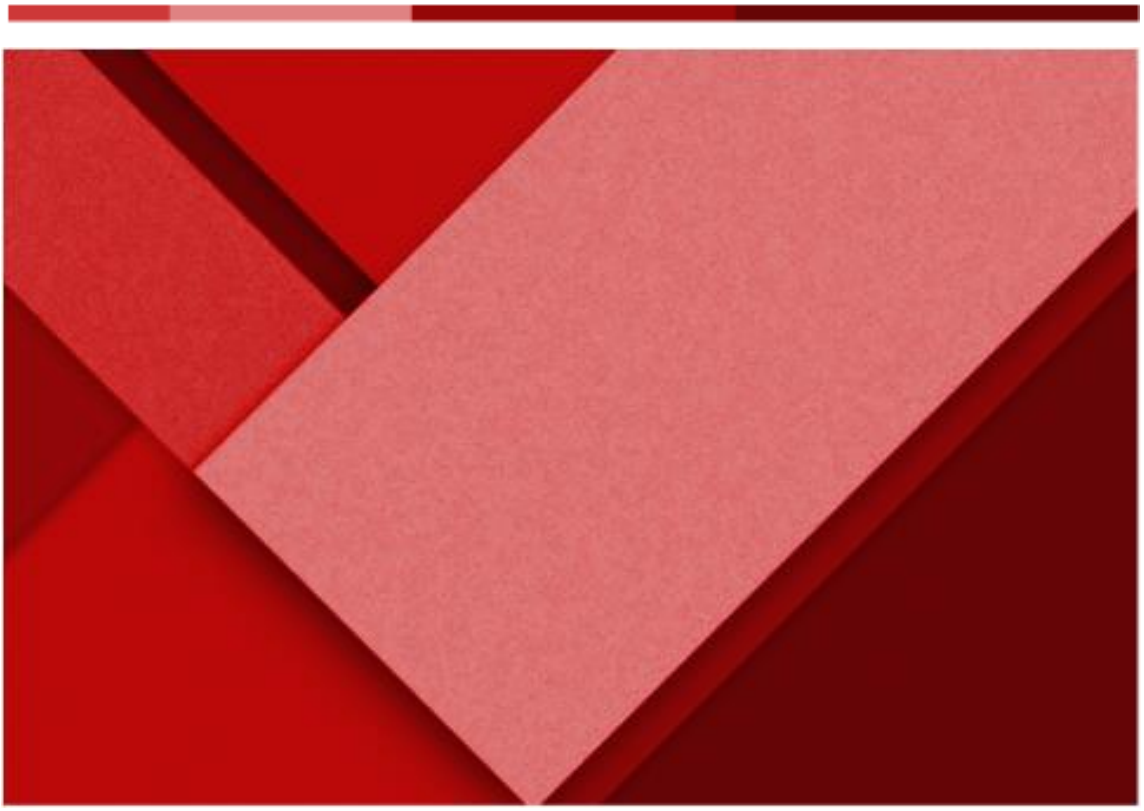

RECONOCEDOR DE EMOCIONES A PARTIR DE VOZ

INTELIGENCIA ARTIFICIAL

17/01/2020

UNIVERSAD EUROPEA DE MADRID
Campus de Villaviciosa de Odón

Índice

Índice de figuras

Introducción

La comunicación con máquina-humano ha incrementado en gran medida en estos últimos años: Alexa, Cortana, Siri, y muchos más sistemas de diálogo han golpeado el mercado de consumo en una base más amplia que nunca, pero ¿cualquiera de ellos realmente nota nuestras emociones y reacciona a ellas como lo haría otro ser humano?
El reconocimiento de voz automático ayuda a enriquecer la inteligencia artificial de próxima generación con habilidades de inteligencia emocional al captar la emoción de la voz y las palabras.

Una serie de estudios basados en la psicología más que en la informática, investigaron el papel de la acústica de la emoción humana. Blanton, por ejemplo, escribió que “el efecto de las emociones sobre la voz es reconocido por todas las personas. Incluso los más primitivos pueden reconocer los tonos de amor, miedo e ira; y este conocimiento es compartido por los animales". El lenguaje de los tonos es el más antiguo y universal de todos nuestros medios de comunicación.

