 PROCESAMIENTO DIGITAL DE SEÑALES**

El Procesado Digital de Señales [15] (PDS) es una disciplina que ha permitido a los ingenieros del área aplicar estas técnicas al modelado de múltiples problemas del mundo que nos rodea, proporcionando una mejor manera de entenderlo y ofreciendo a menudo soluciones que consiguen transformarlo radicalmente.

El campo de actuación del PDS [16] abarca desde los sistemas de telecomunicaciones a métodos de análisis y predicción financiera, desde los sistemas de control y de toma de decisiones, a las tecnologías multimedia, etc. y afecta a todos los campos de la actividad humana, incluyendo nuestra forma de interactuar con las máquinas. También se centra en el estudio de diferentes señales que son la fuente de algunas de las principales aplicaciones de estas tecnologías: el audio, la voz humana y las imágenes.

### 2.3 PROCESAMIENTO DE AUDIO

El **procesamiento de audio** [1] consiste en el estudio de los sonidos registrados mediante uno o más micrófonos e incluye múltiples objetivos: análisis de los sonidos para su caracterización, extracción de parámetros objetivos de la voz para su detección, codificación para su envío o almacenamiento en medios digitales, mejora del texto, etc. En este campo se incluye también el procesado de señales.

El **procesado de voz** [4] se puede ver como un subconjunto del procesado de audio en el que la señal bajo estudio es la voz humana. Es un área que abarca también muchas disciplinas en función de los objetivos perseguidos: codificación, mejora, reconocimiento de habla, síntesis, análisis, detección de idioma, de locutor.

Una señal digital no es audible, ya que requiere ser decodificada antes de su reproducción. Sin embargo, la codificación y posterior decodificación de una señal digital para su audición tiene varias ventajas, aunque también algunos inconvenientes.

#### **VENTAJAS:**

- La señal discreta (digital) es más fácil de transmitir, almacenar o manipular (en el caso del sonido: editar, comprimir, etc).
- La señal digital es inmune al ruido. Es menos sensible que la analógica a las interferencias, etc.
- Se puede tomar una muestra de sonido y cambiar cualquiera de sus parámetros para generar un sonido diferente sin tener que recrearlo en la realidad (las aplicaciones de esta ventaja en la generación de efectos especiales son infinitas).
- La señal digital permite la multigeneración infinita sin pérdidas de calidad.

- Ante la pérdida de cierta cantidad de información, la señal digital puede ser reconstruida gracias a los sistemas de regeneración de señales (utilizados también para amplificarla, sin introducir distorsión). También cuenta, con sistemas de detección y corrección de errores que, por ejemplo, permiten introducir el valor de una muestra dañada, obteniendo el valor medio de las muestras adyacentes(interpolación).
- La señal digital puede ser enviada a casi cualquier punto del planeta en cualquier momento a un muy bajo costo a través de internet y a partir de aquí puede ser reenviada a su remitente o a algún otro destino. Esto sin que la señal sufra variaciones o alteraciones de calidad severas.
- Con el tiempo no se degrada.

## **DESVENTAJAS**

- Se necesita una conversión analógica-digital previa y una decodificación posterior, en el momento de la recepción.
- Hay una pérdida inherente de información al convertir la información continua en discreta. Por mínimo e insignificante que resulte siempre hay un error de cuantificación que impide que la señal digital sea exactamente equivalente a la analógica que la originó.
- La señal digital requiere mayor ancho de banda para ser transmitida que la analógica. Además, requiere una sincronización precisa entre los tiempos del reloj de transmisor, con respecto a los del receptor. Un desfase, por mínimo que sea, cambia por completo la señal.
- Si se utiliza compresión con pérdida, será imposible reconstruir la señal original.

## **2.4 RECONOCIMIENTO DE VOZ**

El reconocimiento de voz [1] es un subcampo interdisciplinario de la lingüística computacional que desarrolla determinadas metodologías y tecnologías que permiten a las computadoras reconocer y traducir el lenguaje hablado en texto. Esto también se conoce como reconocimiento automático de voz, reconocimiento de voz en computadora o voz en texto. Por ello, incluye conocimientos e investigaciones en lingüística, informática e ingeniería eléctrica.

Algunos sistemas de reconocimiento de voz requieren de aprendizaje (también llamado inscripción) cuando un orador individual lee un texto o un diccionario aislado en el sistema. El sistema analiza la voz específica de una persona y la utiliza para ajustar el reconocimiento de voz de esa persona, lo que resulta en una mayor precisión. Los sistemas que no utilizan el aprendizaje se denominan sistemas independientes del altavoz, mientras que los que lo utilizan se denominan dependientes de los altavoces.

El reconocimiento de audio se realiza en tres pasos:

- Las palabras se destacan. Por lo general, hay varias hipótesis de una misma palabra reconocida.
- Las hipótesis se prueban con un modelo de lenguaje. El modelo verifica cómo es de consistente una nueva palabra con aquellas reconocidas anteriormente.
- A continuación, se procesa el texto reconocido: los números se convierten en números, se colocan algunos signos de puntuación como los guiones, etc. Este texto convertido es el resultado final del reconocimiento que se envía al cuerpo de la respuesta.

## 2.5 TRANSCRIPCIÓN DE UN AUDIO O VIDEO A TEXTO

Existen diferentes herramientas hoy en día para transcribir nuestro audio a texto, dichas herramientas son de gran utilidad, puesto que nos facilitan tiempo.

Algunas herramientas populares son:

- Google Cloud Speech-to-Text: Ofrece una API para transcribir archivos de audio en texto con alta precisión (<https://cloud.google.com/speech-to-text/>).
- IBM Watson Speech-to-Text: Proporciona un servicio basado en la nube para convertir la voz en texto con soporte para varios idiomas y formatos de archivo (<https://www.ibm.com/mx-es/watson>).
- Microsoft Azure Speech to Text: Ofrece capacidades de transcripción de voz a texto con soporte para múltiples idiomas y formatos de archivo (<https://azure.microsoft.com/es-mx/free/>).
- Descript: Es una herramienta que combina la edición de audio y la transcripción automática en una sola plataforma ([filmora.wondershare.net](http://filmora.wondershare.net)).

