Universidad Industrial de Santander

Patrimonio
educativo y cultural

CLASIFICACIÓN DE VOZ A PARTIR DE GRABACIONES CON SONIDO AMBIENTAL

Víctor Alfonso
Mantilla Villamizar

Código: 2151846

Contenido

Objetivo

Crear un clasificador que, a partir de una grabación de audio en condiciones ruidosas, determinar en qué partes de la grabación una o más personas están hablando.

Ejemplo

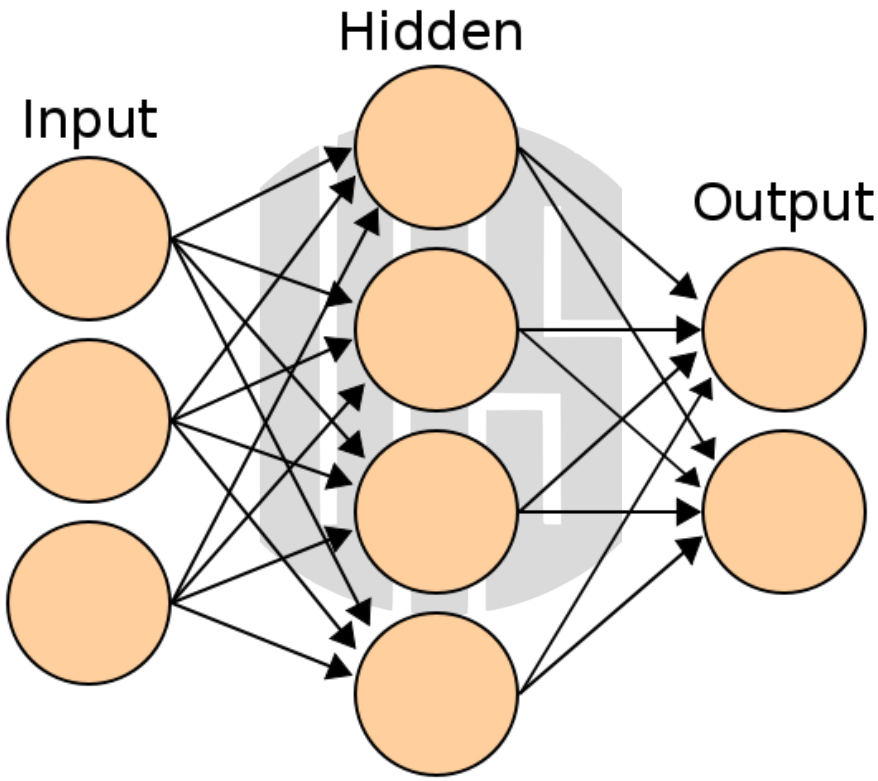
Motivación

Ayudar a crear archivos de subtítulos automáticamente para personas con discapacidades auditivas, personas que deseen consumir productos audiovisuales en otro idioma, y la creación automática de datasets para otras investigaciones.