Hasta ahora, el público en general ha experimentado sorprendentemente poco reconocimiento automático de la emoción en la vida cotidiana. De hecho, solo unos pocos productos comerciales relacionados han llegado al mercado, incluido el primer producto de hardware, el "Handy Truster", que apareció alrededor del cambio de milenio y afirmó ser capaz de detectar el nivel de estrés humano y engaño contenido en el discurso.

Curiosamente, un reciente estudio muestra que solo la voz como modalidad parece mejor para la empatía de los humanos precisión en comparación con solo video o comunicación audiovisual.

Descripción de la aplicación SER y breve estado del arte

Modelado

El objetivo es construir un motor capaz de reconocer la emoción del discurso.
Lo primero necesario para acercarse al reconocimiento automático de emoción requiere un apropiado modelo de representación de emociones. Esta plantea dos preguntas principales: cómo representar la emoción y cómo cuantificar de forma óptima en el tiempo el eje del tiempo.

Otros aspectos del modelado incluyen la resolución temporal, la calidad y el enmascaramiento de la emoción (actuada, provocada, naturalista, fingida, y/o regulada).

Anotación

Una vez que se decide un modelo, el siguiente tema crucial generalmente es la adquisición de datos etiquetados para entrenamiento y pruebas que se adapten al modelo de representación emocional correspondiente.

Una particularidad del campo es el alta la subjetividad e incertidumbre en el etiquetado de destino. Esto no es sorprendente, incluso los humanos generalmente no están de acuerdo en cuanto a qué emoción debe expresarse en el discurso de los demás. La autoevaluación podría ser una opción, y a menudo se usa cuando no hay información a los anotadores está disponible o es de fácil acceso, como para los datos fisiológicos. Existen herramientas adecuadas como PANAS, que permite la autoevaluación de lo positivo y negativo. Sin embargo, el afecto auto informado puede ser complicado también, ya que nadie tiene exactamente conocimiento o memoria de la emoción experimentado en un momento en el tiempo. Además, la calificación de los observadores puede ser una etiqueta más apropiada en el caso de un sistema automático de reconocimiento de emociones, ya que cabe la posibilidad de que en la evaluación de objetivos de lo expresado, prime más la emoción expresada que la emoción sentida.

Las comparaciones por parejas que conducen a una clasificación han emergido recientemente como una alternativa interesante, ya que puede ser más fácil para un evaluador comparar dos o más estímulos en lugar de tener que encontrar una asignación de valor absoluto para cualquier estímulo.

Para evitar necesidades de anotación, en trabajos pasados, a menudo se utilizan actuando (fuera de una experiencia) o provocando (dirigido) emociones. Esto tiene una desventaja porque la emoción puede no ser realista o puede ser cuestionable si se siguió el protocolo correcto de recopilación de datos.

Funciones de audio

Con datos etiquetados a mano, uno tradicionalmente necesita características de audio y texto antes de introducir datos en un algoritmo de aprendizaje automático adecuado. Esto es un subcampo activo en curso de investigación en el dominio SER: el diseño del ideal de características que mejor reflejan lo emocional debe ser robusto contra ruidos ambientales, idiomas diferentes o incluso influencias culturales.

En la síntesis de emoción, hay un fuerte enfoque en características prosódicas, es decir, describir la entonación, intensidad y ritmo del discurso junto a las características de calidad de

voz. El análisis automático del discurso emocional a menudo agrega o incluso se enfoca completamente en características espectrales o MFCCs.

La tendencia actual es aumentar la cantidad de funciones hasta algunos miles de características brutas que a menudo estaban marcadas contraste con la escasa cantidad de material de capacitación disponible en este campo.

Características textuales

Las características textuales, se fijan principalmente en palabras individuales o secuencias de estos así como su probabilidad posterior de estimar una clase o valor de emoción particular. En los enfoques de bolsas de palabras, las palabras forman un texto "feature", la frecuencia de aparición de las palabras se utiliza como valor de característica real.

Posiblemente se normaliza a el número de ocurrencias en el material de capacitación o en la cadena actual de interés, longitud de la cadena actual, o representado por logaritmo en formato binario, y así sucesivamente.

La detención o la eliminación de entidades que no ocurren suficientemente o con frecuencia parece irrelevante desde el punto de vista lingüístico o experto. Más bien, el reencuentro por clase de palabra, el etiquetado de parte del discurso puede ayudar a agregar más representaciones significativas.

