SISTEMA DE RECONOCIMIENTO DE VOZ MEDIANTE APRENDIZAJE SUPERVISADO Y LA CORRELACIÓN DE PEARSON

A VOICE RECOGNITION SYSTEM THROUGH A SCRIPT LANGUAGE

Anel Ramírez Álvarez
Benemérita Universidad Autónoma de Puebla
anel.ramirez.al@gmail.com

Resumen

En la actualidad, el reconocimiento de voz se está convirtiendo en un proceso necesario, debido a los avances tecnológicos que permiten imitar la voz de alguna persona o personaje. El reconocimiento automático del habla o reconocimiento automático de voz es una disciplina de la inteligencia artificial, que tiene como objetivo permitir la comunicación hablada entre seres humanos y computadoras. Por tanto, en este trabajo de investigación se propone un sistema de reconocimiento de voz, que está basado en la biometría y utiliza la extracción de características distintivas de la voz, como la sonoridad, el volumen, el timbre, etc. Las cuales tienen su expresión análoga matemática, es decir, puede verse reflejada esta característica en cierto grado mediante una formula. Además, se hace uso de un entrenamiento de los datos para hacer el reconocimiento. Finalmente, un caso de estudio es presentado para probar la funcionalidad y precisión del sistema

así que se ingresará un audio específico de algún personaje y como salida tendremos el nombre de quien pertenece esa voz.

Palabras Claves: Biometría, extracción de características, entrenamiento, sistema de reconocimiento de voz, lenguaje script.

Abstract

Currently, voice recognition is becoming a necessary process, due to technological advances that allow imitating the voice of some person or character. Automatic speech recognition or automatic voice recognition is a discipline of artificial intelligence, which aims to allow spoken communication between humans and computers. Therefore, in this research work, a voice recognition system is proposed, which is based on biometrics and uses the extraction of distinctive characteristics of the voice, such as loudness, volume, timbre, etc. Which have their mathematical analogous expression, that is, this characteristic can be reflected to some degree by means of a formula. In addition, a training of the data is made in order to do the recognition. Finally, a case study is presented to test the functionality and precision of the system.

Keywords: Biometrics, feature extraction, training, voice recognition system script language.

1. Introducción

La biometría realiza estudios de reconocimiento de humanos basados en sus rasgos conductuales o físicos intrínsecos y particulares, es decir, identificando y verificando la anatomía o el comportamiento de una persona. En la literatura se ha establecido [Ortega, 2013]:

- **Biometría estática.** Define algo que el usuario es, un rasgo físico o anatómico, ya sea la huella dactilar, la cara, las líneas de la mano. Que son un rasgo característico y ´único en cada ser humano.
- **Biometría dinámica.** Se refiere a la conducta o comportamiento, algo que el humano hace, como su escritura, sus gestos, su forma de caminar, movimientos corporales o su propia voz.

Tomando como base la biometría es posible llevar a cabo estudios de reconocimiento de humanos fundamentados en rasgos conductuales o físicos intrínsecos y particulares de cada persona. De esta manera, la biometría permite autenticar individuos, mediante [About, 2011]:

- **La identificación.** Dice quién es una persona, dependiendo de sus características físicas o de su conducta.
- **La verificación.** Aclara si una persona es quien dice ser, partiendo de análisis biométricos y realizando comparaciones con otros candidatos.

La biometría tiene, principalmente, dos ámbitos de aplicación: la salud y la seguridad. Hasta ahora estos sistemas se separan en dos grandes módulos [About 2011]:

- Algo que el usuario sabe (contraseña).
- Algo que el usuario tiene (tarjeta personal).

En sistema de seguridad biométrico se introduce un tercer módulo, que analizará algo que el usuario es o hace.

Este trabajo de investigación, se centra en la biometría dinámica, que es algo que un ser humano hace. Además, se considera que en el desarrollo del reconocimiento automático del habla intervienen diversas disciplinas, tales como: la fisiología, la acústica, la lingüística, el procesamiento de señales, la inteligencia artificial y la ciencia de la computación [Tordera, 2011]. Por tanto, se presenta un sistema de reconocimiento de voz que determina quién es la persona o personaje que habla. Para ello, se consideran las voces de cinco personajes de “Los Simpson” y se utiliza el procedimiento para el reconocimiento de voz, que consta de [Big, 2017]: adquisición de voz, espectrograma y gráfica en el tiempo, extracción de características, entrenamiento y verificación.

Este artículo se encuentra organizado de la siguiente manera: Sección 2 explica el método que fundamenta el sistema de reconocimiento de voz. Sección 3 describe los resultados del sistema de reconocimiento de voz. Sección 4 presenta la discusión. Finalmente, Sección 5 detalla las conclusiones y el trabajo futuro.

2. Método

El sistema de reconocimiento de voz presentado en este artículo comienza desde la adquisición de los audios a procesar en el sistema, en este caso se tomaron muestras de audios de capítulos de los Simpson de 5 personajes distintos en

promedio con una duración de 3 segundos, tomando 18 muestras por personaje; 15 muestras de cada personaje, i.e. en total 75 muestras como la base de datos para el procesamiento y reconocimiento. Y 3 muestras de cada personaje, i.e. 15 muestras para las pruebas de reconocimiento. En la adquisición de los audios, para cada personaje se tomaron de un capítulo en específico, para cuidar cambios en las voces y sonoridad, ya que se pudo notar que esto pasa entre capítulos; así como también que no existieran ruidos de fondo o interferencia con otras voces.

Para segunda etapa, ver Figura 1, se realizó el procesamiento de las muestras en Python 3.8.2 primero adquiriendo la señal con el uso de la librería *scipy,* normalizamos y preparamos la señal para la extracción de parámetros en el tiempo y en la frecuencia.

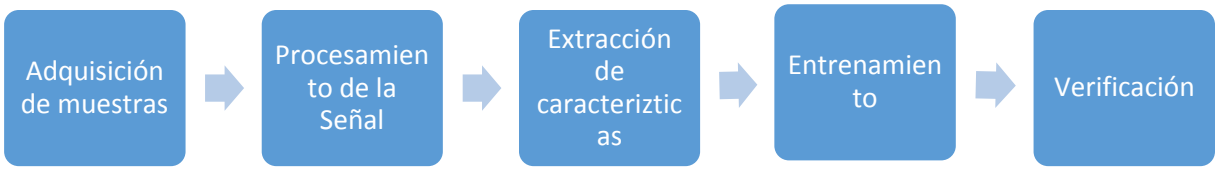


Figura 1. Sistema de reconocimiento de voz.

En la tercera etapa tenemos la extracción de características en dos dominios tiempo y frecuencia. Para la primera calcularemos la Energía y Raíz Media Cuadrática (RMS) de las muestras, por otro lado para la Frecuencia calcularemos el Centroide Espectral y el Mel-Frequency Cepstral Coecients (MFCC), ver Figura 2. Esto se realiza ya que la base de este trabajo consiste en Entrenar el Sistema con varias muestras de audio para agrupar los sonidos por similitud en base a descriptores o características particulares de cada personaje, lo cual nos servirá para clasificar en grupos a cada uno de ellos. Y en base a la incorporación de técnicas de *Inteligencia Artificial*, de análisis de metadatos obtenidos para las muestras de la base de datos [Big, 2017].

Posteriormente, se hace la clasificación juntando todas las clases de los 5 personajes y así poder entrar a la etapa de Evaluación. Esta última etapa, se basa en un algoritmo de Inteligencia Artificial para predecir a cuál de estas clases

pertenece el audio externo ingresado, terminando así nuestro Sistema de Reconocimiento de Voz.

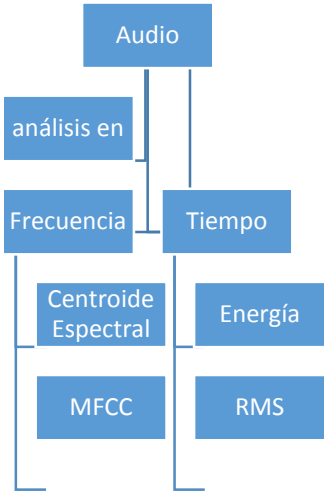


Figura 2. Tipos de análisis para la señal de audio.

