BATS: Bridging Acoustic Transparency in Speech

Diego Quezada
Departamento de Informática
Universidad Técnica Federico Santa María
Valparaíso, Chile
diego.quezadac@sansano.usm.cl

Felipe Cisternas
Departamento de Informática
Universidad Técnica Federico Santa María
Valparaíso, Chile
felipe.cisternasal@sansano.usm.cl

Resumen—El reconocimiento de voz se basa en representaciones de señales acústicas, como espectrogramas y MFCCs. Sin embargo, los modelos actuales son en gran medida opacos en cuanto a cómo toman decisiones en este proceso. La naturaleza física de los datos de entrada en el reconocimiento de voz agrega una capa adicional de complejidad, lo que plantea el desafío de mejorar la transparencia y la comprensión de estos modelos para garantizar un reconocimiento de voz más preciso y confiable.

Index Terms—ASR, XAI, CNN, RNN, Transformers

I. Introducción

En el ámbito del reconocimiento de voz, la selección de un modelo y su arquitectura es fundamental para enfrentar los retos específicos que esta disciplina impone. Modelos de vanguardia como Whisper de OpenAI, que se basan en la arquitectura de Transformers, son altamente eficaces, pero actúan como cajas negras, lo que significa que los procesos de decisión que siguen para convertir señales de audio en palabras no son transparentes. Esta falta de explicabilidad, esencial para generar confianza en los usuarios, hace imperativo el uso de técnicas de eXplainable Artificial Intelligence (XAI) para desentrañar y comprender cómo estos modelos avanzados toman sus decisiones.

La explicabilidad en modelos de reconocimiento de voz no solo mejora la comprensión del proceso de toma de decisiones, sino que también es clave para aumentar su robustez, especialmente cuando dichos modelos se integran como componentes centrales en un software. Esta característica se vuelve esencial para ganar la confianza de los usuarios en el producto final. Sin embargo, el desafío se intensifica debido a la naturaleza física y compleja de los datos de entrada, lo que dificulta proporcionar explicaciones claras y de alto nivel basados en estos datos. Por ello, es crucial mantener un equilibrio entre avanzar en la precisión y eficiencia de los modelos y desarrollar soluciones que mejoren su explicabilidad, asegurando así un balance óptimo entre rendimiento y comprensión del modelo.

A pesar de la importancia de la explicabilidad en el reconocimiento de voz, librerías de explicabilidad ampliamente conocidas, como LIME [1], SHAP [2] y Captum [3], no ofrecen métodos de explicabilidad para estos modelos. Por ende, es necesario ajustar los métodos existentes para que sean aplicables a esta tarea.

Esta investigación tiene como objetivo mejorar la comprensión de los modelos de reconocimiento de voz para que los usuarios, especialmente aquellos con discapacidades auditivas, puedan confiar en su funcionamiento.

I-A. Trabajo relacionado

Las investigaciones sobre explicabilidad en modelos de reconocimiento de voz son escasas, pero destacan por su aplicación innovadora de métodos de explicabilidad.

En [4] se aborda la tarea reconocimiento de voz utilizando el conjunto de datos CommonVoice [5] y tres sistemas de reconocimiento de voz: Google API, Sphinx y DeepSpeech. Los autores plantean las explicaciones como un subconjunto de fotogramas de audio que son causas mínimas y suficientes de la transcripción. Para esto, adaptan las técnicas de clasificación de imagen SFL [6] y explicaciones composicionales [7] para luego compararlas con las ofrecidas por LIME [1].

En [8] se aborda la tarea de reconocimiento de fonemas utilizando el conjunto de datos TIMIT [9] y el método LIME junto a dos variantes propuestas: LIME-WS y LIME-TS. El conjunto de datos TIMIT segmenta los distintos fonemas en los datos de entrada. Los autores utilizan como representación interpretable un vector que indica la presencia o ausencia de cada segmento de audio. El objetivo es identificar los segmentos más importantes para la predicción del siguiente fonema. Inicialmente, para la generación del vecindario se aplican máscaras aleatorias a los segmentos de TIMIT. Luego, considerando que al analizar un fonema es probable que solo los segmentos cercanos incidan en su identificación, LIME-WS solo aplica máscaras aleatorias a segmentos cercanos al fonema analizado y mantiene los lejanos constantes. Finalmente, en LIME-TS se consideran segmentos creados por un intervalo de tiempo en vez de los definidos por TIMIT y se aplica la misma idea de localidad que en LIME-WS.

En [10] se aborda la tarea de etiquetado de música. Los autores plantean la necesidad de dar explicaciones audibles

y proponen audioLIME como una extensión de LIME que utiliza explicaciones basadas en separación de fuentes.

II. Método propuesto

II-A. Marco teórico

II-A1. Speech Recognition: El reconocimiento de voz, o más conocido en inglés como speech recognition, es la tarea de asignar una secuencia de palabras a señales acústicas que contienen lenguaje hablado. Implica reconocer las palabras pronunciadas en una grabación de audio y transcribirlas a un formato escrito. El objetivo es transcribir con precisión el discurso en tiempo real o a partir de audio grabado, teniendo en cuenta factores como el acento, la velocidad del habla y el ruido de fondo. Cuando la transcripción se realiza en tiempo real se habla de reconocimiento automático de voz o Automatic Speech Recognition (ASR) en inglés. Considerando $\mathbf{X} = (x^{(1)}, x^{(2)}, \ldots, x^{(T)})$ como una secuencia de audio de largo T e $y = (y_1, y_2, \ldots, y_N)$ como una secuencia de palabras de largo N podemos definir la tarea de reconocimiento de voz de manera precisa mediante el siguiente problema de optimización:

$$f^*(\mathbf{X}) = \arg\max_{\mathbf{y}} P^*(\mathbf{y}|\mathbf{X} = X) \tag{1}$$

Donde P^* es la verdadera distribución de probabilidad condicional que relaciona las entradas $\mathbf{X}$ con las salidas $\mathbf{y}$ [11].