Una tendencia reciente prometedora es utilizar clustering suave, es decir, no asignar una palabra observada a una sola palabra en el vocabulario.

Además, hay redes neuronales recurrentes, que han aumentado la memoria a corto plazo, esto permite otras formas de representación de contextos más largos.

Cabe señalar que el campo tradicional del análisis de sentimientos está altamente relacionado con el reconocimiento de la emoción del texto, aunque tradicionalmente es más común tratar con el escrito y, a menudo, los pasajes de texto son más largos. Este campo ofrece multitud de enfoques adicionales. Una mayor incertidumbre está dada por la incertidumbre con la que uno tiene que lidiar, incorporando medidas de confianza o hipótesis alternativas del reconocedor del habla. Además, el lenguaje hablado naturalmente difiere del texto escrito por un menor énfasis en la corrección gramatical, uso frecuente de fragmentos de palabras...etc.
En particular, las comunicaciones no verbales como risas, vaciles, consentimiento, respiración y suspiros, que ocurren con frecuencia, y deberían ser mejor reconocidos, ya que a menudo aportan información en cuanto al contenido emocional. Una vez reconocidos, pueden ser incrustados en una cadena.

La información de características acústicas y lingüísticas se puede fusionar directamente mediante concatenación en un solo vector de características si ambos operan en el mismo nivel de tiempo.

Modelos holísticos de voces: Se deberá tener en consideración otros aspectos que impactan en la producción de la voz; ya que, la voz no solo se categoriza por la emoción sino que también la persona podría estar cansado o tener un resfriado. Así que los motores de reconocimiento de voz deberían ver la imagen más amplia de los estados y rasgos de un hablante más allá de la emoción de interés para reconocerla mejor independientemente de influencia de factores como estos.

Estudios relacionados con SER usando Redes Neuronales Profundas

- ***"Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine".***

  Metodología: Características MFCC, características basadas en tono: período de tono y relación armónicos a ruido (HNR), su función delta a través de marcos de tiempo. Base de datos interactiva Emocional Dyadic Motion Capture (IEMOCAP): emoción, frustración, felicidad, neutralidad y sorpresa.

  A favor: Metodología bien descrita sobre cómo realizar experimentos. Los resultados muestran que la metodología DNN propuesta supera a HMM (Hidden Markov Model) y SVM (support vector machine) en un 20% de precisión relativa. El paradigma de ELM (Extreme Machine Learning) propuesto es 10 veces más rápido que el de SVM.

  En contra: Si bien se intenta el análisis de comparación entre DNN y HMM y SVM, se proporciona menos información sobre cómo se realiza este último (HMM y SVM). Si bien el promedio ponderado y no ponderado es más preciso, no se menciona la tasa de reconocimiento general.

- ***"Acoustic Emotion Recognition using Deep Neural Network".***

  Metodología: MFCC, predictivo lineal perceptivo (PLP) y bancos de filtros (FBANK) 9595 frases de emoción: Enojado, feliz, miedo, triste, sorpresa, neutral.

  A favor: Mejor documentación de análisis de comparación entre GMM (Gaussian Mixture Model) y DNN, en igualdad de condiciones. La precisión de DNN muestra un aumento de 8.2 puntos porcentuales en comparación con las líneas base GMM, hasta 92.3%.

  En contra: No se menciona el tiempo de procesamiento. No se menciona la base de datos en aras de verificar resultados o comparaciones.

- ***"Deep Learning Based Affective Model for Speech Emotion Recognition".***

  Metodología: La extracción de características es administrada automáticamente por redes profundas. Base de datos alemana del discurso emocional de Berlín: ira, aburrimiento, asco, ansiedad, felicidad, tristeza y estado neutral.

  A favor: Propone un sistema afectivo que elegirá las características por sí mismo. La precisión del reconocimiento alcanza el 65% en el mejor de los casos, una mejora de su punto de referencia, que es del 22%.

  En contra: El uso de un sistema que elige las características apropiadas es prometedor, pero debe mencionarse qué características distintas son prominentes. La falta de profundidad implica dependencia en la caja de herramientas.

- ***"Speech Emotion Recognition from Spectrograms with Deep Convolutional Neural Network".***

  Metodología: Características extraídas del espectrograma generado a partir del habla. Base de datos alemana de discurso emocional de Berlín: ira, aburrimiento, asco, ansiedad, felicidad, tristeza y estado neutral.

