# Análisis comparativo de esquemas de entrenamiento para la transferencia de estilo en voz en Español

León Darío Arango Amaya*, Jose Alberto Arango Sánchez†,
Carlos Daniel Montoya Hurtado‡, Julián D. Arias Londoño§
Departamento de Ingeniería de Sistemas
Universidad de Antioquia
Medellín, Colombia
*leon.arango@udea.edu.co, †jose.arangos@udea.edu.co, ‡carlos.montoyah@udea.edu.co, §julian.ariasl@udea.edu.co

***Resumen*—La conversión de voz es un campo en creciente estudio con la llegada de arquitecturas de redes neuronales profundas que permiten transferir la voz de un hablante fuente a uno objetivo, las cuales se han probado principalmente en el idioma inglés. En este informe se compara, a través de medidas perceptuales, el desempeño de un modelo al realizar transferencia de estilo en el idioma español usando tres enfoques diferentes: entrenándolo con audios en inglés y español, utilizando características previamente aprendidas en el idioma inglés para ajustarlas al español, y finalmente entrenándolo únicamente con audios en español. También se exploran medidas de similitud basadas en la distancia euclidiana y la Divergencia de Kullback-Leibler para intentar explicar las variaciones observadas en las conversiones de voz entre diferentes hablantes. Adicionalmente, se describe la aplicación construida para exponer el mejor modelo obtenido como un servicio web. Se concluyó que para una mejor transferencia en el idioma español, es necesario entrenar a este modelo directamente con audios en español.**

***Palabras Claves*—Conversión de voz, transferencia de estilo, hablantes, similitud entre hablantes, aprendizaje profundo**

## I. Introducción

La inteligencia artificial es uno de los campos más relevantes actualmente, está permitiendo el desarrollo de sistemas informáticos con la capacidad de ejecutar tareas que normalmente necesitan inteligencia humana, por ejemplo, el servicio al cliente automatizado, la traducción entre idiomas, el reconocimiento de objetos, etc. Más específicamente el Machine Learning y el Deep Learning han hecho posible abarcar estos mismos problemas de tal manera que las máquinas, o más específicamente los modelos matemáticos, se entrenen para que tomen decisiones por sí mismos con base a ciertas reglas y criterios. A todo esto, sumándole que el avance en la tecnología y el hardware hacen que cada vez sea posible el procesamiento de más datos y por tanto de modelos matemáticos más complejos.

Desde el 2015 el Deep Learning tiene un gran campo en crecimiento que es el Neural Style Transfer (NST) [1], arquitecturas que permiten manipular imágenes o videos para que adopten los estilos de otra imágen (existen varios ejemplos de pinturas famosas de artistas como Vincent van Gogh) a través de redes neuronales. Ahora también se ha llevado este mismo objetivo pero a los sonidos, intentando transferir los estilos de una voz o canción a otra, o también el problema de convertir un texto a voz el cual ha sido explorado incluso desde mucho antes de ser abarcado por el Deep Learning. A pesar de que hay más estudios en esta área, la mayoría de estos trabajos han sido desarrollados para funcionar con el idioma inglés, por lo que en este proyecto se tiene como objetivo comprender y hacer uso de una arquitectura convolucional para realizar transferencias de estilos en español; la arquitectura seleccionada fue previamente entrenada para realizar transferencias de estilos en inglés, por lo tanto se decidió atacar el problema desde dos esquemas de entrenamiento: desde cero y aplicando Transfer Learning, para determinar entre ellos cuál genera mejores resultados. Finalmente se desplegará una aplicación web que permita a los usuarios evaluar y usar el mejor modelo.

## II. Estado del arte

Con el objetivo de estudiar y comprender las transferencias de estilo por voz, se presentan en orden cronológico algunos trabajos relacionados. La transferencia de estilo en voz se comenzó a estudiar en el trabajo seminal de Chorowski y colaboradores [2]. El objetivo de los autores fue probar que es posible realizar transfer learning en modelos diseñados para procesar la voz, de forma similar a como se realiza con imágenes, además se trata de realizar con pocos datos del hablante y se menciona que se obtienen buenos resultados con una red neuronal convolucional entrenada para reconocer la voz.

En [3] se aclara que la clonación y la conversión de voz no son exactamente lo mismo, pero si están muy relacionadas. Los modelos que realizan clonación de voz buscan aprender la voz de un hablante a partir de unas pocas muestras para luego sintetizarla, generalmente a partir de un texto de entrada, mientras que los modelos de conversión de voz utilizan las señales de audio, modificándolas para hacer que el mensaje emitido por un hablante conocido como fuente sea reproducido en la voz de otro hablante, conocido como objetivo, conservando el contenido lingüístico del mensaje. Este trabajo propone una arquitectura basada en redes neuronales convolucionales usado como un modelo generativo, con algunos módulos de pre y post procesado. Además, como punto interesante, emplean un *vocoder Griffin-Lim*, el cual es un decodificador de voz con

un algoritmo que no requiere ningún tipo de entrenamiento, y para evaluar el desempeño de su arquitectura utilizan similitud y naturalidad como medidas.