2.1. Adquisición de la señal de audio

Como se mencionó anteriormente, el sistema de reconocimiento de voz, se debe entrenar con muestras de audio de los personajes que se deseen reconocer.

2.2. Procesamiento de la señal

La recuperación de información musical (*Music Information Retrieval*, MIR) [MIR, 2011], es una ciencia interdisciplinaria que busca extraer información de la música. Entre otras cosas, agrupa una serie de algoritmos que permiten obtener metadatos del audio, calculando valores que lo describan de alguna forma de interés. Estos valores se denominan descriptores o características de un sonido, haciendo que el usuario o compositor piense en términos de descriptores MIR [Li-Chung, 2017], ampliando así el lenguaje y posibilidades. De los ejemplos más comúnmente usados son: Beats Por Minuto o tiempo (BPM), el centro de gravedad espectral (*spectral centroid*), duración del archivo, contenido en alta o baja frecuencia, clave musical, la ubicación temporal de los comienzos de eventos musicales (*onsets*), cantidad de disonancia armónica e índice de complejidad espectral (*spectral complexity*) [Big, 2017].

Para el Procesamiento del audio en Python se usan las librerías numpy, scipy.io, matplotlib y scipy.fftpack; mientras para el espectrograma (véase la figura 3) plt.specgram(signalData,Fs=samplingFrequency). En el tratamiento de señales (conjunto de datos), se implementa la transformada de Furier Discreta (véase la figura 4); que da como resultados una secuencia de números complejos —parte real y parte imaginaria (véase la figura 5)—, análogo a un vector.

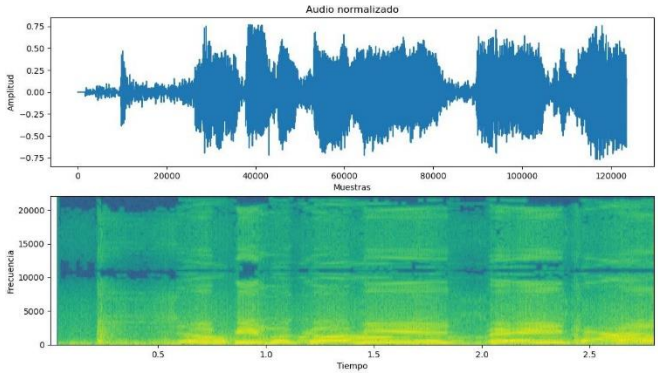


Figura 3. Audio y espectrograma.

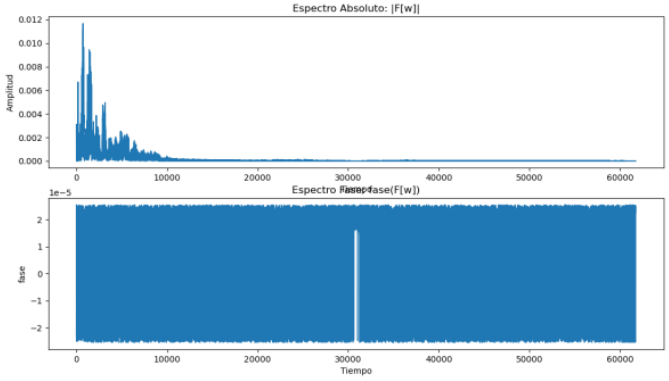


Figura 4. Transformada de Fourier: Absoluto y Fase.

2.3. Extracción de características

En esta fase se agrupan, previamente, los sonidos por similitud en base a descriptores, lo cual es posible hacer al incorporar técnicas de *Machine Learning* al análisis de los metadatos obtenidos para cada muestra de la base de datos. El sonido se agrupa de acuerdo a sus descriptores (véase la Tabla 1) como el timbre (centroide espectral, MFCC, etc.), características relacionadas con la dinámica (volumen en una señal acústica particular, el nivel medio, etc.) y características

relacionadas con el *pitch*. Se considera un sonido como un grupo de características. Cada característica tiene un valor numérico [Chu, 2012].

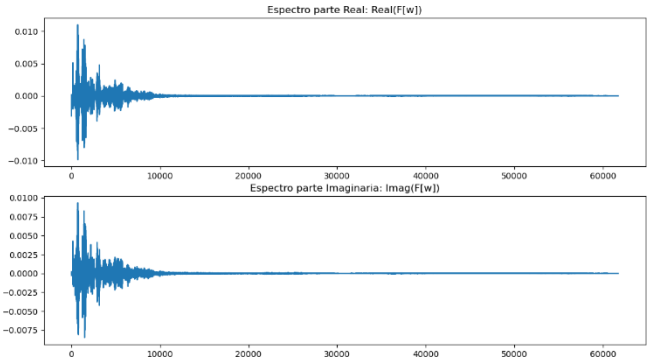


Figura 5. Parte real e imaginaria.

Tabla 1. Descriptores del sonido.

Perceptual	Sensorial	Física
intervalos sucesivos o simultáneos	pitch	frecuencia
tiempo (beat)	tiempo	duración
timbre (envolvente espectral)	timbre	espectro (centroide)
dinámica	volumen	intensidad

2.3.1. Características de la señal en el tiempo

A continuación se muestran la Envolvente, Energía y Root Mean Square (RMS); utilizadas en este artículo.

2.3.1.1 Envolvente.

La envolvente de una señal son los máximos locales de valores absolutos de la gráfica de audio normalizada. Esta grafica es importante para la posterior parametrización en el tiempo de dicha señal de audio, ver Figura 6.

2.3.1.2. Energía

La energía representa el volumen del audio, este es calculado a partir de *frames* desde el dominio del tiempo o la frecuencia. Si estamos con el dominio del tiempo lo obtenemos sumando todos los cuadrados de sus magnitudes, como lo indica la ecuación 1 para una señal continúa.

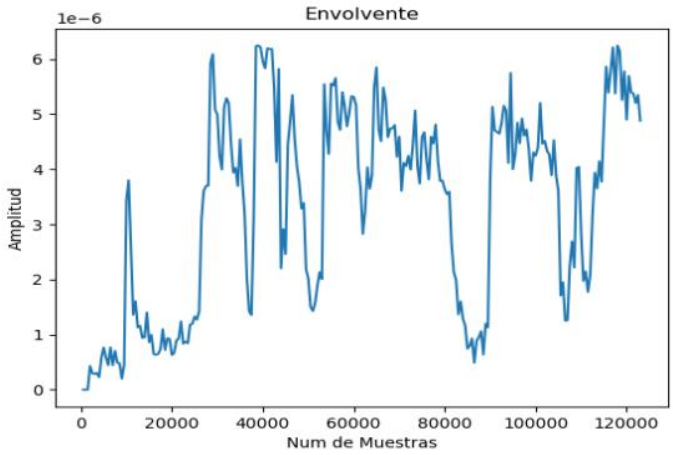


Figura 6. Envolvente de la señal de audio normalizada.

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2 \, dt \tag{1}$$

Donde t es el tiempo y x la magnitud de la señal en función del tiempo.

Para una señal discreta tenemos que:

$$E_x = \sum_{n=1}^{N} |x[n]|^2$$

(2)

Donde n es el número i-ésimo de la muestra (frame) y x la magnitud de la señal en función del núm. de frame. La sumatoria va desde la primera muestra hasta el N número de muestras total de la señal.