La representación utilizada para $\mathbf{X}$ es de vital importancia para el desempeño de un modelo de reconocimiento de voz. La representación más simple es mediante una serie temporal univariada que modela la amplitud de la señal de audio en el tiempo. Al dividir la señal de audio en pequeñas ventanas de tiempo y calculando el espectro de frecuencia para cada ventana se obtiene un espectrograma: una representación visual de la señal de audio en el tiempo y en el dominio de la frecuencia. A partir del espectrograma se pueden extraer características de audio tales como los coeficientes cepstrales de Mel (MFCCs) que son ampliamente utilizados en la literatura para el reconocimiento de voz.

II-A2. Métricas de Evaluación: Para poder evaluar un modelo de ASR se utilizan métricas como:

- Word Error Rate (WER): La métrica Word Error Rate se calcula como el número total de errores, que es la suma de sustituciones, inserciones y eliminaciones de palabras necesarias para cambiar una secuencia de palabras hipotética a una secuencia de referencia, dividido por el número total de palabras en la secuencia de referencia. WER proporciona una medida de cuántas palabras fueron reconocidas incorrectamente en proporción al total de palabras habladas. La fórmula para calcular el Word Error Rate (WER) es:

$$\text{WER} = \frac{S + D + I}{N} \tag{2}$$

Donde:
- S es el número de sustituciones,
- D es el número de eliminaciones,
- I es el número de inserciones,
- N es el número de palabras en la secuencia de referencia.

WER se expresa a menudo como un porcentaje, y cuanto más bajo es el WER, mejor es el rendimiento del sistema de reconocimiento de voz.
- Match Error Rate (MER): La métrica Match Error Rate es similar a WER, pero en lugar de centrarse solo en las palabras, evalúa la precisión en el emparejamiento de cualquier tipo de elementos, como fonemas, letras o incluso palabras en tareas de reconocimiento. La fórmula para calcular MER es:

$$\text{MER} = \frac{S + D + I}{S + D + C} \tag{3}$$

Donde:
- S es el número de sustituciones,
- D es el número de eliminaciones,
- I es el número de inserciones,
- C es el número de aciertos correctos.

MER también se suele expresar en porcentaje y proporciona una medida de cuántos elementos fueron incorrectamente emparejados en relación con el total de elementos que debían emparejarse, cuanto más bajo es MER mejor es el rendimiento del modelo.
- Word Information Lost (WIL): La métrica Word Information Lost es una medida teórica de la información basada en la entropía desde la perspectiva de la información de las palabras que se pierden en la transcripción de la hipótesis comparada con la referencia. La tasa de WIL se puede calcular con la fórmula:

$$\text{WIL} = 1 - \frac{C}{N} + \frac{C}{P} \tag{4}$$

donde:
- C es el número de palabras correctas,
- N es el número de palabras en la referencia,
- P es el número de palabras en la predicción.

La métrica WIL está diseñada para estar acotada entre 0 y 1, proporcionando una forma normalizada de medir la información perdida en la transcripción. Un valor más bajo de WIL indica un mejor rendimiento del sistema de ASR, siendo 0 el puntaje perfecto. Es importante mencionar que, aunque WIL comparte similitudes con WER en cuanto a su uso para evaluar transcripciones de ASR, está diseñada para abordar ciertas limitaciones de WER al proporcionar una medida normalizada que está efectivamente limitada.

- Word Information Preserved (WIP): La métrica Word Information Preserved mide el porcentaje de palabras correctas predichas entre un conjunto de oraciones verdaderas y un conjunto de oraciones hipotéticas. La fórmula para calcular la tasa de WIP es:

$$wip = \frac{C}{N} + \frac{C}{P} \tag{5}$$

donde:
- C es el número de palabras correctas,
- N es el número de palabras en la referencia,
- P es el número de palabras en la predicción.
Un valor más alto de WIP indica un mejor rendimiento del sistema ASR, siendo 1 la puntuación perfecta.
- Character Error Rate (CER): La métrica Character Error Rate se calcula de manera similar a WER, pero en lugar de considerar las palabras, se enfoca en los caracteres individuales. La fórmula para calcular CER es:

$$\text{CER} = \frac{S + D + I}{N} \tag{6}$$

Donde:
- S es el número de sustituciones de caracteres,
- D es el número de eliminaciones de caracteres,
- I es el número de inserciones de caracteres,
- N es el número total de caracteres en la secuencia de referencia.
Al igual que WER, el CER se expresa en porcentaje, y un valor más bajo indica un mejor rendimiento del sistema de reconocimiento.

II-A3. Estado del Arte: Hoy en día existen variadas arquitecturas de Deep Learning para la tarea de ASR, cada una con sus ventajas y desventajas, es por esto que se utiliza como referencia el Open ASR Leaderboard de Hugginface, donde se evalúan diversos modelos frente a distintos datasets. Dado que evaluar un sistema de reconocimiento de voz es una tarea difícil, en el Open ASR Leaderboard se utiliza la estrategia de evaluación multi-dataset propuesta en ESB [12] para obtener evaluaciones robustas de cada modelo. ESB es un punto de referencia para evaluar el rendimiento de un único sistema de reconocimiento automático de voz (ASR) en un amplio conjunto de conjuntos de datos de voz. Comprende ocho conjuntos de datos de reconocimiento de voz en inglés, que capturan una amplia gama de dominios, condiciones acústicas, estilos de hablantes y requisitos de transcripción. Como tal, proporciona una mejor indicación de cómo es probable que funcione un modelo en ASR descendente en comparación con evaluarlo en un solo conjunto de datos. La puntuación ESB se calcula como un macropromedio de las puntuaciones WER en todos los conjuntos de datos de ESB. Los modelos en la tabla de clasificación se clasifican según sus puntuaciones WER promedio, de menor a mayor, y los dataset utilizados son:

- LibriSpeech: LibriSpeech [13] es un corpus de aproximadamente 1000 horas de voz en inglés leída a 16 kHz. Los datos se derivan de audiolibros leídos del proyecto LibriVox y han sido cuidadosamente segmentados y alineados.
- Common Voice 9: Common Voice [14] consta de 14973 horas validadas en 93 idiomas junto con sus transcripciones, en el conjunto de datos también incluyen metadatos demográficos como edad, sexo y acento que pueden ayudar a mejorar la precisión de los motores de reconocimiento de voz.
- VoxPopuli: VoxPopuli [15] es un corpus de habla multilingüe. Los datos se recopilan de grabaciones de eventos del Parlamento Europeo de 2009-2020. Esta implementación contiene datos de voz transcritos para 18 idiomas. También contiene 29 horas de datos de voz transcritos en inglés no nativo destinado a la investigación en ASR para habla con acento.
- TED-LIUM: TED-LIUM [16] son charlas TED en inglés, con transcripciones, muestreadas a 16 kHz. Las tres versiones del corpus oscilan entre 118 y 452 horas de datos de voz transcritos.
- GigaSpeech: GigaSpeech [17] es un corpus de reconocimiento de voz en inglés multidominio y en evolución con 10.000 horas de audio etiquetado de alta calidad. Los datos de audio transcritos se recopilan de audiolibros, podcasts y YouTube, y cubren estilos de lectura y de habla espontánea, y una variedad de temas, como artes, ciencias, deportes, etc.
- SPGISpeech: SPGISpeech [18] es un corpus de 5000 horas de audio financiero transcrito profesionalmente. SPGISpeech contiene una amplia muestra representativa de acentos en inglés, calidad de audio que varía mucho y habla tanto espontánea como narrada. Cada una de las transcripciones ha sido verificada por varios editores profesionales para garantizar una alta precisión y están completamente formateadas, incluidas las mayúsculas, la puntuación y la desnormalización de palabras no estándar. SPGISpeech consta de 5000 horas de llamadas grabadas sobre resultados de empresas y sus respectivas transcripciones. SPGISpeech contiene aproximadamente 50.000 hablantes. El formato de cada archivo WAV es audio de un solo canal, 16 kHz y 16 bits.
- Earnings-22: Earnings 22 [19] es un corpus de 119 horas de llamadas de ganancias en inglés recopiladas de empresas globales. El objetivo principal es servir como punto de referencia para los modelos de ASR industriales y académicos en el habla con acento del mundo real.
- AMI: AMI Meeting [20] es un corpus que consta de 100 horas de grabaciones de reuniones. Las grabaciones utilizan una variedad de señales sincronizadas con una línea de tiempo común. Estos incluyen micrófonos para hablar de cerca y de campo lejano, cámaras de video individuales y de vista de sala, y

salida de un proyector de diapositivas y una pizarra electrónica. Durante las reuniones, los participantes también tienen a su disposición bolígrafos no sincronizados que registran lo escrito. Las reuniones se grabaron en inglés utilizando tres salas diferentes con diferentes propiedades acústicas e incluyeron en su mayoría hablantes no nativos.

También en ESB incluyen una métrica de latencia, Real Time Factor (RTF). El factor de tiempo real es una medida de la latencia de los sistemas automáticos de reconocimiento de voz, es decir, cuánto tiempo tarda un modelo en procesar una determinada cantidad de voz. Generalmente, se expresa como un múltiplo del tiempo real. Un RTF de 1 significa que procesa la voz tan rápido como se pronuncia, mientras que un RTF de 2 significa que tarda el doble. Por tanto, un valor RTF más bajo indica una latencia más baja.

El modelo escogido para esta investigación fue Whisper de OpenAI, un modelo Open-Source que se encuentra en el top del leaderboard Open ASR.

II-A4. Whisper: Whisper [21] es un modelo de reconocimiento de voz de propósito general. Está entrenado en un gran conjunto de datos de audio diverso y también es un modelo multitarea que puede realizar reconocimiento de voz multilingüe, traducción de voz e identificación de idioma.

La arquitectura Whisper es un enfoque simple de extremo a extremo, implementado como un Transformer encoder-decoder. El audio de entrada se divide en fragmentos de 30 segundos, se convierte en un espectrograma log-Mel y luego se pasa a un encoder. Se entrena un decoder para predecir el título de texto correspondiente, entremezclado con tokens especiales que dirigen al modelo único a realizar tareas como identificación de idioma, marcas de tiempo a nivel de frase, transcripción de voz multilingüe y traducción de voz al inglés.

Los modelos Whisper están entrenados para tareas de reconocimiento y traducción de voz, siendo capaces de transcribir audio de voz al texto en el idioma en que se habla (ASR), así como traducirlo al inglés (traducción de voz). Los investigadores de OpenAI desarrollaron modelos para estudiar la solidez de los sistemas de procesamiento del habla entrenados bajo una supervisión débil a gran escala. Existen 7 modelos de diferentes tamaños y capacidades, resumidos en la siguiente tabla.

Tamaño	Parámetros	Solo Inglés	Multilingüe
Tiny	39 M	✓	✓
Base	74 M	✓	✓
Small	244 M	✓	✓
Medium	769 M	✓	✓
Large	1550 M	x	✓
Large-V2	1550 M	x	✓
Large-V3	1550 M	x	✓

Los modelos se entrenan con 680.000 horas de audio y las

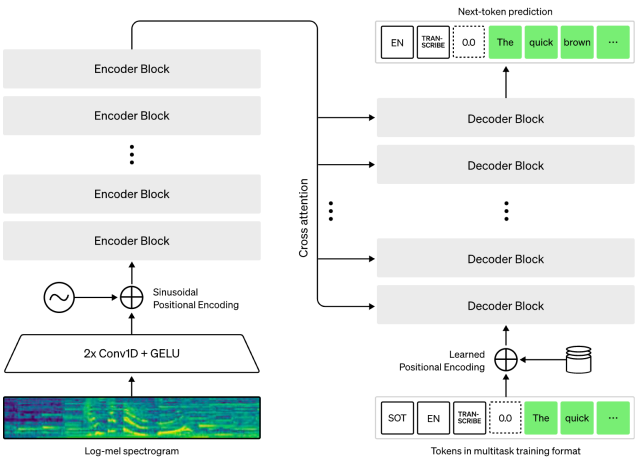

Figura 1. Arquitectura Whisper, Fuente: OpenAI.