  A favor: Utiliza un enfoque novedoso de usar el reconocimiento de imágenes del espectrograma generado a partir del habla. Logró una tasa de reconocimiento del 84.3%.

  En contra: La necesidad de analizar el espectrograma del audio agrega una capa de complejidad al sistema SER, que puede no aplicarse en la vida real.

- ***"Speech emotion recognition based on Gaussian Mixture Models and Deep Neural Networks".***

  Metodología: Energía, tono, probabilidad de voz y registro de 26 dimensiones de características del espectrograma Mel, total 58 características de cada cuadro. Total 10; 527 enunciados utilizables en mandarín de tráfico real desde un sistema de diálogo hablado de Microsoft. Neutral, feliz, triste, enojado.

  A favor: Proporciona documentación confiable y completa de recopilación de datos. Se aplican 4 algoritmos diferentes: GMM, DNN y 2 variaciones de aprendizaje automático extremo (ELM).

  En contra: Con la gran cantidad de expresión emocional, debería ser posible una mayor variación de la clasificación de las emociones. Si bien deja más espacio para futuras investigaciones para mejorar, la mejor tasa de reconocimiento es solo del 57.9% con ELM-DNN.

Proceso de preprocesamiento

El preprocesamiento de datos permite convertir los audios iniciales en datos que contengan un formato más fácil y adecuado para utilizarlos con los modelos generados.

Características extraídas en un audio

A continuación se van a explicar las características extraídas de cada uno de los audios:

MFCCs (Coeficientes Cepstrales en las Frecuencias de Mel)

Son coeficientes para la representación del habla basados en la percepción auditiva humana.

La señal del audio completo cambia constantemente en el tiempo, lo cual dificulta la extracción de características que la puedan diferenciar o identificar en otras señales. Por esto y con el objetivo de simplificar el tratamiento de la señal se deben asumir pequeños periodos de tiempo donde sus características no cambien "mucho" y por tanto se puede realizar a la señal todo un conjunto de procesamientos con el objetivo de extraer características "estáticas".

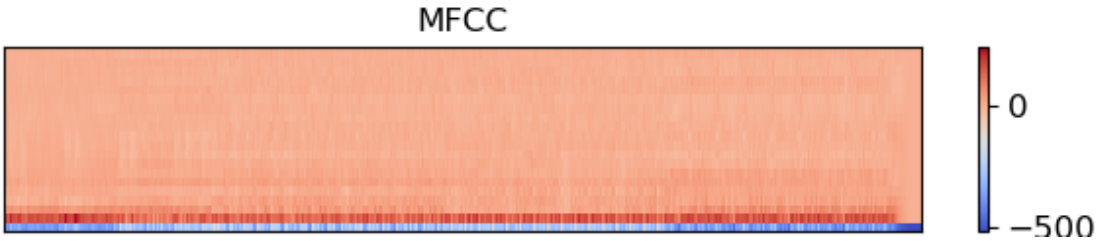

***Figura 1**- Coeficientes Cepstrales en las Frecuencias de Mel*

El número de periodos de tiempo para analizar la señal de los audios será igual a cien en nuestro caso. La duración estimada de estos está en torno a los dos segundos.
Con el siguiente código de Python extraemos la MFCCs de cada uno de los audios:

```
def extract_feature(file_name):
    with soundfile.SoundFile(file_name) as sound_file:
        X, sample_rate = librosa.load(file_name, mono=True)
        mfccs_mean = np.mean(librosa.feature.mfcc(y=X, sr=sample_rate, n_mfcc=100).T, axis=0)
    return mfccs_mean
```

***Figura 2** - Código de Python para extraer las MFCCs de los audios*

Velocidad

Para cada uno de los MFCCs anteriores, guardamos la velocidad del habla de los estos. Para esto calculamos la primera derivada del MFCC de entrada.