La creciente adopción de las GAN (Generative Adversarial Networks) como modelos generativos en distintos problemas hizo que pronto se introdujeran al campo de la conversión de voz. En [4] se propone un modelo generativo que pretende imitar la calidad y el estilo en la voz de un hablante utilizando este tipo de redes, con las que se busca generar muestras sintéticas indistinguibles de las reales. Al igual que otros trabajos, utiliza Griffin-Lim como decodificador de voz.

Por otro lado, la arquitectura planteada en [5] se compone de un codificador de hablante para capturar las características y un sintetizador que realiza un proceso secuencia a secuencia para admitir múltiples hablantes. Además se utiliza un modelo pre entrenado con ciertos parámetros congelados para tener una convergencia más rápida y un decodificador de voz neuronal que utiliza WaveNet auto regresivo para convertir los espectrogramas a través de treinta capas convolucionales. Estos componentes (codificador, decodificador y sintetizador) son comunes en una amplia gama de implementaciones de arquitecturas de clonación y conversión de voz. Los audios generados por este modelo fueron evaluados utilizando el Mean Opinion Score (MOS), el cual es una prueba subjetiva para medir el desempeño de los audios generados. Este método de evaluación ha sido adoptado por múltiples autores para medir el desempeño de sus modelos.

En [6] se dan a conocer tres grandes problemas que existen en la conversión de voz; el primero se trata de que muchos modelos asumen la disponibilidad de datos paralelos (audios que contienen las mismas oraciones), éste tipo de datos no es tan abundante y existen pocos modelos capaces de entrenar con datos no paralelos. La segunda habla sobre los pocos algoritmos que existen para trabajar con datos no paralelos, los cuales no sirven para conversiones de tipo *many to many*, es decir, de muchos hablantes fuente a muchos hablantes objetivo. Y el tercer problema expuesto dice que no existe un sistema de conversión *zero-shot*, el cual es un enfoque en el que el modelo debe ser capaz de convertir la voz de una hablante que jamás ha visto utilizando tan solo un par de audios. Además describen dos algoritmos utilizados comúnmente para resolver estos problemas. Por un lado están las GAN que dan buenos resultados, pero son difíciles de entrenar y de frágil convergencia. Por otro lado encuentra el CVAE (Conditional Variational Autoencoder), el cual es más fácil de entrenar, ya que sólo necesita realizar una reconstrucción del audio para ajustar la salida. Sin embargo, esta última aproximación no garantiza que la salida pertenezca a la misma distribución del hablante original.

El modelo propuesto por [7] responde a dos de los problemas expuestos por [6]. Esta arquitectura integra un *auto-encoder* que realiza la conversión de voz con una GAN que mejora la calidad de la transferencia. Cuenta con la capacidad de que un solo modelo es capaz de realizar conversión de voz a múltiples hablantes, y, adicionalmente, es posible entrenarla con datos no paralelos, lo que significa que los audios suministrados al modelo para la fase de entrenamiento no necesitan tener el mismo contenido.

## III. Métodos

### *III-A. Modelo seleccionado*

De las diferentes arquitecturas existentes para realizar conversión de voz, se decidió explorar en detalle la propuesta en [7] debido a su capacidad para trabajar con audios sin necesidad de tener la transcripción de su contenido. Además, los autores publicaron su implementación original, facilitando evaluar y experimentar con la arquitectura.

Cada audio es preprocesado para obtener dos vectores de características: uno que hace referencia a los Coeficientes Cepstrales en las Frecuencias de Mel (MFCC) y otro a las magnitudes de los valores complejos de coeficientes de Transformada de Fourier de corto plazo (STFT). Ambos vectores son convertidos a decibeles, normalizados y se transponen. Para el caso de los MFCC, las dimensiones resultantes son $(T, n_{mels})$, en donde $T$ es el número de ventanas y $n_{mels}$ es el número de bancos Mel a generar. Para la magnitud de la STFT se tiene un vector de dimensiones $(T, 1 + n_{fft}//2)$, en donde $n_{fft}$ es la longitud de la ventada a procesar.

La arquitectura está compuesta por 5 bloques que son entrenados en tres fases. La fase 0 consiste en un pre-entrenamieto, donde se entrena un autoencoder conformado por un *Encoder* que se encarga de extraer las características lingüísticas del audio, y un *Decoder* que reconstruye la señal de audio a un hablante objetivo indicado usando las características extraídas por el Encoder. Este autoencoder por sí solo es capaz de realizar conversión de voz. Para remover la identidad de los hablantes de las características extraídas en el proceso de codificación, se agrega el bloque *Classifier*, el cual es un clasificador que toma el resultado del Encoder y que sirve como un regularizador para el autoencoder. Este clasificador se entrena de manera alternada con el autoencoder en la fase 1, como se puede ver en la figura 1.