2.3.1.3. Raiz Media Cuadrada

La raíz media cuadrada o RMS por sus siglas en inglés representa la presión sonora del audio; en otras palabras, es otra forma de visualizar la energía. La presión sonora RMS es la raíz cuadrada del promedio ponderado de la Energía, se calcula con la ecuación 3.

$$RMS_x = \sqrt{\frac{1}{N^2} \sum_{n=1}^{N} |x[n]|^2} \tag{3}$$

2.3.1.4. Implementación Gráfica: Energía vs RMS.

Al considerar el sonido como un grupo de características, cada una tiene un valor numérico; como la energía y RMS son valores constantes estos se pueden expresan en una dimensión en una gráfica X vs Y, véase la figura 7. Para probar esta idea se propusieron tres audios grabados con distintas palabras perro, agua y casa: 'casa.wav', 'perro.wav' y 'agua.wav'.Aplicando las operaciones de Energía y RMS tuvimos los siguientes resultados:

Casa: Energía= 584.3195553522062, RMSx= 0.0006043175672567602

Perro: Energía= 830.0490110166623. RMSx= 0.0007202642791957782

Agua: Energía= 2517.8710373610616 RMSx= 0.0012544598034017126

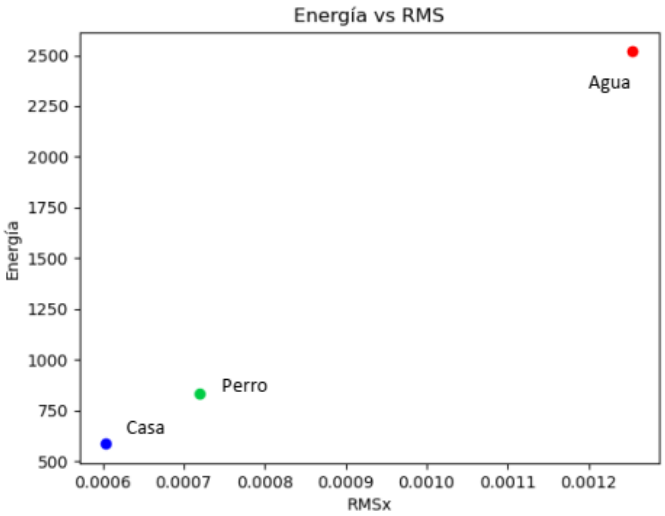


Figura 7. Características en el dominio del tiempo: Energía vs RMS.

Y se agregó un cuarto audio llamado 'prueba.wav' donde la palabra dicha fue *Perro*. Arrojando los siguientes datos:

Prueba: Energía= 732.068258819977, RMSx= 0.000676418998670562

Se graficaron los datos anteriores donde en las Ordenadas van los datos de Energía y en las Abscisas RMSx, ver Figura 8.

Se aprecia que se obtiene información confiable, ya que como se mencionó Perro fue la palabra de 'prueba.wav', y la palabra más cercana a ella fue Perro. Sin embargo, con más palabras en la base de datos o grabadas las mismas palabras

con un volumen más bajo se podrían generar resultados incorrectos. Por tanto, aun que la Energía y el RMS son parámetros muy necesarios a valorar, es necesario incluir más características que aporten información de los audios sin que estén sujetos a cambios tan vulnerables como lo es el volumen, esto se logra con parámetros el dominio de la frecuencia.

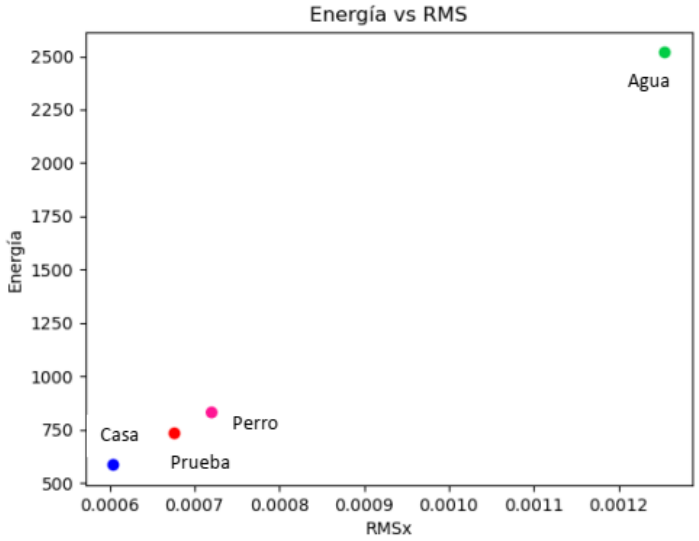


Figura 8. Características en el dominio del tiempo: Energía vs RMS.

2.3.2. Características de la señal en la frecuencia

Para la caracterización de parámetros en frecuencia aplicaremos la Transformada Rápida de Fourier, FFT por sus siglas en inglés y se tomará el ancho de banda de la voz humana de 1Hz a 4kHz como frecuencias a caracterizar.

2.3.2.1. Centroide Espectral

Es una característica que trata de definir la forma espectral de un sonido, indicando la parte más concentrada del espectro. Perceptualmente, se relaciona con la claridad que puede tener un sonido. Se calcula como la media ponderada de las frecuencias presentes en la señal, determinada mediante una transformada de Fourier, con sus magnitudes como los pesos, véase la ecuación 4.

$$Centroide = \frac{\sum_{n=0}^{N-1} f(n)x(n)}{\sum_{n=0}^{N-1} x(n)} \quad (4)$$

Donde x(n) representa el valor de frecuencia ponderado, o la magnitud, del número de elemento n, y f(n) representa la frecuencia central de ese elemento.

2.3.2.2. Escala de Mel

Usando la librería *librosa* y a partir de la Transformada de Fourier sacamos la escala de Mel de las señales de audio (véase la figura 9).

2.3.2.3. Mel-frecuency Cepstral Coefficients (MFCC)

MFCC es una característica que refleja la forma del espectro, de una manera más compleja. Es una representación de la magnitud espectral y se resuelve partiendo de la transformada del coseno del logaritmo de su magnitud espectral en una escala no lineal, llamada escala de Mel. La ecuación 5 muestra como el espectro completo xl[k] es multiplicado por un banco de filtros, una ventana. De forma que, cada frecuencia que sea dependiente de la escala Mel cambie. El objetivo es hacer más perceptiva la magnitud espectral del resultado de la FFT. Después se hace el logaritmo y por último la DCT. La extracción de coeficientes MFCC [Salomón, 2011] es la técnica de parametrización más utilizada en el área de verificación de locutor. El objetivo es tener una representación robusta para tener un modelo preciso del locutor.

$$mfcc_l = DCT\left(log_{10}\left(\sum_{k=0}^{N/2} |X_l(k)| H_l(k) \right) \right) \quad (5)$$

Donde $\|X_l(k)\|$ es la parte positiva de la magnitud espectral, $H_l(k)$ es el banco de filtros de la escala de Mel y $DCT(m) = \sum_{n=0}^{N-1} f(n)\cos(\pi/N(n+1/2)m)$.

2.3.2.4. Implementación Gráfica: MFCC vs Centroide Espectral

Después de calcular el Coeficiente de MFCC y el del Centroide Espectral, graficamos ambos parámetros y los sometimos a la misma prueba de X vs Y con audios. La palabra de 'prueba.wav' fue Agua, y como resultado la palabra más cercana a esta fue Agua (véase la figura 10), que es un resultado correcto, igual sucedió para otros casos de prueba.

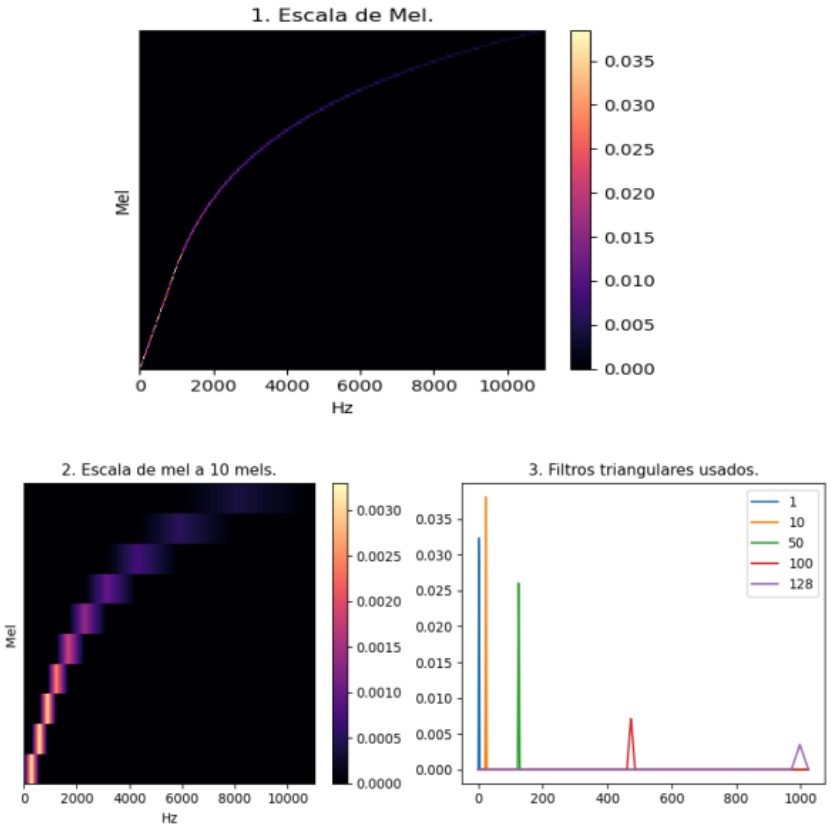


Figura 9. Características en el dominio del tiempo: Energía vs RMS.

