Computación Blanda - Reconocimiento de voz

Soft Computing - Speech recognition

Autor: Karen Posada Muñoz
IS&C, Universidad Tecnológica de Pereira, Pereira, Colombia
Correo-e: karen.posada@utp.edu.co

***Resumen*— Este documento muestra el problema de reconocimiento de voz a través de técnicas de aprendizaje automático**

***Palabras clave*—numpy, python, scipy**

***Abstract*—**
***Key Word*— numpy, python , scipy**

I. INTRODUCCIÓN

Existen diferentes problemas que podemos solucionar a través del aprendizaje automático uno de esas es el reconocimiento de voz de ciertos personajes y que la computadora sea capaz de identificar a quién pertenece.

II. IDENTIFICANDO EL PROBLEMA

El problema a resolver es el siguiente. tenemos unos audios correspondientes a unos personajes importantes del mundo, cada audio está debidamente etiquetado y debemos encontrar un modelo que nos diga si audio ingresado corresponde a uno de los personajes con los cuales entrenamos el modelo.

Como ya sabemos en el aprendizaje automático lo que haremos será ingresar los datos que queremos clasificar

En este caso usaremos audios de diferentes personas que debemos clasificar para que el usuario cuando ingrese un audio de prueba, el modelo pueda identificar a quien pertenece.

En el problema no se nos especifica si debemos etiquetar estos audios o no, pero en caso de necesitarlo la librería TensorFlow tiene un método para ayudarnos con las etiquetas.
Para efectos prácticos haremos de cuenta que los audios no tienen etiquetas, así que el programa deberá buscar en los datos que características resaltan en los audios, para agruparlos por familias que tengan características similares.

III. PREPARACIÓN DE LOS DATOS

Debemos saber que no le queremos enviar información errónea a nuestro programa porque nos puede dar un mal modelo y no nos permite identificar el personaje detrás del audio. Por eso es necesario hacer una etapa de preparación de los datos para poder potencializar las características más dicientes y reducir el ruido u otras características que pueden entorpecer nuestro proceso

Un aliado que podemos tener en este proceso pueden ser las diferentes transformadas de wavelets o de fourier, ya que con ellas podremos eliminar el ruido de nuestra señal.

y además podemos utilizar una serie de filtros en nuestro audio base que nos sirva a entregarle unas mejores características al modelo.

IV. MFCC

“Los MFCC (Coeficientes Cepstrales de las frecuencias de Mel – Mel Frequency Cepstral Coefficients) son coeficientes para la representación del habla basados en la percepción auditiva humana. Los MFCC muestran las características locales de la señal de voz asociadas al tracto vocal (dependiendo del instante de análisis)”

V. CONSTRUCCIÓN DEL MODELO

Como se dijo anteriormente el uso de filtros no será de gran ayuda a la hora de reconocer patrones en nuestros datos, además se eso también se recomiendo el uso del pooling para hacer una compresión de la información y así tener un proceso más eficaz.

Además de usar la convolución también tendremos en cuenta un concepto que se llama el MFCC en el cual observaremos su comportamiento que se puede extrapolar al comportamiento que tienen estas redes en imágenes.

Comenzaremos desde el uso de un modelo secuencial en la cual agregaremos convoluciones con el método CONV2D y pooling con MAXPOOLING2D.

VI. COMPILAR EL MODELO

Como en nuestro problema debemos identificar las voces de diferentes usuarios, debemos tratar de encontrar una función de pérdida que sirve para una cantidad de salidas diferente a dos, una función común para este tipo de problemas es Categorical_crossentropy.

Para el optimizador podremos usar Adam que es un algoritmo para la optimización basada en gradientes de primer orden de funciones objetivas estocásticas.