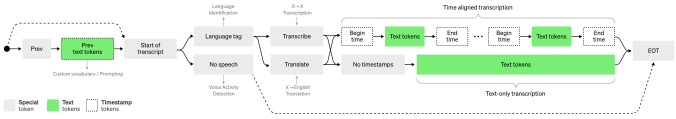

Figura 2. Tokens MultiTask Training Format, Fuente: OpenAI.

transcripciones correspondientes recopiladas de Internet. El 65 % de estos datos (o 438.000 horas) representa audio en inglés y sus transcripciones, aproximadamente el 18 % (o 126.000 horas) representa audio en otros idiomas y transcripciones en inglés, mientras que el 17 % final (o 117.000 horas) representa audio en inglés y las transcripciones en el idioma correspondiente. Estos datos no ingleses representan 98 idiomas diferentes.

II-A5. LIME: LIME [1] es un modelo agnóstico que explica de manera local las predicciones de un modelo caja negra f mediante la aproximación de f con un modelo explicable g en una vecindad de la instancia de interés $\boldsymbol{x}$. LIME utiliza una representación interpretable $\boldsymbol{x'} \in \{0,1\}^{d'}$ de la instancia $\boldsymbol{x} \in \mathbb{R}^d$. De esta forma, el dominio de la explicación g es $\{0,1\}^{d'}$ y, por lo tanto, g actúa sobre la presencia o ausencia de componentes interpretables. Para generar la vecindad se utiliza una función $\pi_x(z)$ que mide la distancia entre dos puntos x y z en el espacio de atributos interpretable. Adicionalmente, dado que la idea es aproximar f mediante un modelo interpretable, se considera una penalización Ω que mide la complejidad de g. Finalmente, y considerando una función de pérdida $\mathcal{L}(f, g, \pi_x)$ que mide la imprecisión del modelo g al aproximar f en la vecindad de $\boldsymbol{x}$ de acuerdo a la función de distancia π, la explicación producida por LIME que asegura interpretabilidad y fidelidad local se obtiene según la siguiente fórmula:

$$\xi(\boldsymbol{x}) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g). \tag{7}$$

Donde G es el conjunto de todos los modelos interpretables considerados, como por ejemplo modelos lineales o árboles de decisión.

II-A6. Representation Erasure: El borrado de representaciones [22] (Representation Erasure) es un método post-hoc y agnóstico al modelo para entender y explicar las decisiones de un modelo de red neuronal mediante la observación de los efectos que tiene el borrar distintas partes de la representación del modelo. Esto puede incluir dimensiones de vectores de palabras de entrada, unidades ocultas intermedias o palabras de entrada. Se utilizan varias técnicas para analizar los efectos de dicho borrado, como calcular la diferencia relativa en métricas de evaluación o usar aprendizaje por refuerzo para borrar el conjunto mínimo de palabras de entrada con el fin de cambiar la decisión de un modelo neural. Este método ayuda a ofrecer explicaciones claras sobre las decisiones de modelos neuronales y también facilita el análisis de errores en dichos modelos.

II-B. Conjunto de datos

En la presente investigación se utilizará el conjunto de datos CommonVoice [5]. En particular, se utilizarán las grabaciones en inglés asociadas a la versión 11 de CommonVoice disponible en HuggingFace Datasets.

Los conjunto de entrenamiento, validación y prueba consisten en 948736, 16354 y 16354 grabaciones de audio respectivamente. Cada grabación de audio tiene asociada una transcripción. Adicionalmente, cada grabación de audio fue grabada con un sampling rate de 48 kHz.

II-C. Descripción propuesta

La propuesta busca evaluar la explicabilidad de modelos estado del arte en la tarea de reconocimiento de voz mediante métodos post-hoc.

II-C1. SLIME: Para explicar las transcripciones del modelo Whisper, se propone el algoritmo SLIME (Speech LIME) que modela cómo diferentes modificaciones en la señal de audio afectan la calidad de la transcripción.

SLIME fragmenta una grabación de audio en segmentos de 100 milisegundos y considera como representación interpretable un vector booleano que indica la presencia o ausencia de cada segmento. Utilizando esta representación, se genera una vecindad de 100 instancias apagando segmentos de manera aleatoria. Cada instancia de la vecindad fue transcrita por Whisper, y mediante la distancia de Levenshtein entre la transcripción perturbada y la original se cuantificó el impacto de los silencios, proporcionando una base para una comparación cuantitativa entre las transcripciones. Finalmente, se ajustó un modelo explicable de regresión lineal considerando la representación interpretable de los audios como entrada y la distancia de Levenshtein como salida, lo cual permitió evaluar la importancia de los segmentos.

Debido a la alta demanda computacional que implica obtener predicciones de Whisper, se limitó la vecindad exploratoria a 100 combinaciones. Esto estableció un equilibrio entre la profundidad del análisis y las restricciones prácticas.

Los coeficientes resultantes de la regresión lineal ofrecen una interpretación directa de la relevancia de cada segmento de audio. Los segmentos asociados con coeficientes negativos son clave; su presencia es vital para mantener la fidelidad de la transcripción original, mientras que su ausencia es indicativa de un incremento en la distancia de Levenshtein, reflejando transcripciones que se desvían de la original. Esta información es útil para discernir las partes del audio que el modelo considera importantes para una transcripción precisa. El siguiente paso es mejorar la segmentación del audio, buscando métodos que puedan captar con mayor precisión las unidades lingüísticas significativas y mejorar la interpretación de las decisiones del modelo.

En 1 se resume el algoritmo propuesto para obtener las explicaciones de Whisper mediante LIME.

