



# **SINTETIZADOR DE TEXTO A VOZ PARA EL IDIOMA ESPAÑOL**

ALEJANDRO OROZCO HURTADO

201744439

orozco.alejandro@correounivalle.edu.co

Director

OSCAR FERNANDO BEDOYA LEIVA, PhD.

oscar.bedoya@correounivalle.edu.co

Facultad de Ingeniería

Escuela de Ingeniería de Sistemas y Computación

Programa Académico de Ingeniería de Sistemas

Julio de 2023

## RESUMEN

En nuestra cultura el lenguaje escrito se ha diferenciado del lenguaje oral que le dio su origen, se ha integrado de manera legítima en sociedad y se ha convertido en una forma fundamental de acceder a la información. En este contexto, los sintetizadores de texto a voz han permitido brindar acceso a la información. Estos avances tecnológicos han superado limitaciones para personas con dificultades de alfabetismo o problemas de visión, además de ofrecer nuevas posibilidades en el ámbito educativo y de entretenimiento.

Actualmente existe una serie de técnicas de inteligencia artificial, tales como el aprendizaje automático y profundo, que permiten el diseño e implementación de modelos sintetizadores de texto a voz, pero estos modelos suelen ser casi exclusivamente entrenados y evaluados con y para hablantes del inglés. Por lo tanto, el problema particular que se aborda en este trabajo de grado es implementar un modelo sintetizador de texto a voz, al que se puede acceder a través de internet, con una voz sintética que cuenta con la fonética y fonología del español en específico el dialecto colombiano.

En este trabajo se seleccionaron los modelos Tacotron 2 y HiFi-GAN, para implementar el sistema de síntesis de texto a voz. El modelo seleccionado fue entrenado cuatro veces de forma independiente, utilizando conjuntos de datos diferentes. Se evaluó la calidad de las voces sintetizadas y estas métricas proporcionaron una evaluación objetiva y cuantitativa del rendimiento del modelo. Finalmente, el modelo entrenado se implementó en Google Colab, lo que permite su acceso y uso de manera gratuita.

# CONTENIDO

1. Introducción	6
1.1. Planteamiento y formulación del problema	6
1.2. Justificación del problema	8
1.2.1. Justificación académica	8
1.2.2. Justificación social	8
1.2.3. Justificación económica	9
1.3. Objetivos	9
1.3.1. Objetivo general	9
1.3.2. Objetivos específicos	9
1.3.3. Resultados obtenidos	10
1.4. Alcances de la propuesta	12
2. Estado del arte	13
2.1. Antecedentes de la síntesis de texto a voz	13
2.2. Primeros modelos de síntesis de texto a voz	14
2.3. Aprendizaje profundo en la síntesis de texto a voz	15
2.4. Síntesis de texto a voz en el mercado actual	16
2.5. Síntesis de texto a voz en español con dialecto colombiano	18
3. Marco teórico	20
3.1. Aprendizaje automático	20
3.2. Aprendizaje profundo	21
3.3. Redes neuronales	22
3.4. Seq2Seq	23
3.5. Redes neuronales recurrentes	24
3.6. Red neuronal convolucional	25
3.7. Atención	26
3.8. Transformer	26
3.9. Espectrograma	27
3.9.1. Espectrograma de Mel	28
3.10. Modelo acústico	29

3.10.1. Tacotron 2	31
3.10.2. Deep Voice 3	32
3.10.3. FastSpeech 2	33
3.11. Vocoder	34
3.11.1. WaveNet	35
3.11.2. WaveRNN	36
3.11.3. WaveGlow	36
3.11.4. HiFi-GAN	37
3.12. Conjunto de datos	38
4. Modelo propuesto	40
4.1. Arquitectura del modelo propuesto	40
4.2. Selección del modelo acústico	41
4.3. Selección del vocoder	42
4.4. Selección del conjunto de datos	43
4.4.1. Creación del conjunto de datos	45
4.5. Entrenamiento del modelo propuesto	46
4.5.1. Voz 149	47
4.5.2. Voz 250	48
4.5.3. Voz 2534	48
4.5.4. Voz 250b	49
4.6. Modelo desplegado	50
5. Pruebas y resultados	52
5.1. Pruebas	52
5.1.1. Pruebas y observaciones iniciales	52
5.2. Métricas (MOS)	53
5.3. Resultados	54
5.4. Análisis	57
5.5. Comparación de resultados	58
6. Conclusiones	60
6.1. Trabajos futuros	61
7. Bibliografía	62