- Trint: Ofrece transcripción automatizada de audio y video con herramientas de edición integradas (<https://turboscribe.ai/es/>).
- Rev: Es un servicio de transcripción humano y automático que garantiza alta precisión (<https://transkriptor.com/>).
- Sonix: Proporciona transcripción automática de audio y video con edición colaborativa en tiempo real (<https://easywithai.com/ai-transcription-tools/sonix/>).

## 2.6 TEXTO A VOZ

Con texto a voz, el dispositivo puede convertir entrada de texto y reproducir audio en voz alta. Algunas herramientas y servicios son:

- Google Text-to-Speech: Es una API (Interfaz de Programación de Aplicaciones) que permite convertir texto en voz con una variedad de voces y opciones de configuración (<https://cloud.google.com/speech-to-text/>).
- Amazon Polly: es una herramienta que ofrece capacidades avanzadas de síntesis de voz con soporte para múltiples idiomas y voces naturales (<https://aws.amazon.com/es/polly/>).
- Microsoft Azure Text-to-Speech: Proporciona servicios de conversión de texto a voz con calidad de voz natural y opciones de personalización (<https://azure.microsoft.com/es-mx/free/>).
- IBM Watson Text To Speech: Ofrece síntesis de voz con capacidades de personalización y soporte para varios idiomas (<https://www.ibm.com/mx-es/watson>).
- Natural reader: Es una aplicación y servicio en línea que convierte texto en voz con opciones de personalización y diferentes voces (<https://www.naturalreaders.com/>).
- Balabolka: Es una herramienta gratuita para Windows que puede convertir texto en voz y guardar los resultados en varios formatos de archivo de audio (<https://www.cross-plus-a.com/es/balabolka.htm>).
- Voice Dream Reader: Es una aplicación para dispositivos móviles que permite convertir texto en voz con opciones de personalización y accesibilidad (<https://www.voicedream.com/>).

La tecnología texto a voz (TTS, por sus siglas en inglés) es un tipo de tecnología de asistencia que lee texto digital en voz alta. En ocasiones se conoce como tecnología de lectura en voz alta [15].

## 2.7 PROCESAMIENTO POR FRECUENCIAS

El procesamiento por frecuencias [1, 15, 16] es una técnica utilizada en varios campos, como la música, la ingeniería de señales y el procesamiento de imágenes, entre otros. Consiste en analizar una señal (ya sea de audio, imagen u otra forma de datos) en el dominio de la frecuencia en lugar del dominio temporal o espacial.

En el procesamiento de señales de audio [4], por ejemplo, el procesamiento por frecuencias implica descomponer la señal de audio en sus componentes de frecuencia, utilizando técnicas como la transformada de Fourier. Esto permite analizar características específicas de la señal en el dominio de la frecuencia, como tonos, armónicos y contenido espectral.

En el procesamiento de imágenes [1], el procesamiento por frecuencias implica aplicar transformadas de dominio de frecuencia, como la transformada de Fourier de dos dimensiones (para imagen) o la transformada de Wavelet, para analizar y manipular la información de la imagen en el dominio de la frecuencia.

El procesamiento por frecuencias es una herramienta poderosa que permite analizar y procesar señal y datos en el dominio de la frecuencia, lo que comúnmente facilita la extracción de características importantes y la realización de tareas específicas de procesamiento.

El procesamiento de frecuencias proporciona representaciones estadísticas y gráficas que son útiles para la descripción de distintos tipos de variables. Permite obtener una descripción de la variable a partir de las tablas de frecuencias, histogramas, gráficos de barras, percentiles, índices de tendencia central e índices de dispersión.

## 2.8 TRANSCRIPCIÓN DE ARCHIVOS (SRT)

Un archivo de subtítulos **SRT** (también conocido como SubRip o archivo SRT) [4] es un archivo de texto plano que permite añadir subtítulos a un vídeo después de producido.

Es uno de los formatos de archivo de subtítulos más populares para contenido de vídeo. Estos archivos de texto sin formato incluyen el texto de los subtítulos en secuencia, junto con los códigos de tiempo de inicio y finalización.

### **Estructura del archivo SRT**

Cada subtítulo tiene cuatro partes en el archivo SRT.

1. Un contador numérico que indica el número o la posición del subtítulo.
2. Hora de inicio y finalización del subtítulo separadas por → caracteres
3. Texto del subtítulo en una o más líneas.
4. Una línea en blanco que indica el final del subtítulo.

En la figura 2.1 se muestra un ejemplo de un archivo SRT de un video.

```
1
00:00:09,989 --> 00:00:13,499
marco antonio bueno tú también eres de
los pies veteranos llegues bien chavo de

2
00:00:13,499 --> 00:00:16,349
esta selección y que influye la
experiencia que le puedes aportar este

3
00:00:16,349 --> 00:00:17,498
equipo

4
00:00:17,498 --> 00:00:18,879
pues

5
00:00:18,879 --> 00:00:24,059
todos dudar de que en la mayoría de
transportes internacionales y también

6
00:00:24,059 --> 00:00:25,158
varios
```

Ln 1, Col 1 | 4,104 caracteres. | 100% | Windows (CRLF) | UTF-8 con BOM

**Figura 2.1.** Ejemplo de un archivo SRT

Para especificar la hora se utiliza formato

HORAS:MINUTOS:SEGUNDOS,MILISEGUNDOS (00:00:00,000).

## 2.9 DESARROLLO DE APLICACIONES WEB

El Desarrollo de Aplicaciones Web [6, 7,] es aquel que permite crear herramientas a las que los usuarios pueden acceder a través de Internet. En la web podremos obtener información o interactuar con elementos que nos permitan la comunicación entre nosotros, los usuarios y la web.

Una aplicación web es una aplicación de software [17] interactiva que permite a los usuarios realizar tareas o funciones específicas a diferencia de un sitio

web que son estáticos y existen para proporcionar información y no se requiere la interacción del usuario para lograr su objetivo.