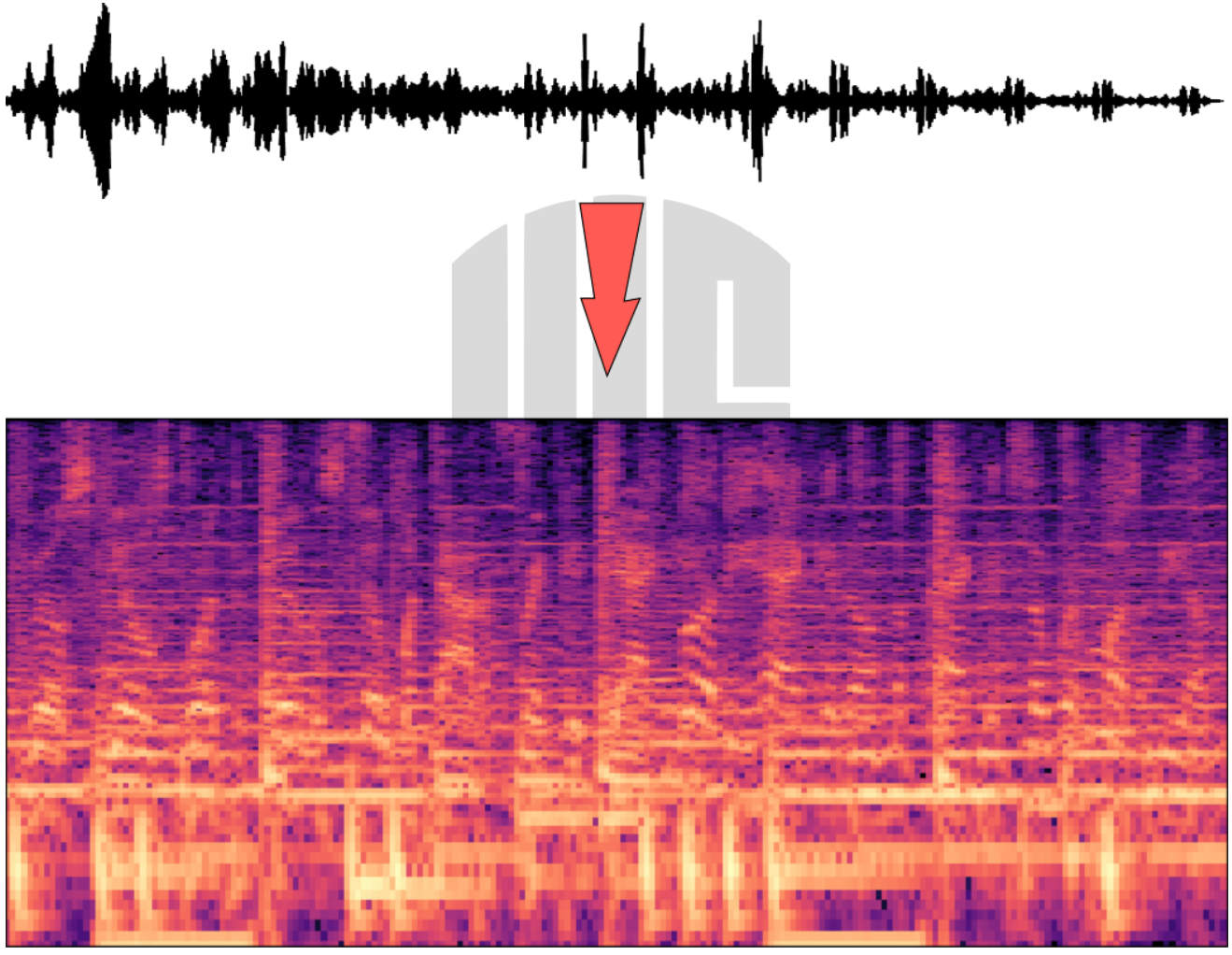
Desafios

- Plantear un mejor conjunto de características
- Buscar el mejor clasificador
- Probar el modelo escogido