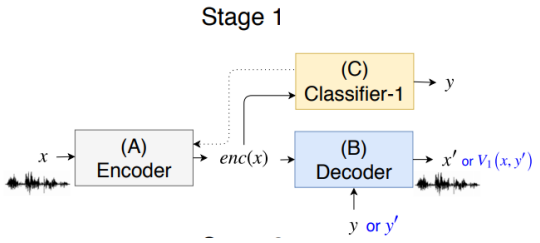



Figura 1: Fase 1 del entrenamiento. [7]

En la fase 2 se congelan los parámetros del autoencoder y se añade una GAN, compuesta por el bloque *Generator* y el bloque *Discriminator*, los cuales se pueden observar en la figura 2. El generador toma las características extraídas por el Encoder y con ellas genera la estructura fina de la señal, que se suma a la salida del Decoder, mejorando así la calidad del audio reconstruido. En esta fase, el generador y el discriminador se entrenan alternándose, el primero intentando

engañar al discriminador, y éste último intentando determinar si un audio generado es real o sintético.

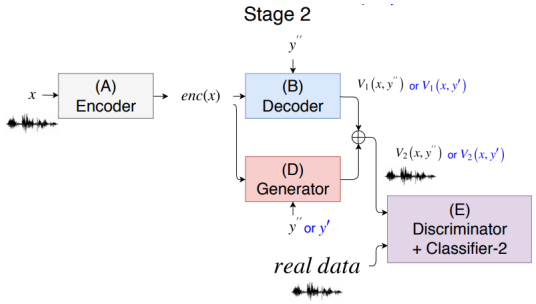



Figura 2: Fase 2 del entrenamiento. [7]

Al realizar pruebas preliminares con este modelo se encontraron variaciones en su desempeño dependiendo de la parejas de hablantes seleccionadas, es decir, perceptualmente se notó que algunas de las transferencias realizadas tenían mejor desempeño para ciertos hablantes objetivo dependiendo del audio ingresado al modelo, por lo que se planteó la necesidad de estudiar si una medida de similitud entre hablantes podría explicar estas variaciones.

### *III-B. Similitud entre hablantes*

No existe una manera estándar de medir la similitud entre hablantes, aunque en el estado del arte se han propuesto varias alternativas para realizar dicho proceso. En este trabajo se exploraron dos aproximaciones, una basada en la distancia euclidiana entre las características acústicas de los hablantes y otra basada en la aproximación simétrica de la divergencia de Kullback-Leibler [8], la cual mide la similitud o diferencia entre dos funciones de distribución de probabilidad, que para este caso son Modelos de Mezclas Gaussianas (GMM) asociados a cada hablante y que son construidos a partir de una adaptación realizada a un *Modelo de Background Universal* (UBM).

*III-B1. Medida basada en distancia euclidiana:* Para medir la distancia euclidiana entre hablantes se usó el enfoque propuesto en [9], donde se describe a través de un único valor la similitud entre hablantes, calculando la distancia euclidiana a partir de los MFCC como vector de características.

*III-B2. Medida basada en la aproximación simétrica de la divergencia de Kullback-Leibler:* La idea de este enfoque es crear para cada hablante un *Modelo de Mezclas Gaussianas* (GMM) para luego calcular la distancia entre las funciones de densidad de probabilidad de cada uno. Para crear el GMM para cada hablante individual primero es necesario crear un UBM, el cual es un gran GMM entrenado para representar la distribución de características generales del habla independientemente del orador. Nuestro UBM fue construido con 250 componentes, entrenándose con los 20 hablantes que usó [7] para entrenar originalmente su modelo.

Después de entrenar el UBM, se procede a adaptar el vector de medias del UBM, usando la estimación *Maximun a Posteriori* (MAP) [8], pero conservando la misma matriz de covarianza. Después de adaptar cada modelo, se procede a calcular la distancia entre las distribuciones de probabilidad de cada hablante usando como medida la Divergencia simétrica de Kullback-Leibker, la cual se define como:

$$D_s\left(\lambda_1, \lambda_2\right) = \frac{D\left(\lambda_1 \lambda_2\right) + D\left(\lambda_2, \lambda_1\right)}{2} \quad (1)$$

En donde $D\left(\lambda_1, \lambda_2\right)$ es la divergencia Kullback-Leibler data por:

$$D\left(\lambda_1, \lambda_2\right) = \frac{1}{T}\left[\log P\left(\mathbf{O}^{(2)}|\lambda_1\right) - \log P\left(\mathbf{O}^{(2)}|\lambda_2\right)\right] \quad (2)$$

$\lambda_i$ representa el modelo del hablante $i$ y $\mathbf{O}^{(i)}$ es un conjunto de $T$ muestras correspondientes al hablante $i$.

### *III-C. Desarrollo y despliegue del modelo*

Además del estudio de la transferencia de estilo en voz en español, se planteó como objetivo desplegar en un ambiente de producción, el modelo con el mejor desempeño durante la fase de validación. Para cumplir con este objetivo se desarrolló una plataforma web[1] en la cual se muestra información respecto al proyecto y el equipo de trabajo, se ofrece la funcionalidad de evaluar las transferencias de estilo obtenidas en el proceso de entrenamiento y además se brinda la opción de interactuar con el modelo, permitiendo a un usuario grabar un audio, procesarlo a un hablante objetivo y escuchar el resultado de aplicar el estilo del hablante seleccionado al audio grabado.