Las aplicaciones web [18] también son herramientas que pueden utilizar los usuarios accediendo mediante un servidor web (gracias a Internet o a alguna intranet) a través de un navegador, que ejecutará la tarea. Las aplicaciones web aportan un valor añadido muy importante a nuestro sitio y pueden ayudar a posicionarnos como un referente en el sector, si llegan a viralizarse. Hay muchas clases de aplicaciones web que pueden ser utilizadas para muchos fines diferentes.

Algunos ejemplos de este tipo de herramientas son editores de fotografía, bases de datos interactivas, calculadoras de impuestos, conversores de moneda o juegos; siempre que se alojen online y no sea preciso efectuar ninguna descarga, ni en el ordenador ni en el teléfono móvil. Como las aplicaciones web se alojan en un servidor, cualquier dispositivo puede acceder a ellas, por lo que deben ser capaces de adaptarse a todos los tamaños de pantalla, a los distintos dispositivos móviles y a los diferentes navegadores que pueda utilizar el usuario.

Por sus características intrínsecas, las aplicaciones web conllevan una serie de ventajas y de desventajas frente a las aplicaciones móviles, que sí necesitan ser descargadas en un dispositivo.

Las aplicaciones web se han vuelto cada vez más populares en los últimos años, ya que ofrecen varias ventajas sobre las aplicaciones de escritorio tradicionales. Sin embargo, también tienen algunas desventajas que es necesario considerar.

## **Ventajas**

- **Compatibilidad multiplataforma:** se puede acceder a las aplicaciones web desde cualquier dispositivo con un navegador web, lo que las hace independientes de la plataforma y de fácil acceso para una amplia gama de usuarios.
- **Bajos costos de instalación y mantenimiento:** las aplicaciones web no requieren instalación ni actualizaciones en dispositivos individuales, lo que reduce los costos de instalación y mantenimiento para usuarios y desarrolladores.

- Fácil escalabilidad: las aplicaciones web se pueden escalar fácilmente para satisfacer las necesidades de una base de usuarios en crecimiento sin necesidad de hardware o infraestructura adicional.
- Actualizaciones en tiempo real: las aplicaciones web se pueden actualizar en tiempo real, lo que garantiza que los usuarios siempre tengan acceso a las funciones más recientes.
- Colaboración mejorada: las aplicaciones web permiten la colaboración en tiempo real y el intercambio de datos entre usuarios, lo que mejora la comunicación y la productividad.

### **Desventajas**

- Dependencia de internet: las aplicaciones web requieren una conexión a Internet para funcionar, lo que las hace inutilizables en modo fuera de línea o en áreas con mala conectividad de red.
- Riesgo de seguridad: las aplicaciones web son vulnerables a amenazas de seguridad como filtraciones de datos, ataques de phishing y secuencias de comandos entre sitios (Cross Site Scripting por sus siglas en inglés XSS).
- Funcionalidad limitada: las aplicaciones web pueden tener un nivel de funcionalidad diferente al de las aplicaciones de escritorio tradicionales, especialmente cuando se trata de acceder a archivos locales o ejecutar cálculos complejos.
- Limitaciones de rendimiento: las aplicaciones web pueden sufrir limitaciones de rendimiento debido a factores como conexiones de red lentas, problemas de compatibilidad del navegador y acceso limitado al hardware del dispositivo.
- Problemas de compatibilidad del navegador: es posible que las aplicaciones web no funcionen correctamente en diferentes navegadores web, lo que requiere pruebas adicionales y recursos de desarrollo para garantizar la compatibilidad.

### **Tipos de aplicaciones web**

Existen varios tipos de aplicaciones web [18], cada una tiene sus propias características y casos de uso únicos las más comunes son:

**Aplicaciones de web estáticas:** son aplicaciones web simples que no requieren ningún procesamiento de lado del servidor. Se componen principalmente de archivos HTML, CSS y JavaScript y se utilizan para mostrar contenido estático, como sitios web informativos.

**Aplicaciones web dinámicas:** estas aplicaciones web utilizan secuencias de comandos del lado del servidor para generar contenido dinámico, como datos específicos del usuario, formularios interactivos y sitios web de comercio electrónico. A menudo utilizan tecnologías backend como PHP, Python, Ruby on Rails o Node.js.

**Aplicaciones de una sola página (SPA):** estas aplicaciones web cargan una página HTML y actualizan dinámicamente el contenido mediante JavaScript. Los SPA ofrecen una experiencia de usuario perfecta, ya que no requieren actualizaciones de página y pueden ser más rápidos que las aplicaciones tradicionales de varias páginas.

**Aplicaciones web progresivas (PWA):** las PWA son aplicaciones web que utilizan tecnologías web modernas para brindar a los usuarios una experiencia similar a la de las aplicaciones nativas, incluida la funcionalidad sin conexión, notificaciones automáticas y acceso al hardware del dispositivo.

**Servicios web:** estas aplicaciones web proporcionan un conjunto de API que permiten que otras aplicaciones se comuniquen con ellas, generalmente utilizando el protocolo HTTP. Se utilizan para crear integraciones entre diferentes aplicaciones y sistemas.

**Aplicaciones web del portal:** estas aplicaciones web proporcionan una puerta de entrada unificada a diferentes recursos, como noticias, correo electrónico y redes sociales. Suelen utilizarse como portales de intranet en grandes organizaciones.

**Aplicaciones web de comercio electrónico:** estas aplicaciones web permiten a las empresas vender productos o servicios en línea. A menudo incluyen funciones como carritos de compras, pasarelas de pago y sistema de gestión de pedidos.

### **Lenguajes más utilizados para hacer sistemas web**

- HTML: Es un lenguaje de marcado utilizado para crear la estructura y contenido de páginas web

- CSS: Es un lenguaje de especificación utilizado para controlar la presentación y diseños de páginas web
- JavaScript
- PHP
- Java
- Python

**JavaScript (lenguaje de programación interpretado).**- Utilizado para agregar interactividad a páginas web.

Se utiliza para crear efectos visuales, animaciones, validaciones de formularios y de manipular el DOM (Modelo de objetos del documento).