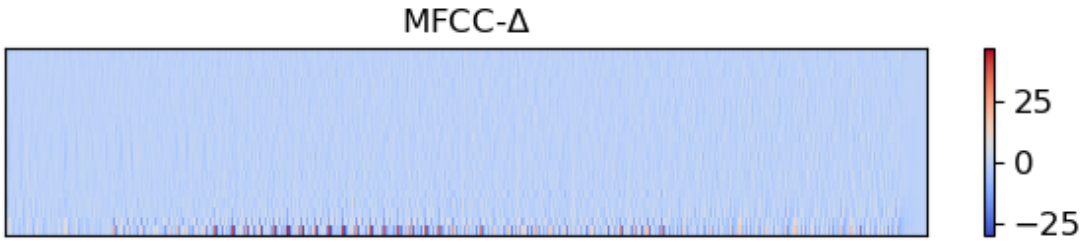

***Figura 3** - Velocidad de una MFCCs*

Por lo tanto si guardábamos 100 MFCCs, tendremos un total de 100 velocidades extraídas a lo largo del audio.
Con el siguiente código de Python extraemos la MFCCs de cada uno de los audios:

```
librosa.feature.delta(mfccs)
```

Figura 4 - Código de Python para extraer la velocidad de un MFCCs

Aceleración

Al igual que en la velocidad, para cada uno de los MFCCs, guardamos la aceleración del habla. Para esto calculamos la segunda derivada del MFCC de entrada.

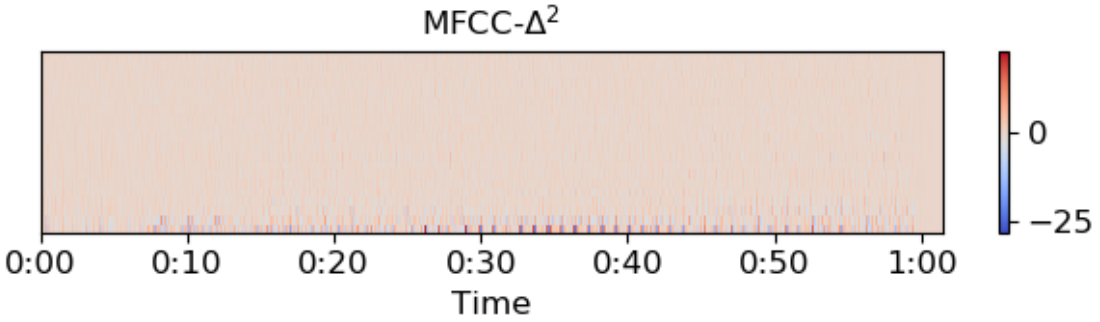

Figura 5 - Aceleración de una MFCCs

Por lo tanto si guardábamos 100 MFCCs, tendremos un total de 100 aceleraciones extraídas a lo largo del audio.

Con el siguiente código de Python extraemos la MFCCs de cada uno de los audios:

```
librosa.feature.delta(mfccs, order=2)
```

Figura 6 - Código de Python para extraer la aceleración de un MFCCs

Resultado obtenido

Tras extraer las características anteriores de cada uno de los audios, las extraemos en un fichero .csv. Este fichero estará formado por un total de 301 columnas. Estas columnas se podrían agrupar de la siguiente forma.

- **Primera Columna (0):** Esta indicará uno de los siete posibles estados de ánimo analizados (Alegría, Asco, Ira, Miedo, Neutro, Sorpresa y Tristeza).
- **Segunda columna a la columna cien (1-100**): En cada una de estas columnas estará el resultado de uno de los 100 MFCCs analizados.
- **Columna ciento uno a la doscientos (101-200):** En cada una de estas columnas estará el resultado de cada una de las velocidades extraídas de los 100 MFCCs.
- **Columna doscientos uno a la trescientos (201-300):** En cada una de estas columnas estará el resultado de cada una de las aceleraciones extraídas de los 100 MFCCs.

Análisis – Clasificación

A lo largo de este punto vamos a explicar cómo hemos llevado a cabo el entrenamiento con el clasificador de Bayes y con las redes neuronales artificiales. Además, explicaremos y compararemos los resultados de la validación y del testeo de los datos.

Entrenamiento – Validación

En este punto vamos a explicar cómo hemos realizado del entrenamiento para cada uno de los casos y que resultados hemos obtenido al hacer la validación de este.