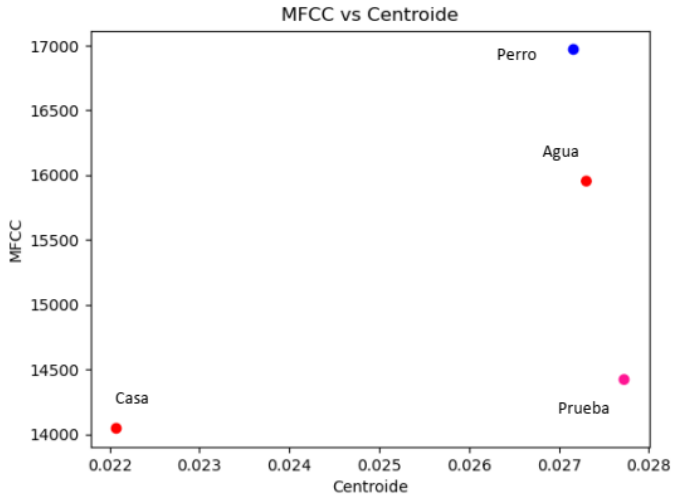


Figura 10. MFCC vs Cenrtoide Espectral

2.3.3. Entrenamiento del Sistema.

Para el entrenamiento del sistema se tomaron muestras de voz de 5 personajes de “Los Simpson”; Homero, March, Bart, Lisa y el Sr. Burns, tomando las muestras del mismo capítulo para cada personaje, esto con el fin de evitar cualquier cambio

en las grabaciones; en total se tomaron 90 audios, 15 para la base de datos del sistema y tres de cada uno para poner a prueba el sistema.

2.3.3.1. Entrenamiento: Características en el Tiempo

Se realizó el entrenamiento ingresando muestras de audio de 5 personajes, 15 por cada uno de ellos; así se marcan rangos distintivos de cada uno de ellos, que se cuantifican para un procesamiento de datos. En la Figura 11 se muestran los datos del personaje de Bart.

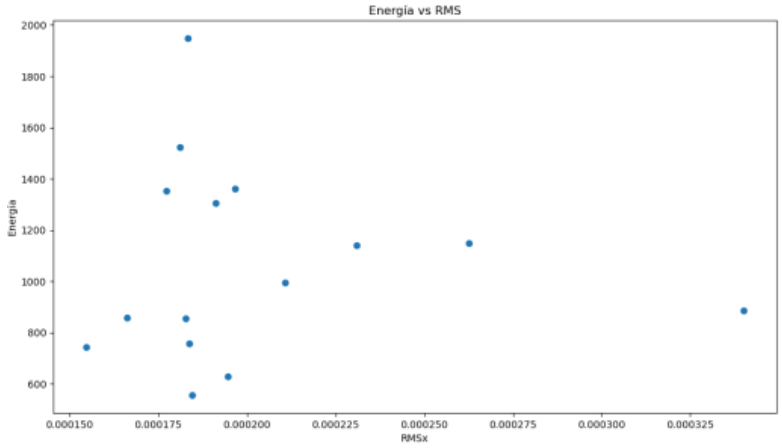


Figura 11. Resultados para el personaje Bart**:** Energía vs RMS.

2.3.3.2. Entrenamiento: Características en la Frecuencia

Ahora identificaremos las características en el dominio de la frecuencia de los audios, con los Coeficientes de MFCC y Centroide Espectral (véase la figura 12).

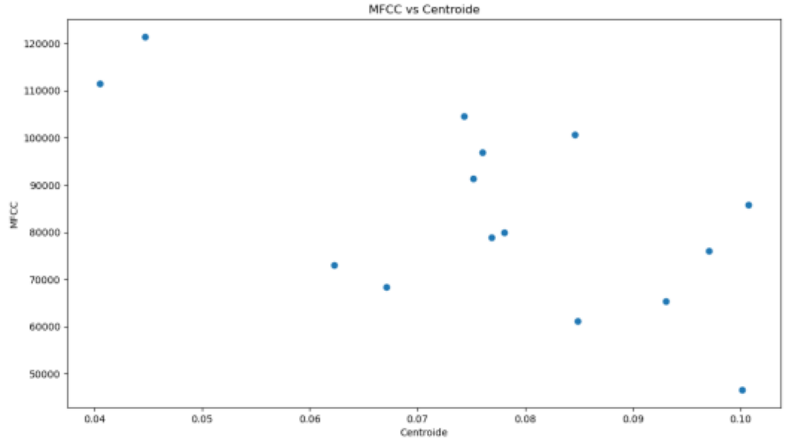


Figura 12. Resultados para el personaje Bart**:** MFCC vs Cenrtoide Espectral.

2.4. CLASIFICACIÓN

En la clasificación se consideran todas las muestras de los 5 personajes, según sus características para ver las zonas o rangos específicos en las que se encuentran. En la figura 13 se aprecia como las características de Energía (análogo al volumen de la voz) y RMS (análogo a la presión sonora) de cada uno de los personajes, los sitúan en zonas distintas en la gráfica. Para March ambas características son más bajas en el grupo, situándose así, en la parte inferior izquierda. Se visualiza como Lisa (valores rojos) se encuentra en una zona superior, después de azul está Bart, luego en verde Homero y finalmente, en la parte superior derecha, al Sr. Burns con valores más altos tanto en Energía como en RMS. En general, se establece que ambos criterios de Energía y RMS fueron muy efectivos, para distinguir parcialmente las zonas en las que se encuentra un personaje, con lo cual se tiene un aporte de información para detectar a quien podría pertenecer la voz, si agregáramos un audio extra de cualquiera de estos 5 personajes. Sin embargo, ya que los puntos se pueden encontrar muy cerca en los rangos de RMS [0.00015 – 0.00025] y de Energía [500 - 1500], esto podría generar ruido en esta zona, por tanto, es necesario incluir más parámetros a evaluar para una mayor certeza en los resultados. También se enfatiza que, a pesar de existir un comportamiento lineal, el parámetro que proporciona más información es la Presión Sonora (RMS) ya que presento mayor ritmo de cambio.

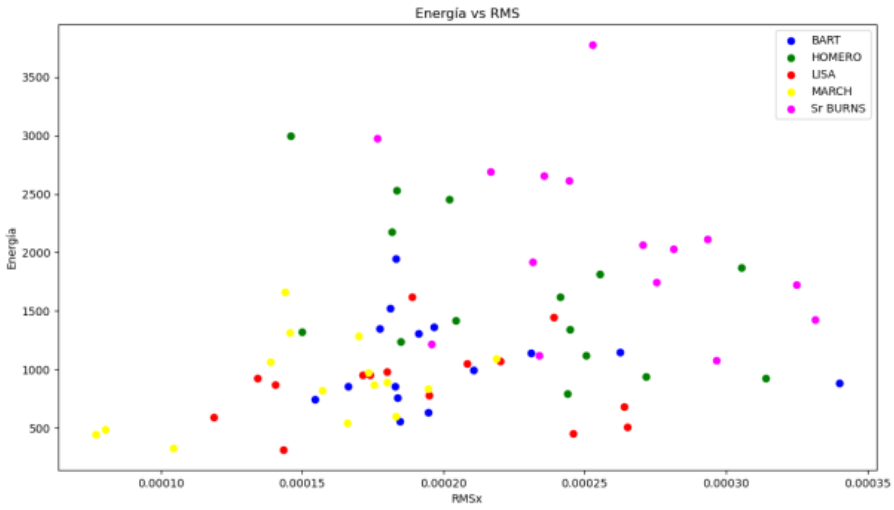


Figura 13. Energía vs RMS de todos los personajes.