Es el cerebro de la página web, proporcionando la lógica y funcionalidad.

**PHP (lenguaje de programación del servidor).**- Es un lenguaje de programación de lado del servidor utilizado para desarrollar aplicaciones web dinámicas.

Se utiliza para interactuar con bases de datos, autenticar usuarios y generar contenido dinámico.

Es el lenguaje de programación del servidor, proporcionando la lógica y funcionalidad en el lado del servidor.

**Java (lenguaje de programación orientado a objetos).**-Utilizado para desarrollar aplicaciones de escritorio, móviles y web.

Se utiliza para crear aplicaciones complejas, como sistemas de gestión de bases de datos, aplicaciones de red y más.

Es un lenguaje de programación multiplataforma, lo que significa que se puede ejecutar en cualquier dispositivo que tenga la máquina virtual de Java.

**Python (lenguaje de programación interpretado).**-Utilizado para desarrollar aplicaciones de escritorio, web y científicas.

Se utiliza para crear aplicaciones de inteligencia artificial, análisis de datos, automatización y más.

Es un lenguaje de programación fácil de aprender y utilizar, con una gran comunidad de desarrolladores.

## **Tecnologías backend [19]**

Son aquellas que se utilizan para construir la parte del servidor de una aplicación o sitio web, es decir, todo lo que sucede detrás de escena para que la aplicación funcione correctamente.

Algunas de las tecnologías más populares incluyen:

- Lenguajes de programación como Java, Python, Ruby, PHP, Go y JavaScript (con Node.js).
- Frameworks como Express.js, Django, Ruby on Rails, Laravel y Spring boot.
- Bases de datos como MySQL, PostgreSQL, MongoDB y SQL Server.
- Servidores web como Apache, Nginx y IIS.
- APIs (Application Programming Interfaces) y RESTful APIs.
- Contenedores como Docker y Kubernetes.
- Mensajería como RabbitMQ y Apache Kafka.

Estas tecnologías se utilizan para crear lógica de negocio, interactuar con bases de datos, autenticar usuarios, gestionar sesiones, y mucho más.

## **Tecnologías Frontend**

Son aquellas que se utilizan para construir la parte del cliente de una aplicación o sitio web, es decir, la interfaz gráfica que interactúa con el usuario. Algunas de las tecnologías frontend más populares incluyen:

- Lenguajes de programación como JavaScript, HTML, CSS y TypeScript.
- Frameworks como React, Angular, Vue.js y Ember.js.
- Bibliotecas como JQuery, Lodash y Ramda.
- Preprocesadores de CSS como Sass, Less y PostCSS.
- Herramientas de construcción como Webpack, Rollup y Gulp.
- Frameworks de CSS como Bootstrap, Tailwind CSS y Material-UI.

- Herramientas de testing como Jest, Enzyme y Cypress.

Estas tecnologías [19] se utilizan para crear la interfaz de usuario, manejar eventos, animaciones, y efectos visuales, así como para conectar con APIs y servicios backend.

Algunas tendencias actuales en frontend incluyen el uso de componentes, arquitecturas basadas en micro-frontends y la creciente importancia de la accesibilidad y el rendimiento.

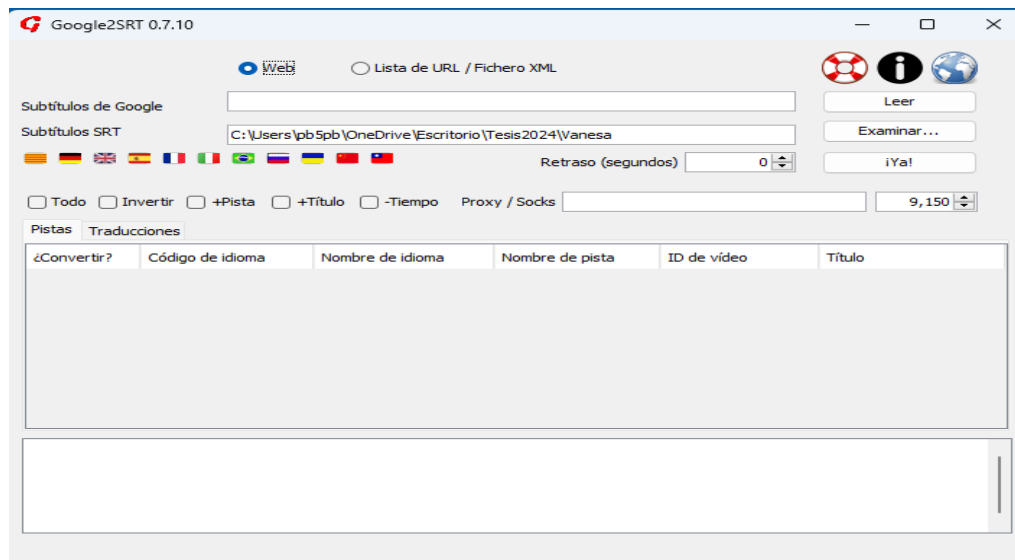
## CAPÍTULO 3. DISEÑO Y PRUEBAS DEL SISTEMA

En este proyecto de detección de muletillas (fillers) se integraron diversas herramientas como se muestra en el modelo del proyecto especificado en la figura 1.1 descrita en el capítulo 1. A continuación se describe el proceso de integración de la tecnología y las pruebas del funcionamiento del sistema web para reconocer muletillas en el idioma español.

### 3.1 TRABAJANDO CON ARCHIVOS SRT

Ubicar el video o audio a obtener su análisis para la detección de muletillas preferentemente de Youtube (www.youtube.com).

1. Copiar la URL del video o audio y pegar en el programa previamente instalado Google2SRT, este software está disponible para descargar en <https://sourceforge.net/projects/google2srt/>. Este software (ver Figura 3.1) permite generar de forma automática el archivo de transcripción del video o audio.