Para acceder a la plataforma, el usuario debe hacer uso de un navegador web. El servicio Amazon Route 53 de Amazon Web Services (AWS) permite exponer la interfaz gráfica de usuario a través de un dominio. La plataforma, cuya infraestructura se muestra en la figura 3, está compuesta por un *frontend* desarrollado en Angular, el cual usa el paquete recordrtc para grabar los audios de los usuarios. El servicio Amplify de AWS facilita el despliegue continuo de los cambios hechos en este componente. El frontend a se comunica a dos *backends*. El primero es un servidor web escrito en Node.js usando express y desplegado en un Dyno de Heroku, éste se encargó de almacenar los datos obtenidos en las encuestas en una instancia de MongoDB Atlas. El segundo es un servidor web escrito en python usando FastAPI y desplegado en un Compute Engine de Google Cloud Platform (GCP) dotado con una Tesla T4, el cual expone un servicio que permite realizar las transferencias de estilo; éste se apoya en los buckets del servicio S3 de AWS, en donde se almacenan el modelo, los audios usados para las encuestas y los audios grabados y generados al realizar la conversión de voz.

## IV. Experimentos

### *IV-A. Base de datos*

El modelo seleccionado fue entrenado originalmente con el corpus [10], el cual recoge a 109 hablantes nativos del idioma inglés, donde cada uno grabó aproximadamente 400 audios, de los cuales 399 poseen transcripción. Para el entrenamiento, los

[1]Disponible en https://www.voicestyletransfer.tech/

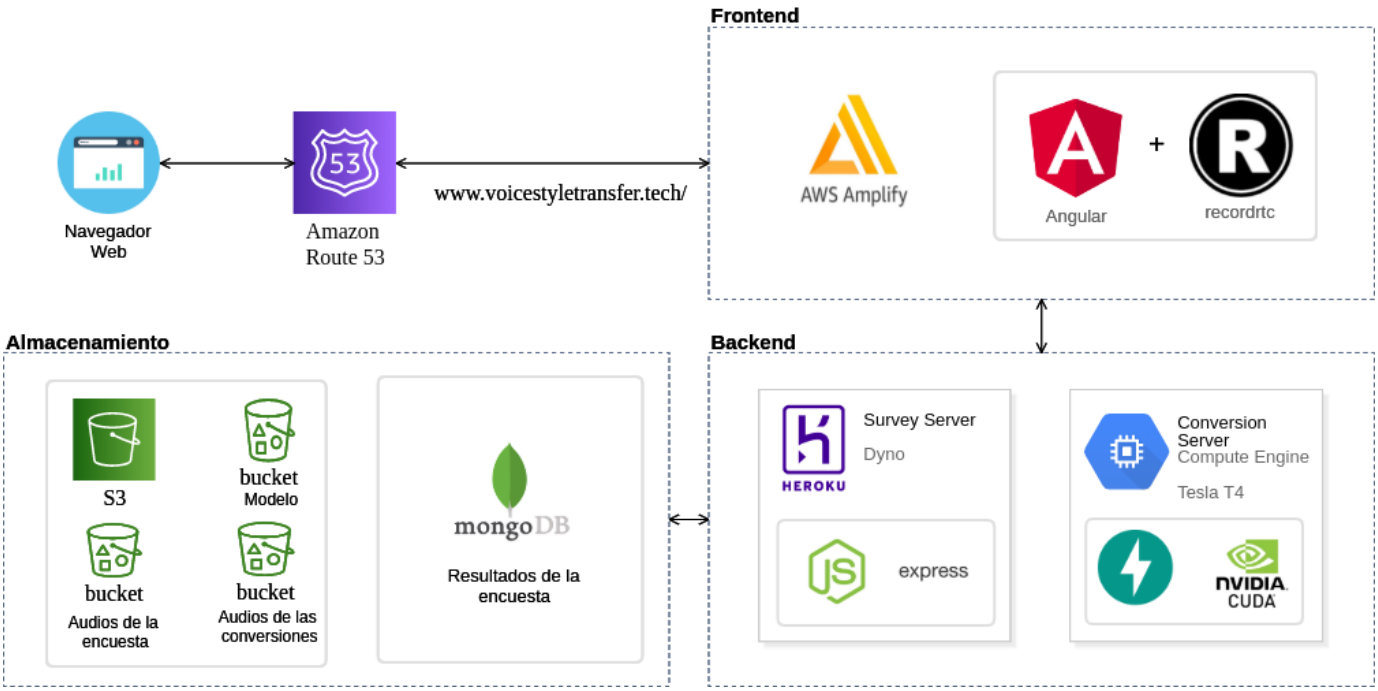



Figura 3: Infraestructura de la plataforma web.

integrantes del equipo de trabajo, León, Jose y Carlos Daniel, grabaron 400 audios con una duración aproximada de 6 segundos cada uno. Cada audio fue grabado usando una frecuencia de muestreo de 16kHz y almacenado en formato WAV. Cada hablante en español usó sus propias frases, aprovechando la capacidad del modelo para procesar datos no paralelos.