En la Figura 14 se aprecia como para el parámetro MFCC (representación muy usada para identificación de la voz) y el Centroide Espectral (indicación la parte más concentrada de espectro) de cada uno de los personajes a evaluar, coloca a cada uno de ellos en distintas zonas de la gráfica. Donde para Homero y March los tenemos casi en la misma zona, luego en rojo a Lisa, después al Sr. Burns y en la parte Superior de la gráfica a Bart, a quien se puede encontrar disperso alrededor de toda la gráfica. En general, el MFCC proporciona mejor información que el Centroide ya que se ve un mayor ritmo de cambio por parte del MFCC.

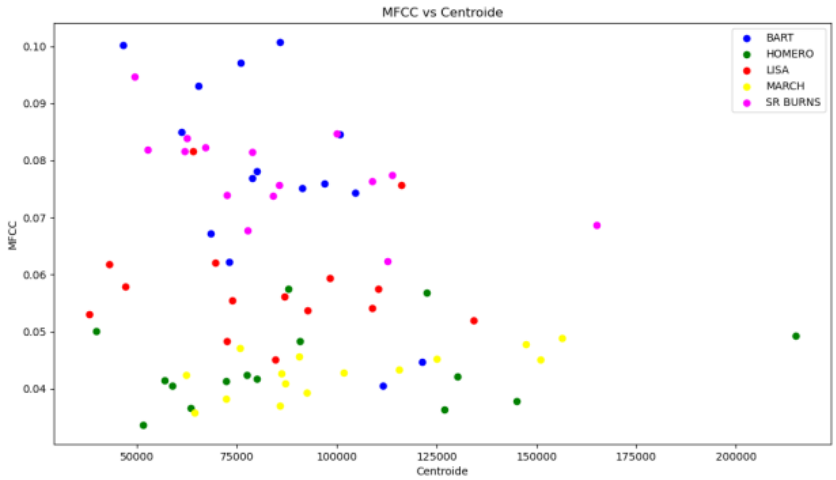


Figura 14. MFCC vs Centroide de todos los personajes.

Para *visualizar mejor las zonas de dispersión* de cada personaje se realizó una gráfica en 3D. Partiendo de las gráficas presentada en las figuras 13 y14, se seleccionaron las características Energía, RMS y MFCC, que mostraron mayor ritmo de cambio, para los 15 audios de todos los personajes, y así visualizar mejor dichas zonas características, véase Figura 15.

2.5. Evaluación

Con la clasificación por zonas de los personajes presentada en las Figuras 13 y 14; se puede proponer un criterio para determinar a quién de los personajes pertenece un audio externo al sistema (en función de la base de datos proporcionada al sistema).

Con lo cual se propone un algoritmo que nos devuelva el grado (porcentaje) de pertenencia a un conjunto de valores cercanos al audio externo ingresado a evaluar.

Con lo cual esté Sistema devolverá resultados en “Porcentajes de Confiabilidad”, obteniendo un porcentaje mayor si es más probable que le pertenezca a algún personaje en específico y porcentajes menores en proporción a que tan probable es que le pertenezca a otro personaje, siempre en función de la posición del sonido externo; téngase en cuenta que el ***Resultado con Mayor Porcentaje*** será tomado como la ***Respuesta Final del Sistema***. En la figura 16, se muestra un audio de prueba ingresado con los audios de todos los personajes, se observa que como el audio de prueba se sitúa en una zona en específico de la gráfica, por tanto, es importante sumar cada porcentaje de certeza de cada grafica para tener un total y final. Con lo cual el criterio de evaluación debe de estar en función de los datos y los resultados en forma de porcentaje.

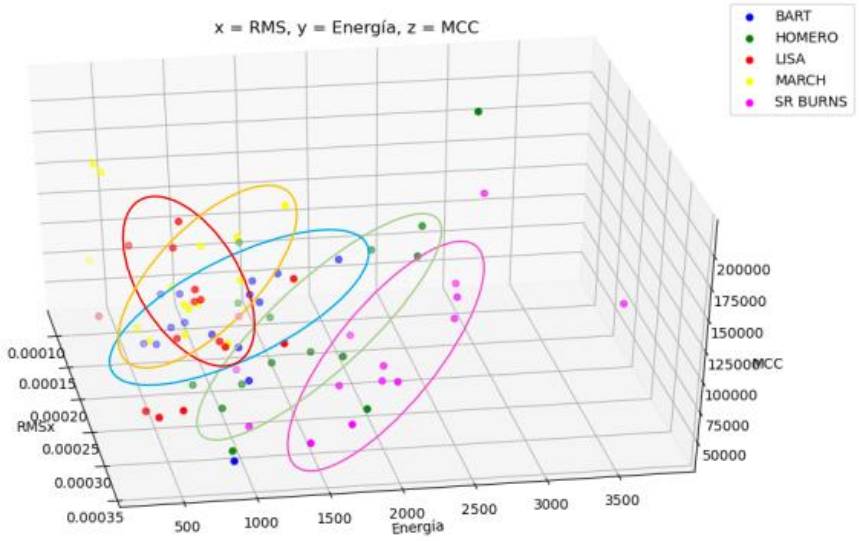


Figura 15. Grafica 3D de todos los personajes.

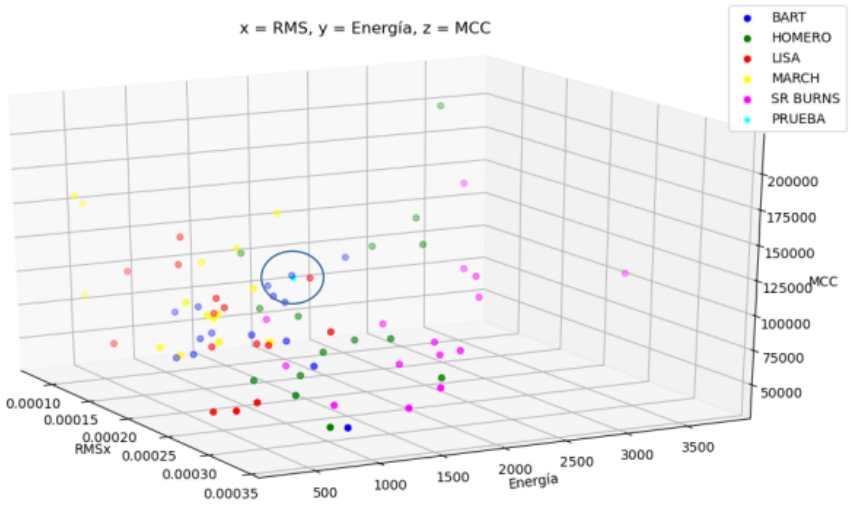


Figura 16. RMS vs Energía vs MFCC con audio de prueba.

2.5.1 Criterio de Evaluación: Algoritmo k-Vecinos Más Cercanos y Correlación de Pearson

Se propone resolver el problema planteado con un algoritmo de aprendizaje supervisado, ya que contamos con los datos de entrenamiento y pretendemos deducir o predecir el resultado en función de dichos datos. El algoritmo propuesto para resolver este problema es el K- vecinos más cercanos o k-NN, K-Nearest Neighbors por sus siglas en inglés. K vecinos más cercanos es uno de los algoritmos de clasificación más básicos y esenciales en Machine Learning. Pertenece al dominio del aprendizaje supervisado y encuentra una aplicación intensa en el reconocimiento de patrones, la minería de datos y la detección de intrusos. Este algoritmo nos ayudara a predecir a que conjunto de datos o clustering pertenece un dato ingresado sin etiquetas, es decir, sin tener información adicional a los clustering.

Se trata de un algoritmo muy sencillo, pero altamente eficiente y que arroja mejores resultados a comparación de algoritmos más complejos.

Cabe señalar que la fase de formación mínima de KNN se realiza tanto a un coste de memoria, ya que debemos almacenar un conjunto de datos potencialmente enorme, como un coste computacional durante el tiempo de prueba, ya que la clasificación de una observación determinada requiere un agotamiento de todo el conjunto de dato. En la práctica, esto no es deseable, ya que normalmente queremos respuestas rápidas.