**Figura 3.1.** Software Google2SRT para generar archivos .srt

La transcripción es generada considerando el formato de estos archivos. Los archivos SubRip (SubRip Text - SRT) son nombrados con la extensión .srt, y contienen líneas que están formateadas con texto plano en grupos separados por una línea en blanco. Los subtítulos están numerados secuencialmente que inician por el valor de 1.

Es importante hacer notar que este software utiliza java con una versión superior a 1.6 en el jdk (java developed kit). Además, este tipo de archivos utilizan un formato de código de tiempo especificado en horas: minutos: segundos, milisegundos con unidades de tiempo fijas a dos dígitos rellenos con ceros y fracciones fijas a tres dígitos rellenos con ceros (00:00:00,000). En la figura 3.2. se muestra un ejemplo de archive SRT de la transcripción de un video.

```
1
00:00:07,269 --> 00:00:09,838
el experto

2
00:00:09,839 --> 00:00:15,118
independiente

3
00:00:15,118 --> 00:00:17,820
el portal

4
00:00:17,820 --> 00:00:19,419
pp

5
00:00:19,419 --> 00:00:23,739
en japonés se dice así si éstos
serán homologados con autoridad que

6
00:00:23,739 --> 00:00:25,900
produce desconocer una verdad

7
00:00:25,899 --> 00:00:28,559
me hace esclavo de una mentira

8
00:00:28,559 --> 00:00:32,969
o que conocer una verdad a medias es una
```

**Figura 3.2.** Ejemplo de un archive .srt

2. Los archivos .srt también pueden ser descargados en línea desde internet, existen diferentes páginas donde se puede realizar este proceso, para lo cual debe seguir los pasos:
  - a. Ubicar el archivo de voz o video por ejemplo desde Youtube
  - b. Ingresar a la url: <https://www.downloadyoutubesubtitles.com/es>
  - c. Colocar la dirección del archivo a descargar como se indica en la Figura 3.3.



Figura 3.3 Página en internet para descargar la transcripción de un video en youtube

- d. Posteriormente en la Figura 3.4 se indica que el sitio web permite modificar algunos formatos para la descarga del archivo .srt



Figura 3.4 Descarga de archivos srt con <https://www.downloadyoutubesubtitles.com/es>

### 3.2 APLICACIÓN WEB PARA EL CÁLCULO DE FRECUENCIAS

Una vez obtenido el archivo SRT se creó un programa en web con el lenguaje PHP + HTML + CSS para analizar el texto y obtener las frecuencias de aparición de cada palabra, posteriormente se comparó con un repositorio de algunas de las palabras consideradas como muletillas en el lenguaje español.

### 3.3 HERRAMIENTAS PARA EL DESARROLLO DE LA APLICACIÓN

Para el desarrollo de la aplicación web (prototipo) son necesarias algunas herramientas tecnológicas que nos permiten la funcionalidad del sistema, entre las herramientas indispensables tenemos:

#### **Servidor web**

El servidor web [19] es un sistema que almacena, procesa y entrega contenido a través de Internet utilizando el protocolo HTTP (o HTTPS). Su función principal es responder a las solicitudes de los navegadores web y enviar de vuelta los archivos solicitados, como páginas HTML, imágenes y otros recursos, entre sus principales funciones se encuentran:

- Guardar archivos estáticos (como HTML, CSS y JavaScript) y dinámicos (generados por lenguajes de programación como PHP o Python).
- Recibir y procesar las solicitudes de los clientes (navegadores) y devuelve la respuesta adecuada.
- Ejecutar scripts y consultas a bases de datos para generar contenido personalizado.
- Implementar protocolos de seguridad, como SSL/TLS, para cifrar la información y proteger datos sensibles.
- Distribuir la carga entre varios servidores en caso de alta demanda, mejorando la disponibilidad y el rendimiento.
- Gestionar registros de las solicitudes y el tráfico del sitio, lo que permite análisis y optimización.

Para el desarrollo de este proyecto se utilizó el servidor web apache de código libre, el servidor web Apache, oficialmente conocido como Apache HTTP Server, es uno de los servidores web más populares y utilizados en el mundo. Desarrollado y mantenido por la Apache Software Foundation, es conocido por su flexibilidad, potencia y extensibilidad. Y en particular se utilizó la herramienta WampServer (Figura 3.5) que es un paquete de software que permite a los desarrolladores web crear y gestionar aplicaciones web en un entorno local. WAMP es un acrónimo que significa Windows, Apache, MySQL y PHP. Este software es especialmente útil para quienes quieren desarrollar en un entorno similar al de producción sin necesidad de un servidor externo.



**Figura 3.5.** Herramienta Wampserver para la aplicación web (<https://www.wampserver.com/en/>)

WampServer incluye el servidor web (Apache) que maneja las solicitudes HTTP [6, 7] y entrega contenido web. MySQL para la gestión de bases de datos que permite almacenar y gestionar datos para aplicaciones web y el lenguaje de programación PHP del lado del servidor que se utiliza para crear contenido dinámico y gestionar interacciones con bases de datos.

## Lenguaje PHP

PHP (Hypertext Preprocessor) [6] es un lenguaje de programación de código abierto, ampliamente utilizado para el desarrollo web del lado del servidor. Es especialmente popular por su facilidad de uso y su capacidad para integrarse con bases de datos, lo que lo hace ideal para crear aplicaciones web dinámicas y sitios web interactivos. Además, PHP se utiliza para crear páginas web que interactúan con bases de datos y ofrecen contenido personalizado, para la Gestión de Contenido (CMS) como WordPress, Drupal y Joomla y PHP puede ser utilizado para construir servicios web. PHP se caracteriza por:

- Ser gratuito y tiene una gran comunidad de desarrolladores que contribuyen a su evolución y soporte.
- Tener sintaxis sencilla, lo que lo hace accesible para principiantes y desarrolladores experimentados.
- Integrar fácilmente con sistemas de gestión de bases de datos como MySQL, PostgreSQL y SQLite, lo que permite gestionar datos de manera eficiente.
- Por funcionar en diferentes sistemas operativos, incluidos Windows, Linux y macOS.
- Ofrecer un gran número de extensiones y bibliotecas que permiten añadir funcionalidades adicionales.
- Desarrollar aplicaciones rápidas debido a su simplicidad y a herramientas como frameworks (Laravel, Symfony, CodeIgniter), permite un desarrollo ágil de aplicaciones.