### *IV-B. Fase experimental*

Se entrenaron cuatro modelos diferentes usando como base el modelo descrito en la sección III-A con el objetivo de compararlos, para así determinar con cual aproximación se obtienen mejores resultados al realizar una transferencia de estilo entre hispanohablantes. Para el entrenamiento de los modelos se usó el mismo conjunto de hiperparámetros definido por [7], donde se específica que la red se entrenó usando un optimizador Adam con una tasa de aprendizaje $lr = 0.0001$, y un tamaño de lote de 32, y que cada fase tuvo un número de *mini-lotes* específico: 8000 para el autoencoder y 20000 para el clasificador en la fase de pre-entrenamiento, 80000 para la fase 1 y 50000 para la fase 2.

Los distintos experimentos se realizaron en un servidor proporcionado por el grupo In2Lab de la Universidad de Antioquia, dotado con un procesador Intel Core i7 6700, 40GB de memoria y una tarjeta de video GTX 1060 6GB. Con estas especificaciones y los hiperparámetros usados, la duración de un entrenamiento completo de un modelo era de aproximadamente tres días.

*IV-B1. Modelo 1 (M-Chou):* El primer modelo entrenado es una réplica del entrenamiento realizado en [7], utilizando el mismo conjunto de datos y de hiperparámetros, sin realizar modificaciones a la arquitectura propuesta. El objetivo de este experimento fue familiarizarnos con el modelo propuesto, su implementación y las herramientas necesarias para entrenarlo y usarlo correctamente.

*IV-B2. Modelo 2 (M-Chou+3):* El segundo modelo fue entrenado por completo con 19 de los 20 hablantes originales y, además, se añadieron los audios grabados al conjunto de datos, para un total de 22 hablantes, 19 del idioma inglés y 3 del idioma español. Con este conjunto de datos mixtos se buscó determinar si el modelo era capaz de extraer características de los hablantes en inglés, de los cuales hay mas datos, para así beneficiar el desempeño de las transferencias de estilo para los hablantes en español.

*IV-B3. Modelo 3 (M-3SS):* El tercer modelo fue entrenado únicamente con los audios grabados en español, para un total de 3 hablantes. Con este número limitado de datos se esperaba poder determinar si el modelo era capaz de extraer las características necesarias para hacer una transferencia de estilo.

*IV-B4. Modelo 4 (M-TL):* Para el cuarto y último modelo entrenado se decidió utilizar el concepto de *transfer learning* para intentar aprovechar los pesos previamente entrenados con los hablantes en inglés y luego, ajustarlos a los 3 hablantes en español. En esta aproximación se utilizaron como base los pesos del modelo *M-Chou*, descrito en la sección IV-B1.

Como el modelo *M-Chou* fue entrenado con 20 hablantes, se reemplazaron las capas cuya dimensión estaba determinada por el número de hablantes por unas nuevas capas ajustadas únicamente a 3 (el nuevo número de hablantes objetivo). Las capas reemplazadas se muestran en la tabla I. Cada capa tiene un impacto particular en el proceso de entrenamiento. En el caso del clasificador, la capa `conv9` se encarga de asignar la

probabilidad de que un audio sintetizado por el autoencoder haya sido producido por un hablante específico. De manera similar la capa `conv_clasiffy` del discriminador se encarga de determinar si un audio es real o generado por una máquina.

Tabla I: Capas reemplazadas para ajustar el modelo *M-TL*

| Componente | Capa |
|---|---|
| SpeakerClassifier | conv9 |
| Decoder | emb1, emb2, emb3, emb4, emb5 |
| Generator | emb1, emb2, emb3, emb4, emb5 |
| PatchDiscriminator | conv_classify |

Inicialmente se planeó un proceso de *ajuste fino*, reemplazando y congelando la menor cantidad de capas posibles. Sin embargo, el hecho de que las capas de *embedding* se ubicaran como las capas iniciales del Decoder y del Generator, implicó que todas las capas posteriores de ambos componentes debieran ser ajustadas. En el caso de los componentes SpeakerClassifier y PatchDiscriminator, se decidió congelar todos los pesos de las capas hasta la inmediatamente anterior a las mostradas en la tabla I, pero los resultados fueron inaudibles. Después de descongelar otras dos capas y tener resultados similares, finalmente se decidió no congelar los pesos de ningún componente. Después de configurar el modelo, se realizó el proceso de entrenamiento, respetando las fases e hiperparámetros definidos originalmente en [7].