Clasificador de Bayes

Para llevar a cabo el entrenamiento primero hemos importado utilizando la librería de pandas, los datos preprocesadors anteriormente que se habían guardado en un fichero .csv. Después hemos transformado las posibles emociones a datos numéricos para la predicción y hemos separado un 10% de los datos para llevar a cabo la validación. Con esto ya podemos entrenar a nuestro clasificador de Bayes.

```
clf = GaussianNB()
clf.fit(X_train, y_train)
```

***Figura 7** - Código de Python entrenamiento del clasificador de Bayes*

Después de entrenar el modelo llevamos a cabo la validación, obteniendo el siguiente resultado:

Validación - Naive Bayes

	precision	recall	f1-score	support
0	0.22	0.25	0.24	8
1	0.31	0.39	0.35	23
3	0.30	0.55	0.39	11
4	0.54	0.32	0.40	22
5	0.24	0.29	0.26	17
7	0.33	0.31	0.32	16
8	0.78	0.37	0.50	19
accuracy			0.35	116
macro avg	0.39	0.35	0.35	116
weighted avg	0.42	0.35	0.36	116

***Figura 8** - Validación clasificador de Bayes*

Hay que tener en cuenta que el 10% de los datos de validación se escogen aleatoriamente, por lo que el resultado obtenido puede variar en torno un 5%.

Redes Neuronales Artificiales

Al igual que en el punto anterior hemos importado los datos preprocesados anteriormente y hemos separado un 10% de los datos para hacer la validación.
En este punto el entrenamiento es más complejo. Para crear la red neuronal que mejores resultados ofrecía hemos utilizado GridSearchCV de Sklearn.
Los parámetros que hemos utilizado para nuestra red neuronal son los siguientes:

1. **Solver (Algoritmo de optimización):** Hemos probado con tres algoritmos de optimización:
    - **Adam:** es un algoritmo de optimización que se puede utilizar para actualizar los pesos de red de forma iterativa en función de los datos de entrenamiento.

- **SGD:** Actualiza los pesos mediante una combinación lineal del gradiente negativo y la actualización de peso anterior. La tasa de aprendizaje es el peso del gradiente negativo. El impulso es el peso de la actualización anterior.

De estos dos el que mejor resultados ofrecían con diferencia era el algoritmo de optimización Adam.

2. **Max_iter (Iteraciones):** Indica el número de iteraciones que hace la red neuronal al dataset.
   En este caso probamos con 500, 1000, 1500, 2000 y 2500 iteraciones.
   Él número de iteraciones que mejor resultado ofreció fueron 1500.
3. **Hidden_layer_sizes (número de neuronas por cada capa oculta):** En este punto indicabas el número de neuronas que quisieras que hubiese por cada capa oculta.
   En nuestro caso cuando teníamos una capa oculta con un gran número de neuronas ofrecía mejor resultado que añadiendo nuevas capas ocultas.
   El mejor resultado lo obtenemos con una capa con unas 500 neuronas aproximadamente.

Tras establecer los parámetros entrenamos a la red neuronal con el siguiente código de Python:

```
clf = GridSearchCV(MLPClassifier(), parameters, n_jobs=-1)
clf.fit(X,y)
```

***Figura 9** - Código de Python entrenamiento de la red neuronal*

Tras entrenarse la red neuronal con las diferentes combinaciones posibles de sus parámetros llevamos a cabo la validación, obteniendo el siguiente resultado:

Validación - Red Neuronal

	precision	recall	f1-score	support
0	0.67	0.57	0.62	14
1	0.70	0.70	0.70	20
3	0.75	0.75	0.75	12
4	0.93	0.76	0.84	17
5	0.56	1.00	0.72	9
7	0.79	0.71	0.75	21
8	0.87	0.87	0.87	23
accuracy			0.76	116
macro avg	0.75	0.77	0.75	116
weighted avg	0.77	0.76	0.76	116

***Figura 10** - Validación redes neuronales*

Cómo podemos ver los resultados con la red neuronal con los siguientes parámetros casi duplican los resultados obtenidos con el clasificador de Bayes.
Tras esto, podemos determinar que la precisión y el aprendizaje de la red neuronal en la validación es mucho mejor que el obtenido con el clasificador de Bayes. Es decir, la red neuronal es capaz de diferenciar mejor él estado de ánimo en los audios de las personas con las que ya se entrena.

Test

Durante el test, tanto el clasificador de Bayes cómo la red neuronal artificial, tratarán de diferenciar el estado de ánimo en los audios de personas con las que no se han entrenado, es decir, personas completamente nuevas.