Este problema puede ser solucionado con algoritmos más eficientes que reduzcan el coste computacional, y ya que la propuesta en este artículo también parte del uso de una base de datos pequeña buscamos dar una solución que no requiera un conjunto de datos enorme.

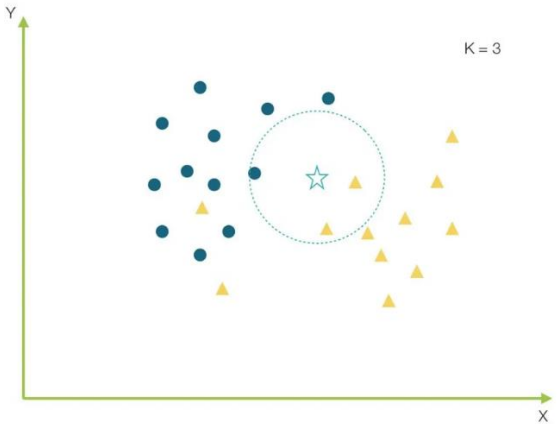


Figura 31. Clasificación por K-NN para k = 3.

Ahora bien supongamos que la estrella en la Figura 31, es el punto el cual se necesita predecir. Primero, se encuentra el número k de puntos más cercano a Z, es decir 3 para este caso, y al determinar que tenemos 2 triángulos y 1 circulo dirá que es 67% probable que sea un triángulo y un 33% un circulo con lo que concluirá que el valor predicho para la estrella es un *triángulo*.

Para determinar que valores son los más cercanos necesitamos calcular *distancias,* la opción elegida en este artículo es la fórmula para la distancia entre dos puntos en el plano cartesiano ver Figura 32, ecuación 6. Pero para poder aplicar esta fórmula antes se pasaron a la misma escala ambos ejes en las gráficas 13 y 14.

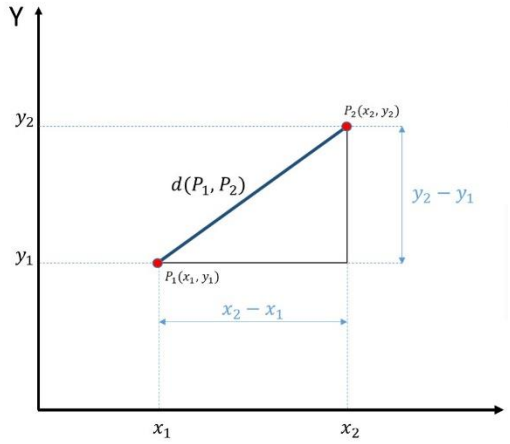


Figura 32 Distancia entre dos puntos

$$d(P_1, P_2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (6)$$

En resumen, KNN tiene los siguientes pasos básicos:

- Calcular distancias.
- Encontrar sus vecinos más cercanos (las K distancias mínimas).
- Calcular grados de pertenencia a las etiquetas (asignar porcentajes).

2.5.1.1 ¿Cómo se decide el número de vecinos en KNN?

Ya que conocemos cómo funciona este algoritmo, es momento de saber cómo se define K. El número de vecinos K es un hiperparámetro que se debe elegir en el momento de la construcción del modelo. Puedes pensar en K como una variable de control para el modelo de predicción. La investigación ha demostrado que no existe un número óptimo de vecinos que se adapte a todo tipo de conjuntos de datos. Cada conjunto de datos tiene sus propios requisitos. [9]

Pero gracias a que tenemos dos clasificaciones, una el tiempo y otra en la frecuencia, en este artículo se propone el uso de la ***Correlación de Pearson*** como parámetro para calcular el valor de k para cada prueba, en vez de usar un valor fijo de K que es lo usual en el uso de K-NN como algoritmo de predicción. Se propone la siguiente ecuación para la elección de k:

$$k(r) = \max(r_1, r_2, r_3, \dots, r_n)) \quad (7)$$

Donde r es la Correlación de Pearson, ecuación 8, que puede tomar valores de 0 a 1 donde a mayor valor mejor relación entre los datos.

$$r_i = \frac{N \sum_{l=0}^{N} xy - (\sum_{l=0}^{N} x)(\sum_{l=0}^{N} y)}{\sqrt{(N \sum_{l=0}^{N} x^2 - (\sum_{l=0}^{N} x)^2)(N \sum_{l=0}^{N} y^2 - (\sum_{l=0}^{N} y)^2)}} \quad (8)$$

Donde N es el número total de datos, X el primer arreglo de datos y Y el segundo arreglo de datos a comparar respecto a X.

Procedimiento de la Evaluación:

- *Aplicamos el Algoritmo K-NN*: Calcular la distancia del valor a evaluar con los demás datos y después calcular los K vecinos más cercanos. Este paso se hace para los datos en el Dom(t) y Dom(f), variando K desde el 10% de los datos (k = 8 para este caso) y 20% (k = 15) en pasos de 1, (k[8,15]). Los

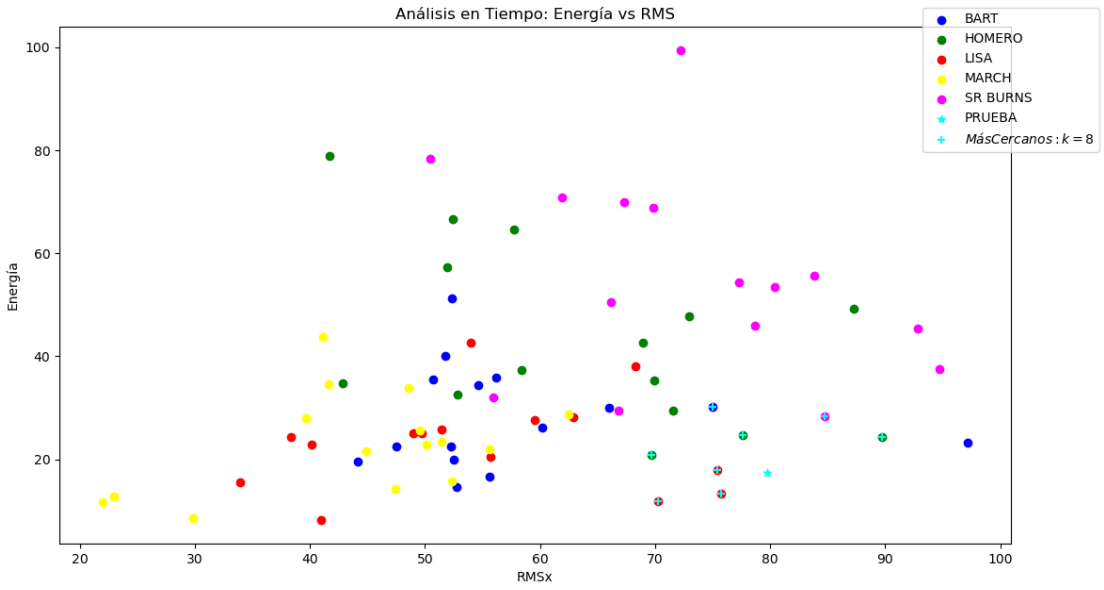
porcentajes se fijaron por experimentación de ver como se comportaban los datos para dichos valores de K.

- *Calcular Correlación de Pearson:* Del paso anterior tendremos 7 cadenas para calcular la correlación donde “x”, ver ecuación (8), serán los datos en el Dom(t) y “y” serán los datos en el Dom(f). Con lo cual tendremos una cadena de 7 correlaciones en función de variar K siete veces de 8 a 15, para este caso.
- *Elección de K:* de la cadena de tamaño 7 elegiremos usar la K con el valor de r máximo de la cadena, ver ecuación 7.
- *Evaluar:* se asignan porcentajes en función del número de datos para cada evaluación (asignación de etiquetas), el porcentaje máximo de la evaluación será nuestro Resultado Final de Predicción.