## HTML y CCS

El lenguaje de marcas HTML (Hypertext Markup Language) y las hojas de estilo en cascada CSS (Cascading Style Sheets) son dos tecnologías fundamentales en el desarrollo web, que trabajan juntas para crear y estilizar páginas web.

Usar HTML [19] junto con CSS en aplicaciones web es fundamental para crear experiencias de usuario atractivas y funcionales. HTML proporciona la estructura básica del contenido, permitiendo organizar textos, imágenes y enlaces de manera lógica y accesible. Por otro lado, CSS permite estilizar ese contenido, controlando aspectos como colores, fuentes, márgenes y disposición en la página. Esta separación entre estructura y presentación no

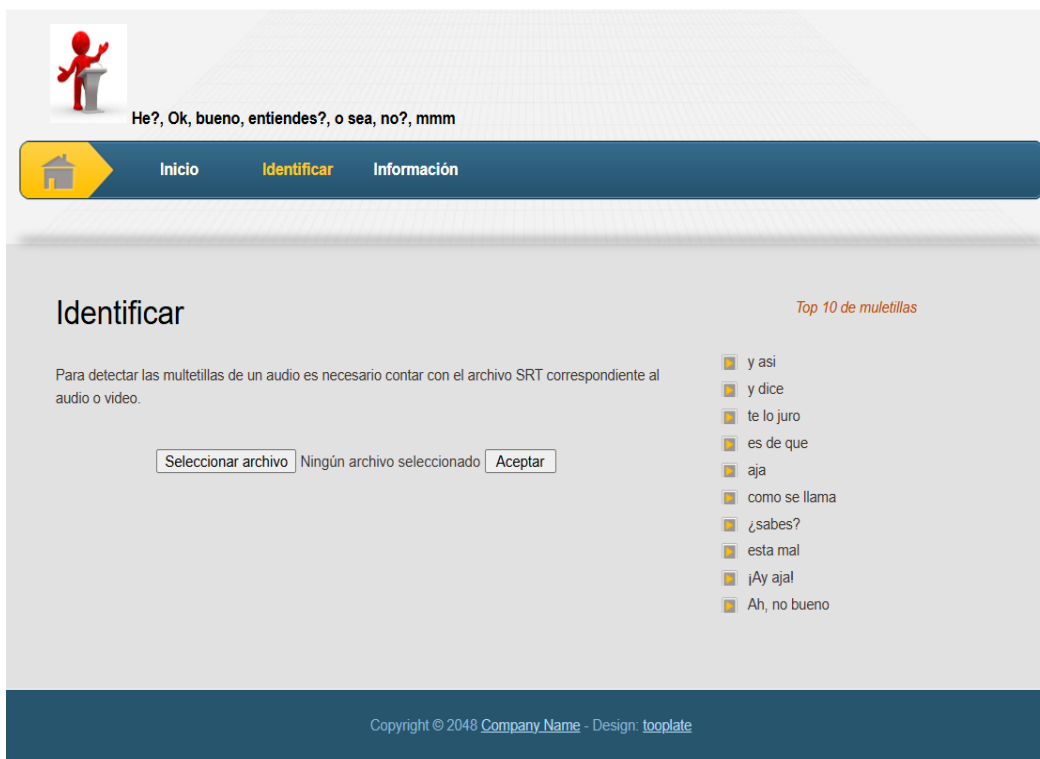
solo facilita el mantenimiento del código, sino que también permite que las aplicaciones web sean más responsivas y adaptables a diferentes dispositivos. En conjunto, HTML y CSS son esenciales para desarrollar sitios web que no solo sean visualmente atractivos, sino también intuitivos y fáciles de navegar, mejorando así la experiencia general del usuario.

En la figura 3.6 se muestra la pantalla principal del sistema web en ejecución para la detección de las frecuencias de las palabras.



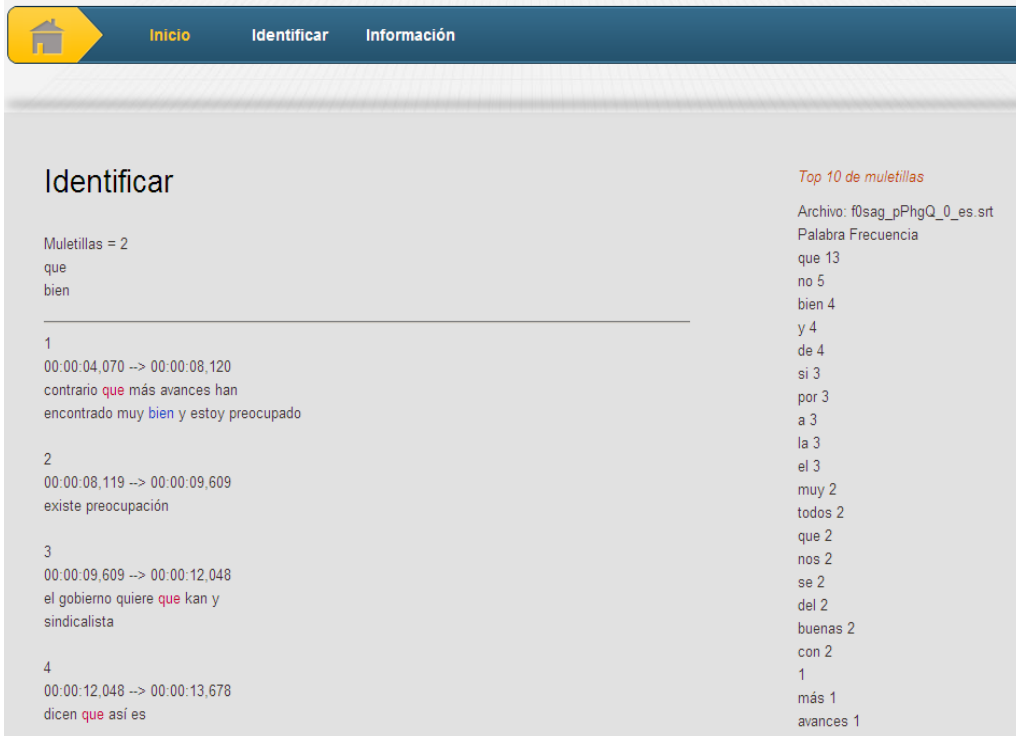
**Figura 3.6.** Pantalla principal del programa para la detección de muletillas

En la Figura 3.7 se muestra la pantalla de resultado al elegir la opción de “identificar” que permite subir un archivo con el formato .srt para aplicar la identificación de muletillas basado en las frecuencias.