### *IV-C. Evaluación de los modelos*

Para evaluar el desempeño de los modelos entrenados se realizó una prueba de opinión media [6]. Para ello, se diseñó una encuesta en la que a cada persona se le presentaron 6 parejas de audios seleccionados al azar por cada modelo entrenado en la fase de experimentos. Cada pareja consiste en dos audios con el mismo contenido, uno pronunciado por el hablante objetivo, y otro resultante de transferir un audio pronunciado pon un hablante desconocido al estilo del hablante objetivo. Por cada pareja de audios, se le pidió a la persona evaluar perceptualmente cuatro características, dos de ellas usadas en trabajos previos [6] [7], y las dos restantes son propuestas por nosotros, al considerarlas relevantes en el contexto del proyecto. Las características son:

1. **Similaridad:** ¿Qué tan parecidos son los dos audios presentados?
2. **Naturalidad:** ¿El audio generado parece entonado por una persona o por una máquina?
3. **Inteligibilidad:** ¿Qué tanto se entiende el contenido del audio sintético?
4. **¿Qué tan nativo suena?:** ¿La pronunciación del audio generado parece de una persona que tiene el español como lengua nativa? Con esto se busca determinar si las características extraídas de los hablantes de inglés tienen un impacto en la transferencia en los hablantes de español.

Antes de comenzar la encuesta, a cada persona se le presentó la definición de las características a evaluar y un ejemplo de cómo evaluarla. Para la encuesta se desarrolló una interfaz gráfica mostrada en la figura 4, buscando facilitar y amenizar su diligenciamiento.

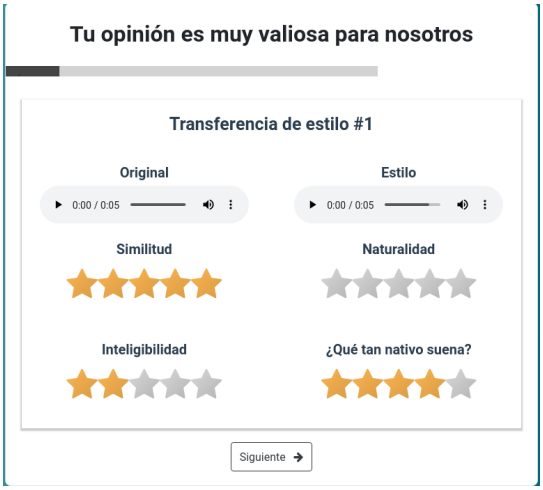



Figura 4: Interfaz propuesta para evaluar una transferencia de estilo

Todas las respuestas en las que todos los campos fueron evaluados con 0 se consideraron inválidas, y por ende fueron descartadas. Después de limpiar los datos, el resultado fue de 591 evaluaciones de diferentes audios provenientes de los 4 modelos entrenados. El recuento de respuestas válidas por modelo se muestra en la figura 5.

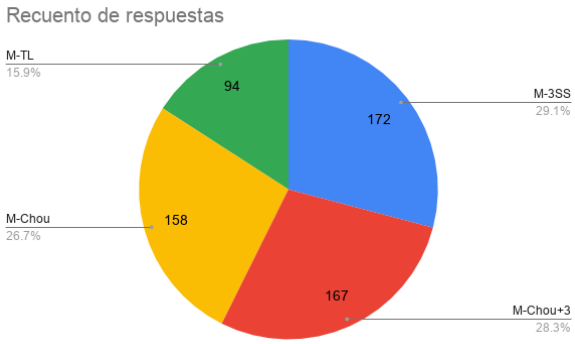



Figura 5: Recuento de respuestas válidas por modelo

## V. Resultados

Los datos recolectados en la encuesta se analizaron usando Google Data Studio. Para cada característica se calculó el promedio por cada uno de los modelos, esto se resume en la figura 6. Se puede notar que en todas las características el modelo M-Chou, que fue entrenado únicamente con hablantes en inglés, tiene el peor de los desempeños al realizar transferencias al español, mientras que los modelos restantes en los que se incluyen hablantes en español para el entrenamiento del modelo, se nota una mejora notable en la percepción de cada característica.

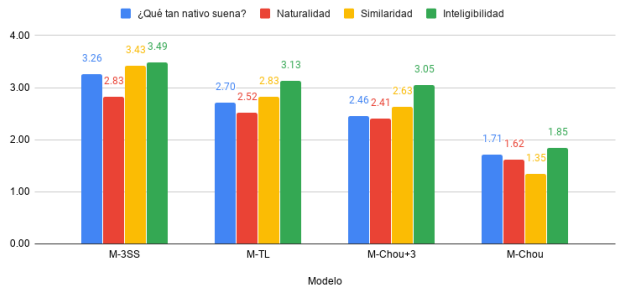



Figura 6: Desempeño por modelo

En los dos modelos en los que se intentó aprovechar las características extraídas del idioma inglés (M-TL y M-Chou+3) hubo una mejora significativa respecto al modelo M-Chou, principalmente en la similaridad del audio generado. Sin embargo, esta mejora no es suficiente para alcanzar al modelo M-3SS, que fue entrenado únicamente con hispanohablantes. Éste último obtuvo los mejores puntajes individuales en cada una de las características evaluadas, lo que sugiere que para realizar conversión de voz en un idioma, en este caso el español, es mejor entrenar directamente el modelo con audios en el idioma objetivo que intentar aprovechar las características acústicas de otro, pues las diferencias entre ambos idiomas pueden ser tan significativas que realmente no le aportan información al modelo para mejorar la transferencia de estilo.