## INDICE DE FIGURAS

Figura 1. Reconstrucción de la máquina de habla de Wolfgang von Kempelen hecha por Wheatstone.	14
Figura 2. Google Cloud voces disponibles.	18
Figura 3. Amazon Polly voces disponibles.	19
Figura 4. Arquitectura de una red neuronal típica.	23
Figura 5. Sistema Seq2Seq basado en RNN.	24
Figura 6. RNN desenrollada.	25
Figura 7. Distribución de autoatención del codificador para la palabra "it".	27
Figura 8. Ilustración de la forma de onda de la señal, espectrograma, y espectrograma de Mel.	29
Figura 9. Pipeline de sintetizadores de texto a voz.	30
Figura 10. Arquitectura de Tacotron 2.	31
Figura 11. Arquitectura de Deep Voice 3.	32
Figura 12. Arquitectura de FastSpeech 2.	33
Figura 13. Stack de capas convolucionales causales dilatada	36
Figura 14. Arquitectura de WaveGlow.	37
Figura 15. Arquitectura del generador de HiFi-GAN.	38
Figura 16. Diagrama del modelo propuesto.	41
Figura 17. Cuaderno de Google Colab con los modelos entrenados.	51
Figura 18. Pregunta de opción múltiple con escala tipo MOS.	54
Figura 19. Pregunta comparativa.	56
Figura 20. Comparación entre las dos voces generadas.	56
Figura 21. Gráfico de dispersión para Voz 149.	57
Figura 22. Gráfico de dispersión para Voz 250.	58

## INDICE DE TABLAS

Tabla 1. Resultados obtenidos.	10
Tabla 2. Comparativa de diferentes modelos acústicos de texto a voz.	30
Tabla 3. Comparativa de diferentes vocoder para texto a voz.	34
Tabla 4. Comparación de modelos acústicos.	42
Tabla 5. Comparación de vocoders.	43
Tabla 6. Diferentes data sets usados para texto a voz.	43
Tabla 7. MOS con 95% de nivel de confianza.	55

# **CAPÍTULO 1**

## **INTRODUCCIÓN**

En este capítulo se presenta el problema a abordar en este trabajo de grado junto con su pregunta de investigación, así como los objetivos, la justificación y el alcance.

### **1.1. Planteamiento y formulación del problema**

Nuestra cultura actual ha posibilitado un desarrollo tecnológico tal que los límites entre diferentes ámbitos de la vida tienden a diluirse. La oralidad es el primer logro lingüístico de la humanidad, pero le es consustancial a su condición natural y debieron pasar muchos milenios para que aparecieran los textos escritos como un logro tecnológico que diferenció la escritura de la oralidad. Esta diferenciación fue de tal grado que hoy en día clasificamos históricamente las culturas o sociedades sin escritura de las que sí la desarrollaron, y se considera a estas últimas como las más desarrolladas [1]. A su vez, se considera que el lenguaje escrito es diferenciable del oral en una misma cultura y en la nuestra se le da el predominio a la escritura de diversas maneras.

Esta distinción lingüística permitió incluso el desarrollo de sistemas de poder y acumulación basados en la escritura, y marginó o segregó de dicho poder a los que tenían poco o nada de uso de la escritura, por falta de entrenamiento o por deficiencias y daños visuales. En nuestro desarrollo tecnológico y social presente, esta diferenciación afecta sobre todo a invidentes, es decir, personas que han sufrido ceguera por causas congénitas o por lesiones adquiridas como las cataratas; pero ya es posible resolver esta exclusión y diluir los límites de estos dos ámbitos lingüísticos para el beneficio de estos seres humanos que no acceden a visualizar la escritura de manera parcial o total. Para ello se han desarrollado sintetizadores de texto a voz para convertir textos escritos a voz, lo cual permite

que una buena parte de la humanidad segregada y excluida de las adquisiciones de la escritura, pueda acceder a ella a través de la oralidad de una voz.

Actualmente existe una serie de técnicas de inteligencia artificial, tales como el aprendizaje automático y profundo, que permiten el diseño e implementación de modelos sintetizadores texto a voz. Desde 2016 con la publicación del modelo de WaveNet [2] y por la calidad que presentaba dicho modelo, el paradigma de los sintetizadores de texto a voz pasó de modelos basados en técnicas como la síntesis por concatenación o la síntesis por reglas fonéticas a modelos de aprendizaje profundo con redes neuronales. En la línea de aprendizaje profundo le siguieron modelos como Tacotron [3], que toman directamente secuencias de letras para generar características acústicas, o modelos como Deep Voice [4], Char2Wav [5] y Clarinet [6], los cuales como modelos end-to-end no necesitan de un modelo complementario para generar audio a partir de texto. La propuesta de usar técnicas de aprendizaje profundo en la síntesis de texto a voz sigue siendo el paradigma actual y modelos posteriores como Tacotron 2 [7], Deep Voice 3 [8], GlowTTS [9] y FastSpeech 2 [10] hacen uso de dichas técnicas.