Algorithm 1 SLIME

$S \leftarrow \emptyset$
$f \leftarrow$ Black-box model
$n \leftarrow$ Number of segments of 100 ms in audio
$x \leftarrow$ Original audio instance
$y \leftarrow f(x)$
$x' \leftarrow$ Vector of n ones
$P \leftarrow$ Generate 100 Binomial perturbations of x'
for each p in P do
    $z \leftarrow$ Apply p to x
    $d \leftarrow$ Levenshtein distance between y and $f(z)$
    $z' \leftarrow$ Apply random mask to x'
    $S \leftarrow S \cup (z', d)$
end for
$g \leftarrow$ Fit linear model to S
end algorithm

II-C2. Borrado de Representaciones: Adicionalmente, analizaremos los cambios en la salida del modelo Whisper al borrar ciertas partes de la representación de la señal de audio a través de MFCCs. Sea f un modelo neuronal de ASR, dado un ejemplo $x \in E$ con una transcripción y, calcularemos el score de una métrica M con la siguiente fórmula:

$$S_M(x, y) = M(y, f(x)) \tag{8}$$

Ahora sea d una dimensión de nuestro vector x la cual queremos explorar, denotaremos $S_M(x, y, \neg d)$ el score

cuando la dimensión d es removida; esto quiere decir que todos sus valores son 0. La importancia de la dimensión d denotada por $I(d)$ estará dada por la diferencia relativa entre $S_M(x, y)$ y $S_M(x, y, \neg d)$:

$$I(d) = \frac{1}{|E|} \sum_{x \in E} \frac{S_M(x, y) - S_M(x, y, \neg d)}{S_M(x, y)} \tag{9}$$

Para esto utilizaremos las métricas mencionadas en el marco teórico y calcularemos la importancia de cada dimensión con cada métrica dada la diferencia relativa entre los outputs del modelo.

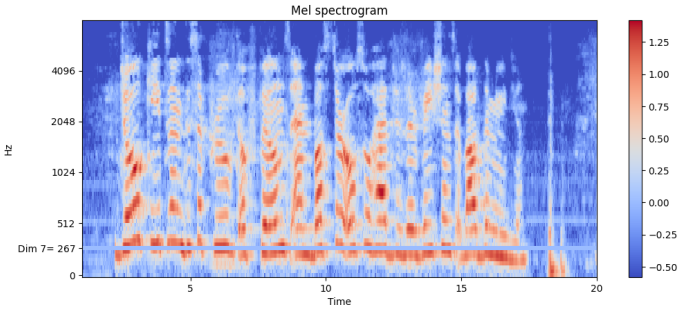

Figura 3. Ejemplo: "Joe Keaton disapproved of films, and Buster also had reservations about the medium." y su espectrograma de MEL, al eliminar la dimensión 7 del MFCC, definiendo todos sus valores en 0.

III. Resultados preliminares

III-A. Entorno de experimentación

El entorno de experimentación utilizado para realizar esta investigación fue el siguiente:

- CPU: Apple M1 Pro 10 Cores.
- GPU: Apple GPU 16 Cores.
- RAM: 16G B LPDDR5.
- OS: macOS 14.0 Sonoma.
- Software: Python 3.10, PyTorch 2.1.0, Transformers 4.35.0, Librosa 0.10.1, Datasets 2.14.6, NumPy 1.25.0, Pandas 2.1.2, NLTK 3.8.1, Scikit-learn 1.3.2, JIWER 3.0.3.

Para todos los experimentos se definió una semilla para poder generar resultados reproducibles, la semilla utilizada fue 42. Como modelo se utilizó whisper-tiny con 39 Millones de parámetros.

III-B. SLIME

Consideremos una grabación de audio x cuyo contenido es they perfectly illuminate the flowerbeds of gardens. Al aplicar SLIME a esta grabación se obtiene la explicación de la Figura 4.

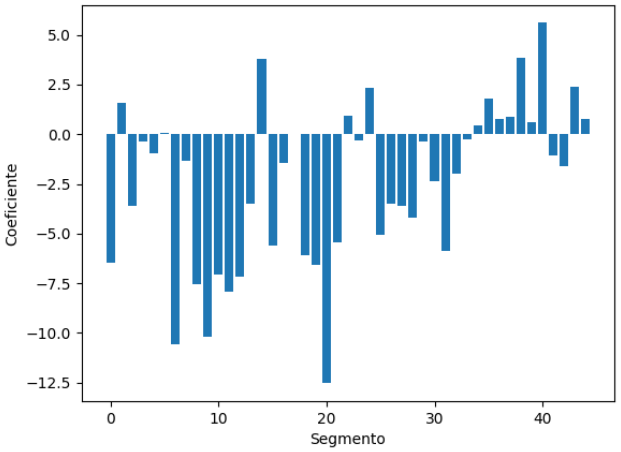

Figura 4. Explicación de SLIME

Como se mencionó anteriormente, un coeficiente negativo indican que un segmento es importante, pues al removerlo la transcripción se desvía de la original considerando la distancia de Levenshtein.

III-C. Borrado de representaciones

En la figura 5 podemos observar el cálculo de importancias para distintas métricas evaluadas en el set de test del dataset Common Voice 11, cada celda muestra la importancia de una dimensión (columna) de los MFCCs en cada métrica (fila) para el modelo entrenado.

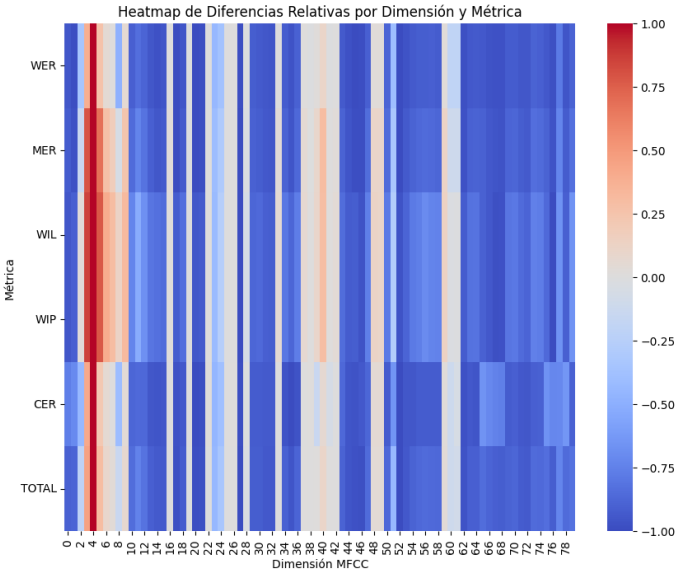

Figura 5. Mapa de calor de importancias $I(d)$ Normalizadas entre [-1,1] calculado usando la ecuación 9.