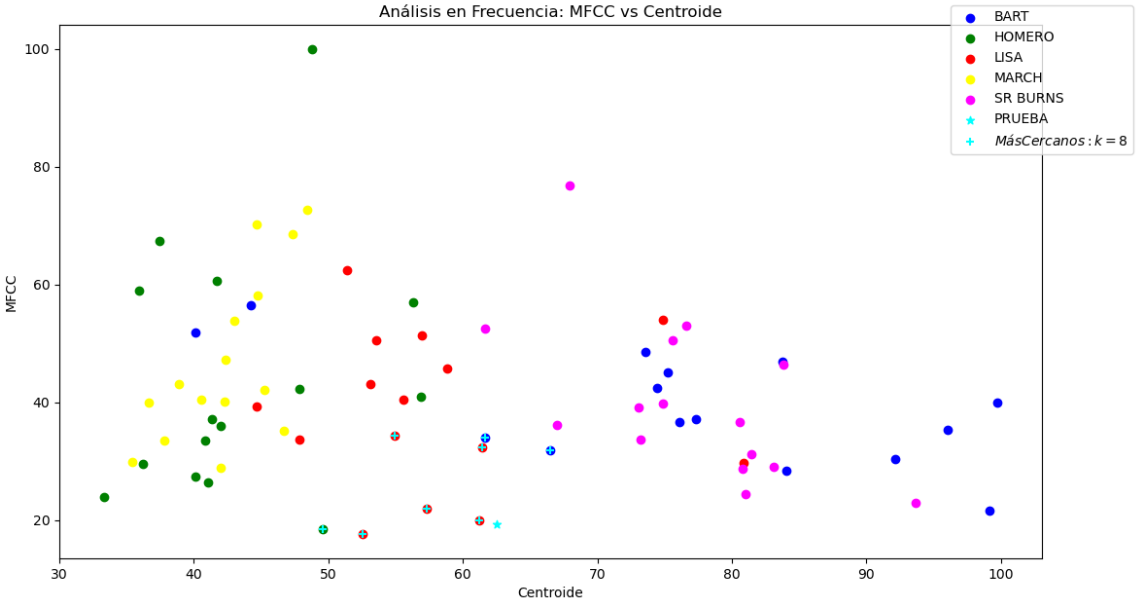
3. Resultados

A continuación, se muestran 2 pruebas de las 15 a las que se sometió el sistema de reconocimiento de voz.

La primera prueba (que sabemos pertenece a la voz de Lisa) arrojo que la mejor correlación entre los datos en el Dom(t) y Dom(f) según k, es para k = 8, ver figura 33 a) y b). Donde los puntos marcados por “+” son los 8 más cercanos a los de la prueba (estrella en la figura). Resultados al evaluar con los datos más cercanos arrojados por el algoritmo K-NN en función del K como mejor opción por la Correlación de Pearson son (Figura 33):



a) Dom(t): K = 8 vecinos más cercanos



b) Dom(f): k = 8 vecinos más cercanos

Figura 33. Prueba Audio Uno.

Los resultados en esta prueba de Correlaciones es:

r(k[8,15]) = [0.6469966392206304, 0.4587596979470412, 0.4225771273642583, 0.42059978233966894, 0.37234154758203314, 0.5086989586601017, 0.3653325655726347]

Con la ecuación 7 elige k = 8 como mejor opción con r = 0.6469966392206304

Parametrizando estos resultados en graficas de Pastel, para k = 8 como mejor opción por correlación:

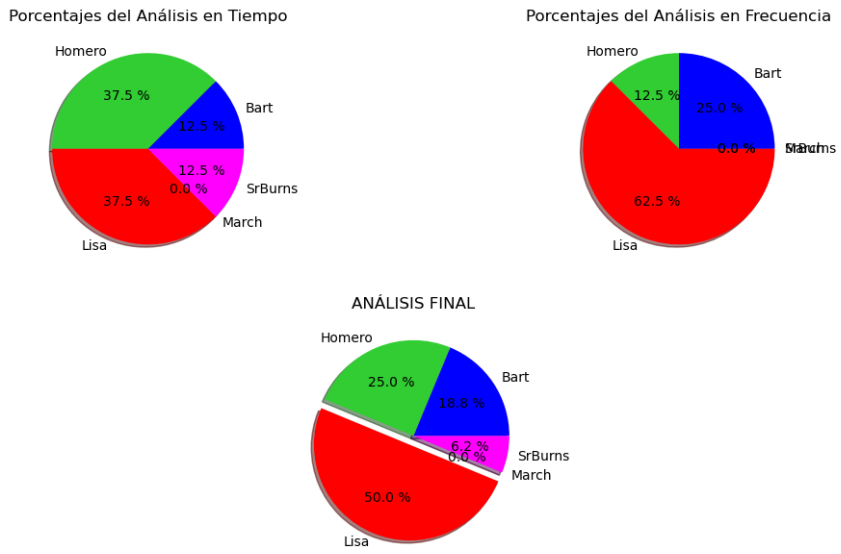
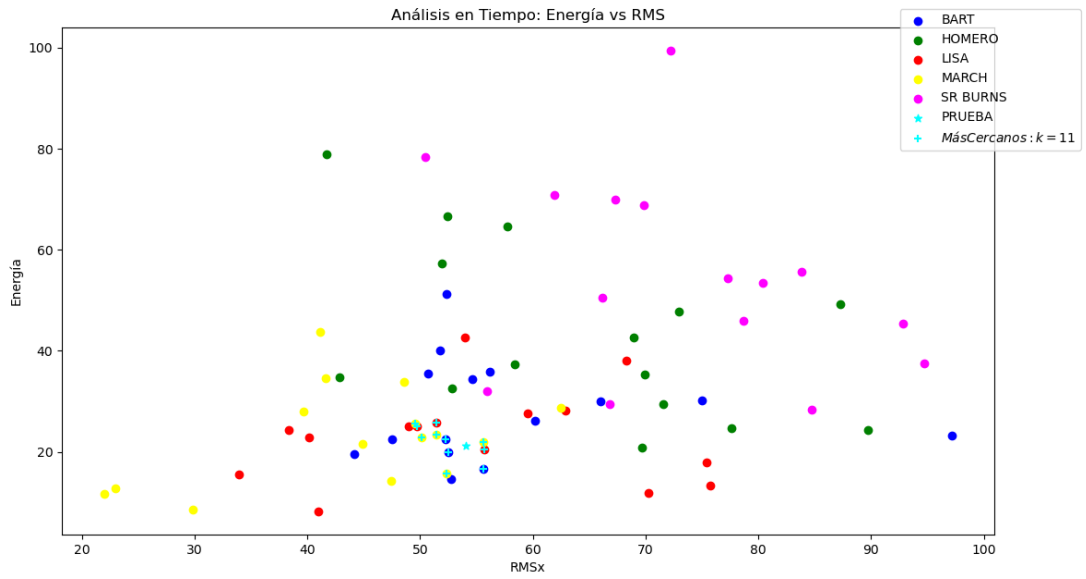


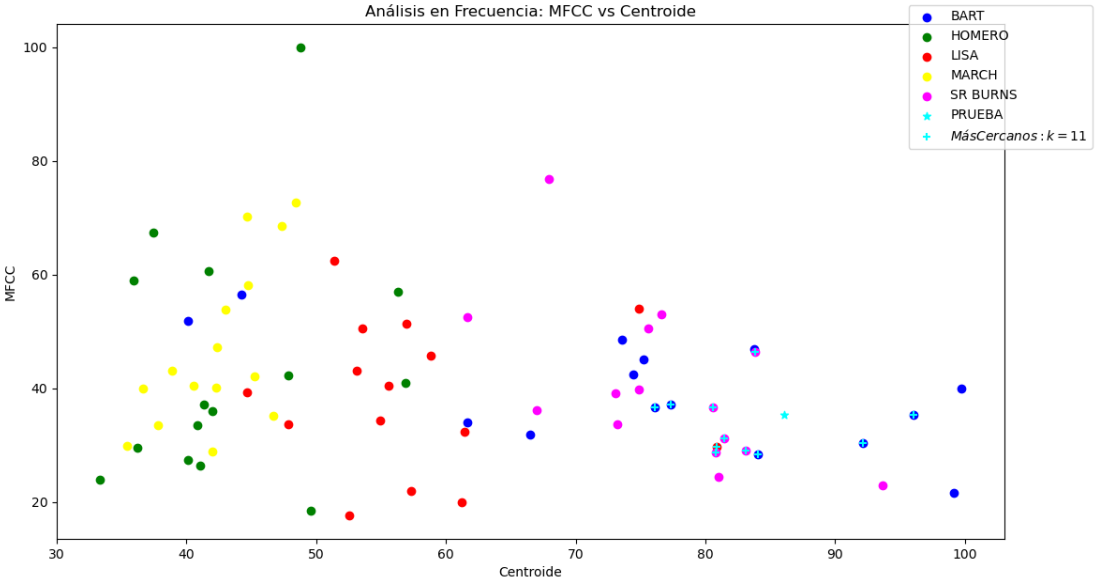
Figura 34. Gráficas de Pastel de predicciones con k = 8 como mejor opción.