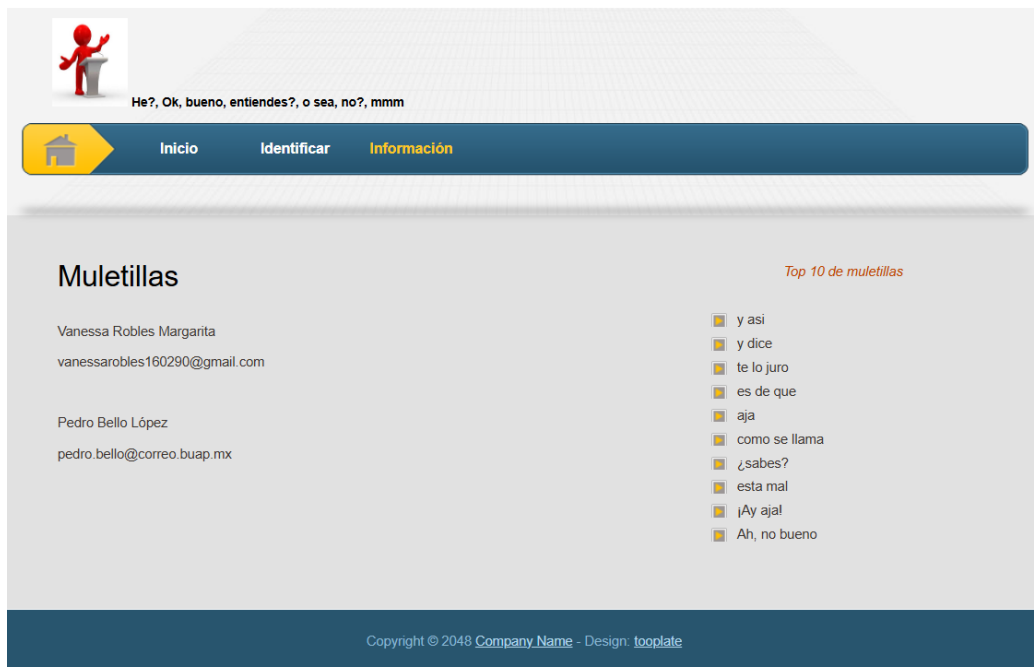
**Figura 3.7.** Pantalla principal del programa para la detección de muletillas

Al seleccionar el archivo *“f0sag\_pPhgQ\_0\_es.srt”* el sistema realiza el cálculo de las muletillas en el texto utilizando el método basado en frecuencias, la figura 3.8 muestra el resultado generado, de lado derecho se describe la frecuencia por palabra, mientras que en la parte central se indica el texto del archivo .srt pero con las principales muletillas identificadas, para lo cual se compara el 10% de las palabras con frecuencia más alta contra las palabras en el repositorio de muletillas. Si una o más palabras de alta frecuencia están dentro de la lista de palabras consideradas como muletillas, entonces se marcan en el archivo SRT con un color diferente, se utilizó rojo, azul, verde, amarillo y morado para las cinco muletillas más significativas. En la misma figura 3.7 se muestra un ejemplo de ejecución del programa donde se marcan con colores las muletillas encontradas., como se puede notar se detectaron 2 palabras que probablemente pueden ser muletillas, en rojo se marca la palabra *“que”*, la cual tuvo una frecuencia de 13 y la palabra *“bien”*.



**Figura 3.8.** Ejemplo de detección de muletillas

La última opción del sistema web creado se muestra en la figura 3.9 y es referente a los datos de contacto.



**Figura 3.9** Página de contacto del sistema web de detección de muletillas

### 3.3 REINCORPORACIÓN DEL ARCHIVO SRT AL VIDEO

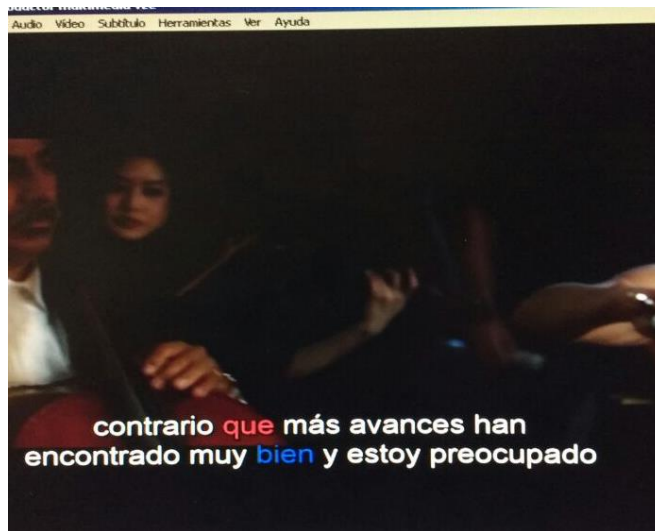
Si los usuarios desean visualizar los cambios aplicados al archivo SRT con las marcas de las muletillas se debe instalar herramienta VLC media Player [8]. Disponible en la página: [https://www.videolan.org/vlc/index.es\\_MX.html](https://www.videolan.org/vlc/index.es_MX.html).

En la figura 3.10 se muestra la página para descargar VLC media player. VLC media Player es un reproductor multimedia gratuito y de código abierto, conocido por su versatilidad y capacidad para reproducir una amplia variedad de formatos de audio y video. Desarrollado por el proyecto VideoLAN, VLC es compatible con múltiples plataformas, incluidos Windows, macOS, Linux, Android e iOS.