Se puede visualizar como la dimensión 4 es la más importante para la métrica WER y CER. Una importancia

mayor implica que al remover esa dimensión las predicciones del modelo empeoran con respecto a la métrica evaluada, por ende es una dimensión importante para el modelo.

Al hacer la transformación inversa de los coeficientes de MEL podemos notar como la dimensión 4 equivale a la frecuencia de 153 Hz, esto nos indica que Whisper se basa en las frecuencias más bajas para realizar buenas predicciones.

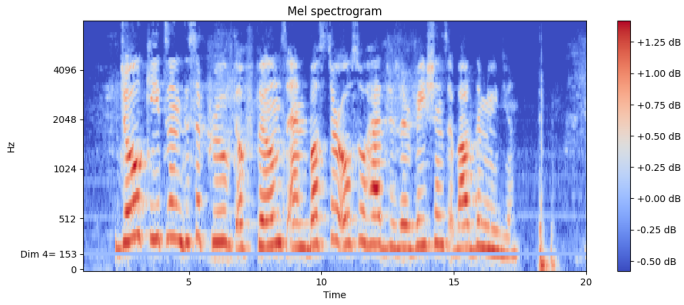

Figura 6. Ejemplo: "The new patch is less invasive than the old one, but still causes regressions." y su espectrograma del MEL al borrar la dimensión 4.

Para comprender mejor las interacciones del modelo, en cada paso se eliminó la dimensión más importante (Ver IV), pudimos observar que las 3 dimensiones más importantes son la dimensión 4,8 y 1 que corresponden a 153 Hz, 305 Hz y 38 Hz respectivamente, todas estas bandas de MEL corresponden a frecuencias bajas del espectrograma, lo que nos revela que Whisper se enfoca mayoritariamente en las frecuencias bajas. Esto puede suceder por varias razones:

1. Información Fundamental en Frecuencias Bajas: Las frecuencias bajas en un espectrograma de MEL suelen contener la mayor parte de la energía vocal y son cruciales para entender la voz humana. Estas frecuencias incluyen tonos fundamentales y formantes, que son esenciales para identificar sonidos de vocales y consonantes.
2. Robustez en Ambientes Ruidosos: Las frecuencias más bajas son menos susceptibles a interferencias de ruido de fondo, lo que hace al modelo más eficiente en entornos ruidosos. Este aspecto es especialmente importante en aplicaciones de la vida real donde el audio no siempre es claro.
3. Mejor Reconocimiento de Voz: Al centrarse en las bandas bajas, Whisper puede ser más efectivo en reconocer y entender diferentes acentos y modulaciones de voz. Además, estas frecuencias suelen ser más estables y consistentes entre diferentes personas, lo que facilita la generalización del modelo.
4. Eficacia en Datos Comprimidos: Las frecuencias bajas a menudo se preservan mejor en grabaciones de audio comprimidas, lo que es común en muchos formatos de audio digitales. Al centrarse en estas frecuencias, Whisper puede mantener un alto rendimiento incluso con datos de menor calidad.
5. Identificación de Hablantes: Las características de las frecuencias bajas pueden ayudar a diferenciar entre distintos hablantes, lo que es útil en situaciones donde hay múltiples personas hablando.

También podemos observar que hay importancias negativas, esto quiere decir que al eliminar esa dimensión las predicciones del modelo mejoran, lo que es un hallazgo muy interesante, ya que podríamos optimizar al modelo borrando ciertas dimensiones.

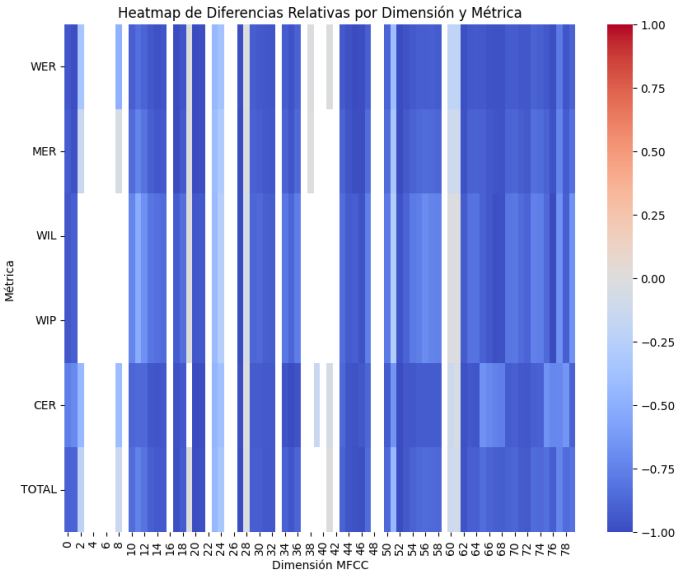

Figura 7. Mapa de calor de importancias $I(d)$ negativas.

Es por esto que proponemos ORTREL un método de aprendizaje por refuerzo para aprender la representación óptima de los espectrogramas para Whisper, eliminando las representaciones con importancias negativas para aumentar el desempeño del modelo una vez ya está entrenado.

III-D. ORTREL (Optimal Representations Trough Reinforcement Learning)

ORTREL ...

Referencias

[1] M. T. Ribeiro, S. Singh y C. Guestrin, "Why Should I Trust You?": Explaining the Predictions of Any Classifier, 2016. arXiv: 1602.04938 [cs.LG].

[2] S. Lundberg y S.-I. Lee, A Unified Approach to Interpreting Model Predictions, 2017. arXiv: 1705.07874 [cs.AI].

[3] N. Kokhlikyan, V. Miglani, M. Martin et al., «Captum: A unified and generic model interpretability library for PyTorch,» CoRR, vol. abs/2009.07896, 2020. arXiv: 2009.07896. dirección: https://arxiv.org/abs/2009.07896.