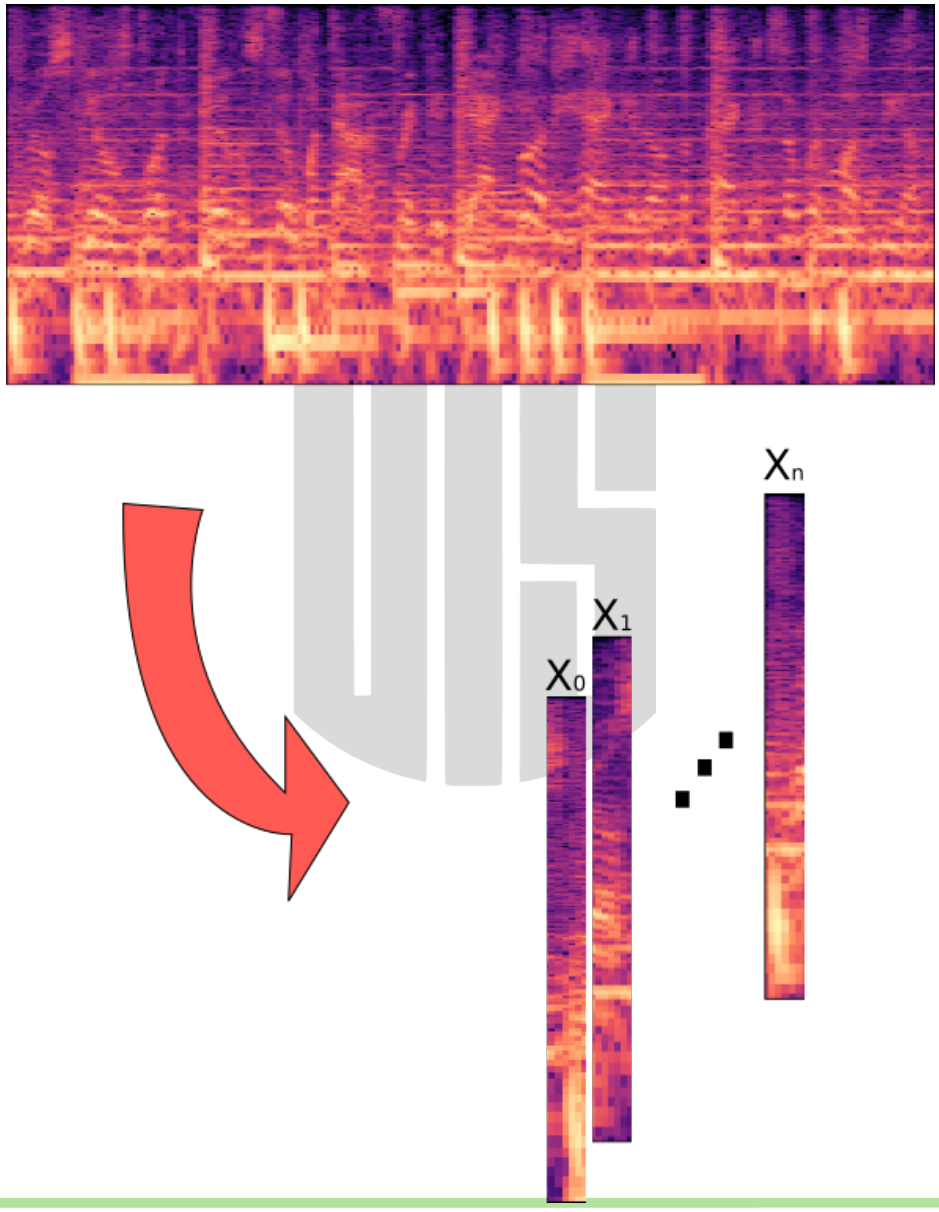
Funcionamiento



El dataset



Las características



El modelo

```
model = Sequential()
model.add(Conv2D(16, (3,3), padding='same', input_shape=(513, 25, 1)))
model.add(LeakyReLU())
model.add(Conv2D(16, (3,3), padding='same'))
model.add(LeakyReLU())
model.add(MaxPooling2D(pool_size=(3,3)))
model.add(Dropout(0.25))
model.add(Conv2D(16, (3,3), padding='same'))
model.add(LeakyReLU())
model.add(Conv2D(16, (3,3), padding='same'))
model.add(LeakyReLU())
model.add(MaxPooling2D(pool_size=(3,3)))
model.add(Dropout(0.25))
model.add(Flatten())
model.add(Dense(64))
model.add(LeakyReLU())
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))
sgd = SGD(lr=0.001, decay=1e-6, momentum=0.9, nesterov=True)
model.compile(loss=keras.losses.binary_crossentropy, optimizer=sgd, metrics=['accuracy'])
```

Capas del modelo usado

Pruebas

```
Train on 12056 samples, validate on 1000 samples
Epoch 1/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6218 - acc: 0.6600 - val_loss: 0.6117 - val_acc: 0.6790
Epoch 2/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6171 - acc: 0.6665 - val_loss: 0.6153 - val_acc: 0.6550
Epoch 3/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6129 - acc: 0.6685 - val_loss: 0.6236 - val_acc: 0.6390
Epoch 4/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6132 - acc: 0.6723 - val_loss: 0.6313 - val_acc: 0.6510
Epoch 5/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6096 - acc: 0.6762 - val_loss: 0.6188 - val_acc: 0.6620
Epoch 6/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6092 - acc: 0.6741 - val_loss: 0.6097 - val_acc: 0.6800
Epoch 7/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6101 - acc: 0.6756 - val_loss: 0.6058 - val_acc: 0.6830
Epoch 8/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6076 - acc: 0.6780 - val_loss: 0.6050 - val_acc: 0.6870
Epoch 9/10
12056/12056 [==============================] - 128s 11ms/step - loss: 0.6081 - acc: 0.6757 - val_loss: 0.6110 - val_acc: 0.6730
Epoch 10/10
12056/12056 [==============================] - 126s 10ms/step - loss: 0.6034 - acc: 0.6806 - val_loss: 0.6010 - val_acc: 0.6820
```

Entrenamiento del modelo

Test accuracy: (same source data) 0.672

Test accuracy: (second source data) 0.488

Conclusiones

- El modelo tiene potencial de mejoramiento (por ejemplo, con más Epochs)
- El modelo parece restringirse a que los datos provengan de la misma fuente.
- El análisis de imágenes puede ser aplicado en el análisis de sonidos, como una herramienta de apoyo

GRACIAS