Con lo cual el algoritmo nos arrojó que el audio de Prueba Uno pertenece a Lisa. Lo cual es correcto.

La siguiente prueba mostrada es la Prueba Dos (que sabemos pertenece a la voz de Bart) donde el algoritmo eligió k = 11 como mejor correlación con r =0.2762137. De **r(k[8,15])** = [0.16333965194414124, 0.15002479953724476, 0.11396057645963796, 0.2762137969161919, 0.2563137711111046, 0.192327314540518, 0.1468293986405344].



a) Dom(t): K = 11 vecinos más cercanos



b) Dom(f): k = 11 vecinos más cercanos

Figura 35. Prueba Audio Dos.

Parametrizando estos resultados en graficas de Pastel, para k = 11 como mejor tenemos que:

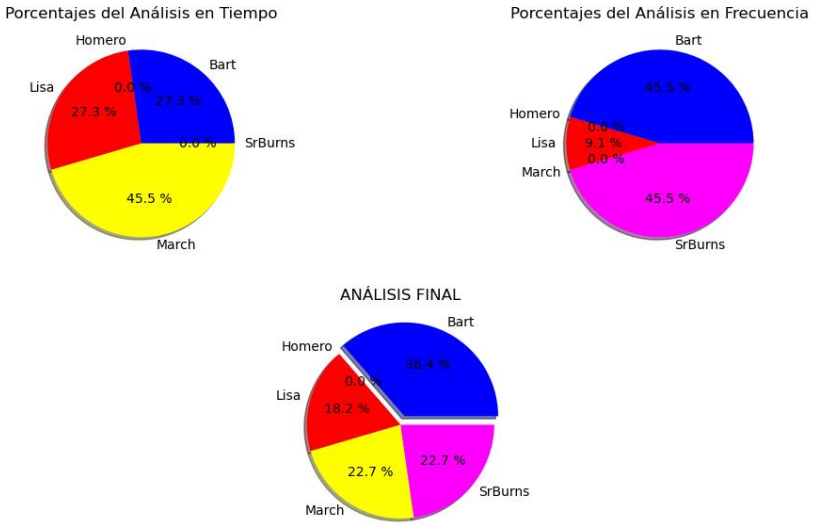


Figura 36. Gráficas de Pastel de predicciones con k = 11 como mejor opción.

Con lo cual el algoritmo nos arrojó que el audio de Prueba Dos pertenece a Bart. Lo cual es correcto.

4. Discusión

El sistema fue probado con los 15 audios de Prueba, 3 por cada personaje. Los resultados se muestran en la Tabla 1.

PERSONAJE	K = 8	K = [8,15]	
		k(Max.correlación)	
BART	correcto	11	correcto
	correcto	11	correcto
	correcto	13	correcto
HOMERO	correcto	8	correcto
	correcto	13	correcto
	correcto	9	correcto
LISA	correcto	8	correcto

	correcto	9	correcto
	correcto	13	correcto
MARCH	correcto	8	correcto
	incorrecto	11	empate
	empate	14	correcto
SR BURNS	correcto	12	correcto
	correcto	14	correcto
	correcto	14	correcto
NO DE ACIERTOS	13.5		14.5
PORCENTAJE DE EFICIENCIA	90%		96.67%

	Pesos
correcto	1
incorrecto	0
empate	0.5

Tabla 1. Resultados….

Ya que usando la correlación el sistema mejoro en los resultados, ese se usará en el artículo poniéndolo en contraste con el uso de k fijo; se usó el valor del 10% de los datos (k = 8) ya que por lo general era la mejor opción. Podemos decir por los resultados de las gráficas anteriores que K fijo al 10 % es más preciso ya que arroja siempre mayores resultados a los PORCENTAJES de los personajes, pero menos exacto ya que se equivoca un poco más. En contraste k variable en función de la correlación es más exacto pues se equivoca menos, pero con menor precisión ya que arroja MENORES PORCENTAJES a los personajes. Con lo cual podemos decir que al tener casi 97% de eficiencia con k variable en función de la correlación es mejor que k fijo ya que este tuvo un 90% de eficiencia en los resultados.

5. Conclusiones

Se logró caracterizar la voz en función de parámetros especiales del audio como sonoridad, volumen y componentes de frecuencia, así como clasificar en zonas a cada personaje en base a dichas características. También se logró el cumplir el propósito del proyecto de implementar un Sistema de Reconocimiento de Voz en un lenguaje script desde cero. Aun que el sistema no es completamente automático ya que en la Etapa de Evaluación necesita ajustes en la función de K-NN de la librería de SKLEARN, no por la lógica del algoritmo ni por problemas en la implementación, sino porque necesitamos una mayor versatilidad a la hora de manejar los parámetros de la función KNeighborsClassifier(). Por ello, al momento de implementar se tuvo la necesidad de ir variando los datos de entrada y los valores de K, para que el algoritmo tuviera una interpretación correcta de los datos. Sin embargo, se logró implementar un Sistema de Reconocimiento de Voz ya que sólo con las entradas de la caracterización realizada al audio, el Sistema fue capaz de dar resultados correctos; identificando a que personaje pertenecía el audio ingresado y desconocido por la base de datos del Sistema, tomando así una decisión con la ayuda de Aprendizaje Supervisado. Como trabajo futuro se propone programar el algoritmo K-NN y amoldarlo al procesamiento por separado de las dos graficas de caracterización, manejando así porcentajes de confiabilidad para cada gráfica, en vez de arrojar un solo resultado de etiqueta; buscaríamos tener dos y de ahí tomar una elección en función de porcentajes de confiabilidad

6. Bibliografía y Referencias

[1] About, I, and Denis, V., Historia de la identificación de las personas, 2011.

[2] Aguirrezabala, M., Estudio de verificación biométrica de voz. Tesis de Maestría, 2015.

[3] Big, Aproximación de Big Data a las Colecciones Musicales. 5to Congreso Nacional de Ingeniería, Informática/Sistemas de Información. Aplicaciones Informáticas y de Sistemas de Información. Noviembre 2017

[4] Chu, S., Narayanan, S., and Jay Kuo C., Environmental sound recognition with time-frequency audio features. IEEE Trans. Audio, Speech and Lang. pp. 1142-1158, 2012.

[5] Li-Chun, W., An Industrial-Strength Audio Search Algorithm. Shazam Entertainment, Ltd. Disponible en: https://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf

[6] Ortega, M., Introducción a la biometría. técnicas avanzadas de procesado de imagen, 2013.

[7] Tordera, J.C.: "Lingüística computacional. Tratamiento del habla". Valencia: Universtitat de València. https://es.wikipedia.org/wiki/Reconocimiento_del_habla 2011

[8] Weisstein, E.W., Fast Fourier Transform. Weisstein, Eric W, ed. MathWorld Wolfram Researc, .2015.

[9] Salamón, J., Gómez, E., Bonada, J., Sinusoid Extraction and salience function design for predominant melody stimation.Music Technology Group Universitat Pompeu Fabra, Barcelona, 2011.

[10] MIR, Music Information Retrieval: Part 2. Feature Extraction. Alexander Schindler.http://www.ifs.tuwien.ac.at/~schindler/lectures/MIR_Feature_Extraction.html

[11] Fix, E.; Hodges, J.L., An Important Contribution to Nonparametric Discriminant Analysis and Density Estimation: Commentary on Fix and Hodges (1951)». International Statistical Review / Revue Internationale de Statistique 57 (3): 233-238, 1989.