[4] X. Wu, P. Bell y A. Rajan, «Explanations for Automatic Speech Recognition,» en ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2023, págs. 1-5. doi: 10.1109/ICASSP49357.2023.10094635.

[5] R. Ardila, M. Branson, K. Davis et al., «Common Voice: A Massively-Multilingual Speech Corpus,» en Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), 2020, págs. 4211-4215.

[6] Y. Sun, H. Chockler, X. Huang y D. Kroening, Explaining Image Classifiers using Statistical Fault Localization, 2020. arXiv: 1908.02374 [cs.LG].

[7] H. Chockler, D. Kroening e Y. Sun, Compositional Explanations for Image Classifiers, mar. de 2021.

[8] X. Wu, P. Bell y A. Rajan, Can We Trust Explainable AI Methods on ASR? An Evaluation on Phoneme Recognition, 2023. arXiv: 2305.18011 [cs.CL].

[9] J. Garofolo, L. Lamel, W. Fisher et al., «TIMIT Acoustic-phonetic Continuous Speech Corpus,» Linguistic Data Consortium, nov. de 1992.

[10] V. Haunschmid, E. Manilow y G. Widmer, «audioLIME: Listenable Explanations Using Source Separation,» CoRR, vol. abs/2008.00582, 2020. arXiv: 2008.00582. dirección: https://arxiv.org/abs/2008.00582.

[11] I. Goodfellow, Y. Bengio y A. Courville, Deep Learning. MIT Press, 2016, http://www.deeplearningbook.org.

[12] S. Gandhi, P. von Platen y A. M. Rush, ESB: A Benchmark For Multi-Domain End-to-End Speech Recognition, 2022. arXiv: 2210.13352 [cs.CL].

[13] V. Panayotov, G. Chen, D. Povey y S. Khudanpur, «Librispeech: An ASR corpus based on public domain audio books,» en 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2015, págs. 5206-5210. doi: 10.1109/ICASSP.2015.7178964.

[14] R. Ardila, M. Branson, K. Davis et al., Common Voice: A Massively-Multilingual Speech Corpus, 2020. arXiv: 1912.06670 [cs.CL].

[15] C. Wang, M. Rivière, A. Lee et al., VoxPopuli: A Large-Scale Multilingual Speech Corpus for Representation Learning, Semi-Supervised Learning and Interpretation, 2021. arXiv: 2101.00390 [cs.CL].

[16] A. Rousseau, P. Deléglise e Y. Estéve, «TED-LIUM: an Automatic Speech Recognition dedicated corpus,» en Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), N. Calzolari, K. Choukri, T. Declerck et al., eds., Istanbul, Turkey: European Language Resources Association (ELRA), mayo de 2012, págs. 125-129. dirección: http://www.lrec-conf.org/proceedings/lrec2012/pdf/698_Paper.pdf.

[17] G. Chen, S. Chai, G. Wang et al., GigaSpeech: An Evolving, Multi-domain ASR Corpus with 10,000 Hours of Transcribed Audio, 2021. arXiv: 2106.06909 [cs.SD].

[18] P. K. O'Neill, V. Lavrukhin, S. Majumdar et al., SPGISpeech: 5,000 hours of transcribed financial audio for fully formatted end-to-end speech recognition, 2021. arXiv: 2104.02014 [cs.CL].

[19] M. D. Rio, P. Ha, Q. McNamara, C. Miller y S. Chandra, Earnings-22: A Practical Benchmark for Accents in the Wild, 2022. arXiv: 2203.15591 [cs.CL].

[20] P. v. Platen, C. Zhang y P. Woodland, «Multi-Span Acoustic Modelling Using Raw Waveform Signals,» en Interspeech 2019, ISCA, sep. de 2019. doi: 10.21437/interspeech.2019-2454. dirección: http://dx.doi.org/10.21437/Interspeech.2019-2454.

[21] A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey e I. Sutskever, Robust Speech Recognition via Large-Scale Weak Supervision, 2022. arXiv: 2212.04356 [eess.AS].

[22] J. Li, W. Monroe y D. Jurafsky, Understanding Neural Networks through Representation Erasure, 2017. arXiv: 1612.08220 [cs.CL].

IV. Anexos

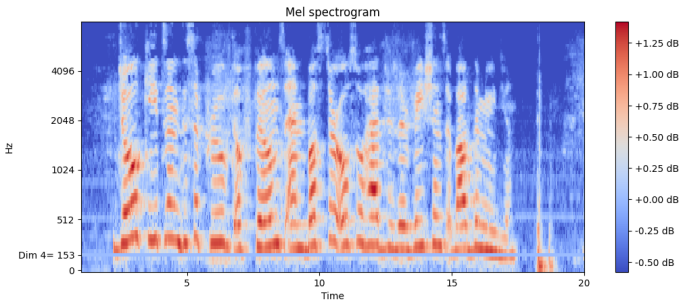

Figura 8. Espectrograma de potencia de MEL al eliminar la banda 4, equivalente a 153 Hz

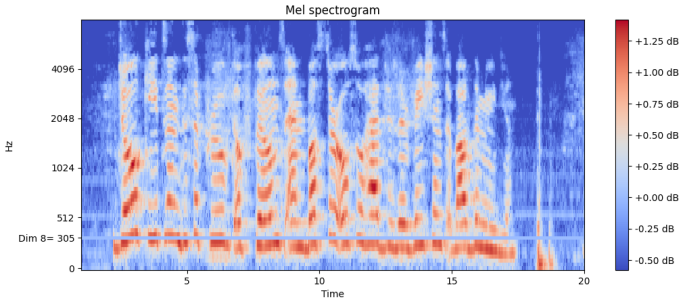

Figura 9. Espectrograma de potencia de MEL al eliminar la banda 8, equivalente a 305 Hz