**Figura 3.10.** Página de descarga de VLC media player

En la figura 3.11. Se muestra en rojo y azul las muletillas detectadas que fueron reintegradas el video original a través de su archivo de transcripción SRT, utilizando la herramienta VLC media player. Como podemos notar



**Figura 3.11** Detección de las muletillas con la incorporación del archivo srt al video original

## CONCLUSIONES

El problema de las muletillas no solo abarca en un problema generalizado que este tipo de herramientas puede ayudar a evitar este problema o al menos a darse cuenta qué uno cae en este tipo de defecto. Muchas veces al escuchar a un orador que incurre en el uso excesivo de muletillas distrae al oyente debido a que esta más pendiente de la siguiente aparición de la muletilla e incluso generalmente las cuenta. En diversos estudios de educación y psicología se asocia este problema a inseguridad, a nerviosismo o incluso a fobia al exponer frente a un público o a una cámara de video.

Así se creó un prototipo funcional en web que integra diferentes tecnologías con el objetivo de detectar muletillas de una audio o video en el idioma español utilizando principalmente frecuencias de palabras, cabe mencionar que la herramienta Google2SRT realiza la validación de si el archivo es viable para extraer la transcripción o no, esto debido a que no todos los videos o audios contienen tal archivo anexo de texto (SRT).

El sistema al ser una aplicación web es viable de estar disponible para los usuarios interesados en realizar algún estudio para detectar las muletillas en el idioma español, por lo que este tipo de herramientas son de gran utilidad desde la perspectiva de la investigación puesto que permite realizar diferentes pruebas para reconocer las muletillas en el idioma español de un orador.

La integración de diversas herramientas puede servir como apoyo a los estudiantes de la facultad como incentivo para realizar trabajos de investigación similares, ya que las técnicas de procesamiento de audio y video son muy útiles para afrontar diversos problemas de la vida real y este es un caso muy específico. Con este trabajo podemos alentar al estudio en profundidad para generar herramientas integrales que resuelvan el problema de manera favorable.

Como trabajo a futuro se puede realizar un sistema para detección de muletillas ahora de idioma inglés. También se podría generar una aplicación móvil que este grabando en vivo a un orador y que al final del discurso detecte las muletillas encontradas.

## BIBLIOGRAFÍA

- ❖ [1] Barrie W. Jervis, Emmanuel C. Ifeachor (1993). Digital Signal Processing: A Practical Approach, AbeBooks.
- ❖ [2] Real Academia Española, consultado en abril de 2024 en la página <https://dle.rae.es/muletilla>
- ❖ [3] Jiménez, Arias, Maria, E, (2015). Uso excesivo de muletillas y comodines en las exposiciones científicas orales. MEDISAN vol.19 no.4 Santiago de Cuba.
- ❖ [4] ARichard Boulanger and Victor Lazzarini (2010). The Audio Programming Book, The MIT Press
- ❖ [5] Sourav Ghosh (2023). Building Low Latency Applications with C++. Develop a complete low latency trading ecosystem from scratch using modern C++, Packt Publishing
- ❖ [6] Welling, Luke y Thomson Laura. (2009). Desarrollo Web con PHP y MySQL Editorial Anaya.
- ❖ [7] Gutiérrez, A. y Bravo, G. (2005). PHP5, Alfaomega Ra-MA, México.
- ❖ [8] Lease, Matthew, Johnson, Mark. (2006). Early deletion of fillers in processing conversational speech. NAACL-Short '06: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers. Pages
- ❖ [9] Laiola, Guimarães R. Avegliano, P. Villa Real, Lucas, C. A. (2016). Lightweight and Efficient Mechanism for Fixing the Synchronization of Misaligned Subtitle Documents. Proceedings of the 2016 ACM Symposium on Document Engineering. Pages 175-184.
- ❖ [10] Yankelovich, Nicole. Lai, Jennifer (1999). Designing speech user interfaces. CHI EA '99: CHI '99 Extended Abstracts on Human Factors in Computing Systems. ACM. Pages 124-125.
- ❖ [11] Dethlefs, Nina. Hastie, Helen. Rieser, Verena. Lemon, Oliver. (2012). Optimising incremental generation for spoken dialogue systems: reducing the need for fillers. INLG '12 Proceedings of the Seventh International Natural Language Generation Conference Pages 49-58.
- ❖ [12] Kawashima, Hiroaki. Nishikawa, Takeshi. Matsuyama, Takashi. (2008). Visual filler: facilitating smooth turn-taking in video conferencing with transmission delay. CHI EA '08: CHI '08 Extended Abstracts on Human Factors in Computing Systems. Pages 3585-3590.
- ❖ [13] Chiara. Mejorar a hablar, consultado en marzo 2024 a través de la página <https://123emprende.com/blog/chiara-la-app-para-aprender-a-hablar-en-publico/>

- ❖ [14] Mejorar la comunicación sin muletillas, consultado en abril de 2024, en la página <https://blog.hubspot.es/marketing/apps-para-mejorar-comunicacion-representantes-de-ventas>
- ❖ [15] Sergio Adrian Martin, Martin Sergio Adrián (2011). Procesamiento Digital de Imágenes, Editorial Academica Espanola
- ❖ [16] Roberto Rodríguez Morales, Juan Humberto Sossa Azuela (2011). Procesamiento y Análisis Digital de Imágenes, RA-MA S.A. Editorial y Publicaciones.
- ❖ [17] Sommerville, I (2005). Ingeniería del software, Séptima edición, Pearson Educación. S.A. Madrid.
- ❖ [18] ROLDÁN, David, VALDERAS, Pedro, PASTOR, Óscar (2015). Aplicaciones Web – un Enfoque Práctico, Coedición: Alfaomega, Ra-Ma.
- ❖ [19] Anik Acharjee, (2020). Aprende las tecnologías web desde cero: Experiencia práctica en tecnologías web, Ediciones Nuestro Conocimiento.