Para analizar el desempeño según el hablante objetivo, se tomaron los audios generados por el modelo M-3SS, y de manera similar, se calcularon los promedios de cada característica evaluada. En la figura 7 se puede notar una diferencia en el desempeño de los tres hablantes, ligera entre el hablante Daniel y León, y un poco más marcada entre estos dos últimos y el hablante Jose.

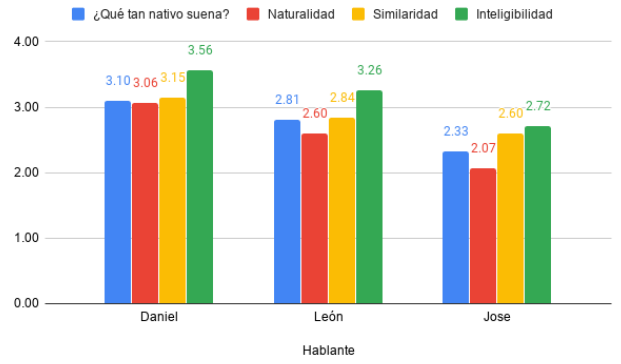
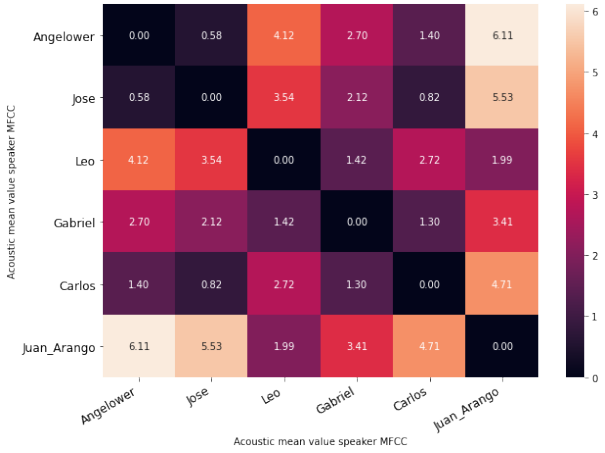
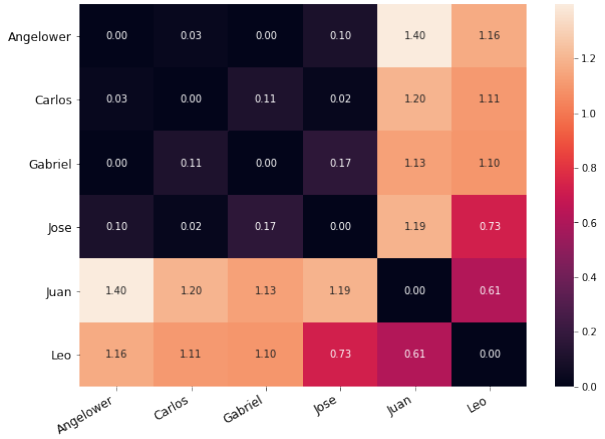



Figura 7: Desempeño por hablante en el modelo M-3SS

Como se discutió en la la sección III-B, la transferencia de estilo presentaba resultados diferentes en términos de naturalidad e inteligibilidad, dependiendo de la pareja de hablantes involucrados. Por lo tanto se decidió evaluar la similitud entre los hablantes como una estrategia para explicar si dicho




(a) Similitud basada en distancia euclidiana




(b) Similitud basada en verificación de hablante

Figura 8: Medidas de similitud entre hablantes

comportamiento se debía a la similitud entre los hablantes involucrados. Para esta evaluación se usaron audios grabados por nuevos hablantes: Angelower, Gabriel y Juan. Al realizar una transferencia de estilo de estos últimos a los hablantes Carlos Daniel, León y Jose, perceptualmente se notó un mejor desempeño en los transferidos al estilo del hablante León, por lo que se esperaba encontrar una baja distancia entre los tres hablantes nuevos y León, pues así al modelo se le dificultaría menos transferir el estilo entre hablantes con una voz similar que con una voz muy diferente. Sin embargo, al obtener las medidas de similitud entre hablantes, se observó un patrón diferente.

En la figura 8 se muestran los resultados de medir la similitud entre hablantes usando los dos enfoques diferentes, siendo uno la distancia euclidiana entre las características acústicas de hablantes que pronunciaron un mismo audio en la figura 8a y el otro calculando la Divergencia de Kullback-Leibler entre las distribuciones de probabilidad de cada hablante, en la figura 8b. Como se puede observar en la figura 8, al comparar al hablante Angelower con los hablantes Jose y Carlos Daniel, la medida de distancia es cercana a 0 en ambos enfoques lo

que indicaría que sus voces son similares, mientras que la medida obtenida al compararlo con el hablante León es mucho mayor. En el caso de Gabriel se observa en la figura 8a que la distancia frente a Carlos Daniel y León es similar, mientras que en la figura 8b se nota una diferencia significativa, esto puede indicar que las medidas exploradas pueden conducir a diferentes interpretaciones. Estos resultados responden de manera negativa a nuestra hipótesis y no nos permiten explicar la variación en el desempeño entre diferentes hablantes, pero deja abierta la posibilidad de plantear nuevas hipótesis, como que las medidas de similitud usadas no son adecuadas para describir el desempeño del modelo de transferencia de estilo y se deben explorar otro tipo de aproximaciones, o que el modelo no es tan eficaz al transferir el estilo entre hablantes con medidas de similitud bajas pues el discriminador deja pasar fragmentos del audio del hablante original al no ser capaz de diferenciarlo correctamente de la voz del hablante objetivo.