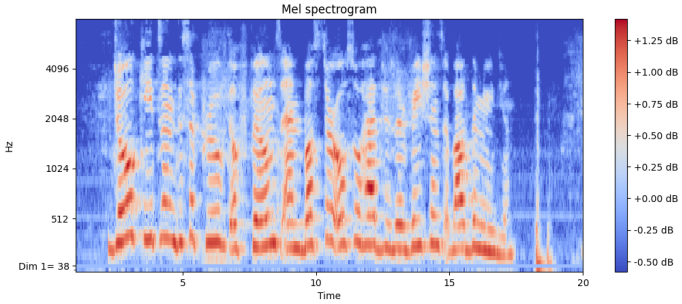

Figura 10. Espectrograma de potencia de MEL al eliminar la banda 1, equivalente a 38 Hz

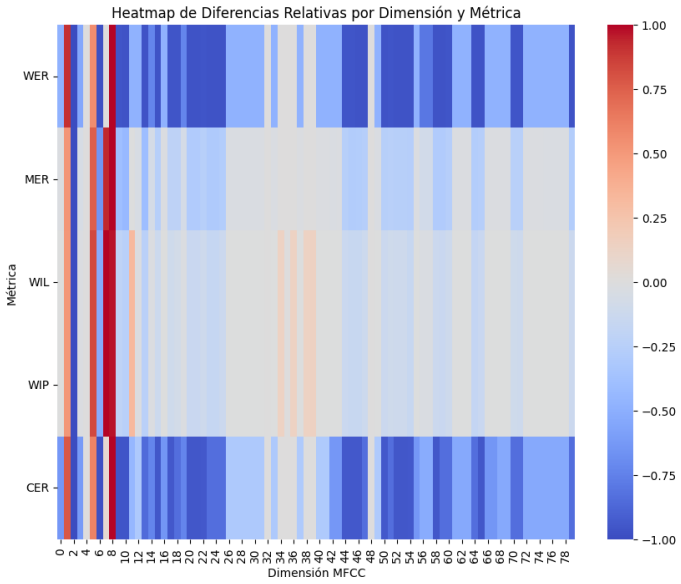

Figura 11. Mapa de Importancias al eliminar la dimensión 4 de los MFCC

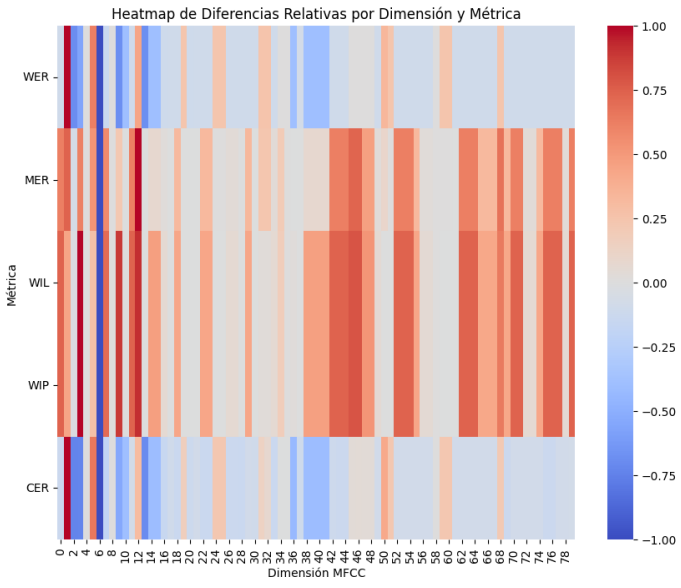

Figura 12. Mapa de Importancias al eliminar la dimensión 4 y 8 de los MFCC

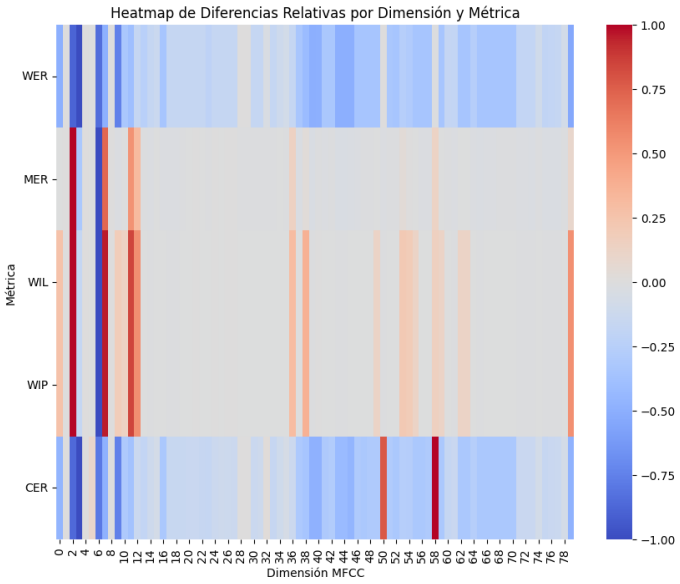

Figura 13. Mapa de Importancias al eliminar la dimensión 4, 8 y 1 de los MFCC

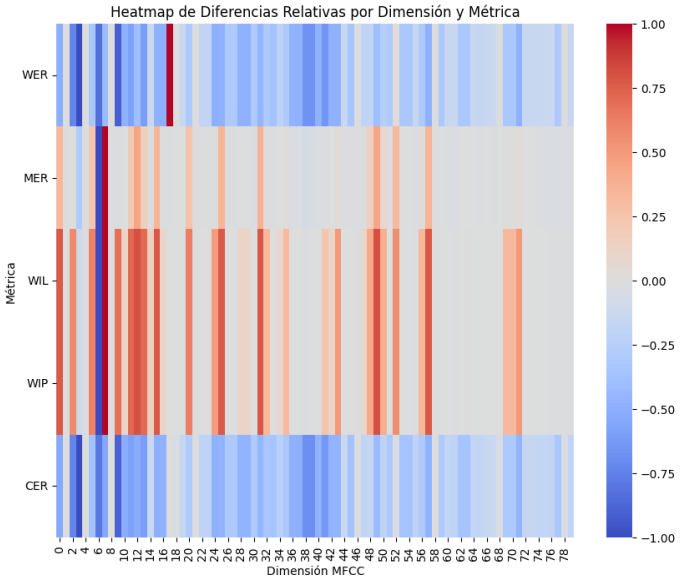

Figura 14. Mapa de Importancias al eliminar la dimensión 4, 8, 1, 58 de los MFCC