Clasificador de Bayes

Resultado Test 01

Test 01 - Naive Bayes

	precision	recall	f1-score	support
0	0.31	0.33	0.32	12
1	0.30	0.92	0.45	12
3	0.67	0.33	0.44	12
4	0.00	0.00	0.00	12
5	0.00	0.00	0.00	12
7	0.40	0.50	0.44	12
8	0.14	0.08	0.11	12
accuracy			0.31	84
macro avg	0.26	0.31	0.25	84
weighted avg	0.26	0.31	0.25	84

Figura 11 *- Clasificador de Bayes: Test 01*

Resultado Test 02

Test 02 - Naive Bayes

	precision	recall	f1-score	support
0	1.00	0.08	0.15	12
1	1.00	0.08	0.15	12
3	0.60	1.00	0.75	12
4	0.29	0.82	0.43	11
5	0.50	0.08	0.14	12
7	1.00	0.08	0.15	12
8	0.41	0.92	0.56	12
accuracy			0.43	83
macro avg	0.69	0.44	0.34	83
weighted avg	0.69	0.43	0.33	83

Figura 12*- Clasificador de Bayes: Test 02*

El clasificador de Bayes obtiene unos resultados similares durante el test y la validación. Este diferencia un 12% mejor los estados de ánimos del segundo test que los del primero.

Redes Neuronales Artificiales

Resultado Test 01

Test 01 - Red Neuronal	precision	recall	f1-score	support
0	0.40	0.33	0.36	12
1	0.40	1.00	0.57	12
3	0.50	0.92	0.65	12
4	0.00	0.00	0.00	12
5	0.12	0.08	0.10	12
7	0.67	0.33	0.44	12
8	0.25	0.17	0.20	12
accuracy			0.40	84
macro avg	0.33	0.40	0.33	84
weighted avg	0.33	0.40	0.33	84

Figura 13 *– Red Neuronal: Test 01*

Resultado Test 02

Test 02 - Red Neuronal	precision	recall	f1-score	support
0	0.50	0.83	0.62	12
1	0.00	0.00	0.00	12
3	0.56	0.42	0.48	12
4	0.54	0.64	0.58	11
5	0.00	0.00	0.00	12
7	0.47	0.58	0.52	12
8	0.40	0.83	0.54	12
accuracy			0.47	83
macro avg	0.35	0.47	0.39	83
weighted avg	0.35	0.47	0.39	83

Figura 14 *– Red Neuronal: Test 02*

Con las redes neuronales observamos que la precisión disminuye hasta un 36%.
Esto se debe a que los datos de los audios de las nuevas personas (tono, rapidez del habla, etc.) no tienen por qué ser similares a los datos de los audios de las personas con las que la red neuronal había entrenado anteriormente.

En este caso, la red neuronal vuelve a dar mejores resultados que el clasificador de Bayes.

Conclusión

Durante esta práctica hemos tratado de diferenciar el estado de ánimo de varias personas según unos determinados audios grabados por estas.

Para tener un formato más fácil y adecuado para utilizar los audios hemos llevado a cabo un preprocesamiento de estos, donde hemos extraído las siguientes características: MFCCs, velocidad y aceleración.

Para diferenciar los estados de ánimos hemos utilizado el clasificador de Bayes y redes neuronales artificiales.

Durante la validación del entrenamiento hemos obtenidos resultados en torno al 75% con redes neuronales y un 35% con el clasificador de Bayes, por lo que podemos determinar que las redes neuronales son capaces de diferenciar mejor el estado de ánimo de personas con las que ha entrenado con anterioridad.

Durante el entrenamiento, esta diferencia entre la precisión de ambos algoritmos disminuye notablemente.
Si comparamos los resultados del primer test, el clasificador de Bayes obtiene una precisión un 10% menor que la de las redes neuronales, por lo que podemos determinar que en este caso la red neuronal es capaz de diferenciar mejor el estado de ánimo según los audios de esta nueva persona.
En el segundo test, la red neuronal vuelve a ser superior logrando alcanzar casi el 50% de precisión.

En conclusión, tanto en la fase de validación como en la fase del test, la red neuronal obtiene una mejor precisión que Naive Bayes.

Bibliografía

BAIR. (s.f.). *Solver*. Obtenido de https://caffe.berkeleyvision.org/tutorial/solver.html

Schuller, B. W. (2018). Speech Emotion Recognition. Communications of the ACM, 61(5), 92-99.