## VI. Discusión y conclusiones

La validación realizada a través de las encuestas permite concluir que de los modelos entrenados, el que mejor desempeño tuvo fue el modelo M-3SS, que fue entrenado únicamente con audios en español. Esto puede deberse a que la acústica, prosodia y articulación de los lenguajes son diferentes y al parecer el modelo debe ajustarse completamente para poder responder a un idioma nuevo, cosa que no se logra a través del transfer learning usando un modelo previamente entrenado con audios en otro idioma. Aún es necesario estudiar si hacer transfer learning con hablantes de un mismo idioma puede mejorar el entrenamiento del modelo cuando se desee usar un nuevo hablante objetivo.

Desde el punto de vista operativo, los experimentos fueron exitosos, pues se logró reproducir el entrenamiento propuesto por [7] y además se logró modificar el modelo para realizar transferencias de estilo en español, tanto agregando más hablantes al conjunto de datos como modificando las capas de la arquitectura para adaptarlas al número de hablantes disponibles. Estudiar el modelo seleccionado permitió un mayor entendimiento del funcionamiento y la implementación de arquitecturas de redes neuronales profundas y su aplicación en un campo como el procesamiento de audio usando una biblioteca como PyTorch. Un nuevo reto desde el punto de vista operativo puede ser modificar la implementación del modelo, de tal manera que las transferencias puedan ser realizadas en tiempo real, aunque es posible que esta aproximación no pueda ser expuesta como un servicio web, debido a la latencia que implica enviar los datos entre un cliente y un servidor remoto.

En la infraestructura de la plataforma web se integraron diferentes componentes usando diversas tecnologías. Cada componente jugó un papel importante en el transcurso del proyecto, en el que se usaron algunas tecnologías conocidas por los integrantes del equipo de trabajo, pero también se tuvo la oportunidad de explorar nuevas herramientas, como los servicios de GCP o AWS y el framework FastAPI del cual se destaca la facilidad con la cual permitió construir el servicio web.

La creación de la base de datos para el entrenamiento de los modelos en español representó una dificultad, pues no se logró garantizar que todos los hablantes tuvieran las mismas condiciones técnicas de grabación; además sólo se consiguió grabar a tres hablantes, todos hombres, por lo que no hay tanta variedad en los datos usados. Teniendo en cuenta que el desempeño del modelo puede depender en gran medida de la calidad de los audios usados para su entrenamiento, es por ello que se considera relevante la construcción de un corpus de audios grabados en mejores condiciones y que abarquen un gran espectro fonético del idioma español.

Por otro lado, es posible que existan medidas de similitud que sean más adecuadas para explicar el desempeño de las transferencias de estilo que las discutidas en la sección III-B, por lo que se pretende explorar medidas basadas en tecnologías más avanzadas de verificación o identificación de un hablante, como los i-vectors o los x-vectors. Éstos últimos están basados en Redes Profundas, por lo que pueden ser un complemento con gran potencial, además de que su estudio podría aportar elementos de mejora a la arquitectura de transferencia de estilo.

Finalmente, la infraestructura desplegada no es capaz de soportar una carga de trabajo de muchas peticiones concurrentes. Un reto a futuro es realizar una implementación escalable, que sea capaz de soportar cargas de trabajo de diferentes volúmenes.

## Referencias


[1] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2414–2423, 2016.

[2] J. Chorowski, R. J. Weiss, R. A. Saurous, and S. Bengio, "On using backpropagation for speech texture generation and voice conversion," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2256–2260, IEEE, 2018.

[3] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Advances in Neural Information Processing Systems*, pp. 10019–10029, 2018.

[4] Y. Gao, R. Singh, and B. Raj, "Voice impersonation using generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2506–2510, IEEE, 2018.

[5] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu, *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, pp. 4480–4490, 2018.

[6] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "Autovc: Zero-shot voice style transfer with only autoencoder loss," 2019.

[7] J.-c. Chou, C.-c. Yeh, H.-y. Lee, and L.-s. Lee, "Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations," *arXiv preprint arXiv:1804.02812*, 2018.

[8] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1, pp. 19 – 41, 2000.

[9] M. K. Singh, N. Singh, and A. K. Singh, "Speaker's voice characteristics and similarity measurement using euclidean distances," in *2019 International Conference on Signal Processing and Communication (ICSC)*, pp. 317–322, 2019.

[10] C. Veaux, J. Yamagishi, and K. MacDonald, *CSTR VCTK Corpus: English Multi-speaker Corpus for CSTR Voice Cloning Toolkit*. University of Edinburgh. The Centre for Speech Technology Research (CSTR), 2017.