Hoy en día, se siguen proponiendo nuevos modelos de síntesis de texto a voz aplicando diferentes técnicas de aprendizaje automático y profundo, o perfeccionando las ya existentes, pero estos modelos suelen ser casi exclusivamente entrenados y evaluados con y para hablantes del inglés. Por lo tanto, el problema particular que se aborda en este trabajo de grado es implementar un modelo sintetizador de texto a voz para el uso generalizado al que se puede acceder a través de internet de manera gratuita, para lograr transcribir textos escritos a una voz sintética que cuenta con la fonética y fonología del español. Lo anterior, se hace para facilitar la lectura de textos a personas de habla hispana con problemas para leer, como las afectadas por impedimentos visuales totales, pero también serviría para personas con diversos grados de deficiencias visuales o de lectura, que prefieren evitar forzar la vista o la atención, leyendo digitalmente, como por ejemplo los adultos mayores. Además, el modelo



propuesto puede ser útil para ávidos oyentes de audiolibros e incluso puede servir a estudiantes del idioma español en la práctica de la pronunciación o para ayudar a comunicarse entre personas con impedimentos del habla.

## **1.2. Justificación del problema**

### **1.2.1. Justificación académica**

En los últimos años, la inteligencia artificial se ha convertido en un tema cada vez más popular y relevante. Con el surgimiento de tecnologías como los deepfakes y la presentación de modelos de inteligencia artificial como ChatGPT, DALL-E 2, AlphaFold 2 y otras I.A. similares, hay un renovado interés y atención en este campo. Por tanto, se ha decidido explorar uno de los campos de aplicación de la inteligencia artificial, específicamente la síntesis de texto a voz, enfocándose en el español. Esta decisión se basa en el reconocimiento de la importancia de desarrollar herramientas y tecnologías que se adapten a las necesidades de las personas hispanohablantes, promoviendo así la inclusión y el acceso equitativo a la información y la comunicación en su idioma. Además, desde un punto de vista lingüístico, un sintetizador de voz en español ofrece una oportunidad para analizar aspectos fonéticos, fonológicos, y prosódicos del español hablado.

### **1.2.2. Justificación social**

Las personas de lengua materna en español con limitaciones visuales parciales severas o totales, presentan grandes limitaciones para el acceso a materiales educativos, informativos, y de entretenimiento en su idioma, debido en primer lugar a que estos textos no suelen estar en lenguaje braille que podría ser accesible a ellos teniendo la educación necesaria o en segundo lugar a que los sintetizadores de texto a voz en su mayoría son de aplicación al inglés. Lo anterior es una forma de exclusión y segregación social que, a nuestra sociedad en lengua hispana y en especial hispanoamericana, no suele preocupar, ya que este tipo de

segregaciones son más del orden material simbólico que comportamental simbólico. Sin embargo, terminará teniendo un gran impacto social y económico de marginalidad para las personas con este tipo de limitaciones.

### **1.2.3. Justificación económica**

En general, existe una menor posibilidad de acceso a bienes económicos para grupos sociales de personas con limitaciones visuales que a su vez implican mayores costos de integración por la falta de materiales no accesibles. Lo anterior se debe principalmente a que acarrear el pago de personas que les lean los textos o los instruyan al respecto y porque tampoco tendrán posibilidad de acceso a empleos que funcionen bajo el criterio del mérito, ya que éste supone igualdad de condiciones de acceso a materiales para su formación. Lo anterior implica desventajas sociales para obtener ingresos necesarios para la autonomía social.

## **1.3. Objetivos**

### **1.3.1. Objetivo general**

Implementar un modelo sintetizador de texto a voz haciendo uso de técnicas de inteligencia artificial, tales como aprendizaje automático y en especial el aprendizaje profundo, para el diseño del modelo y el correspondiente entrenamiento.

### **1.3.2. Objetivos específicos**

- Revisar los diferentes modelos de aprendizaje profundo de síntesis de texto a voz para el español.
- Identificar el conjunto de datos en español a usar en el entrenamiento de los modelos.

- Entrenar un modelo de aprendizaje profundo para la síntesis de texto a voz sobre el conjunto de datos seleccionados.
- Usar métricas de desempeño para determinar la precisión del modelo.
- Desplegar en la web el modelo ya entrenado para su uso de manera pública.

### 1.3.3. Resultados obtenidos

Objetivo específico	Resultado
Revisar los diferentes modelos de aprendizaje profundo de síntesis de texto a voz para el español.	Se seleccionaron los modelos Tacotron 2 y HiFi-GAN, para la implementación del sistema sintetizador de texto a voz. Tacotron 2 se utilizó como modelo acústico para generar las representaciones de espectrogramas de los textos de entrada, mientras que HiFi-GAN se utilizó como vocoder para sintetizar los espectrogramas en señales de audio de alta calidad.  Ver <a href="#">Capítulo 4</a> .
Identificar el conjunto de datos en español a usar en el entrenamiento de los modelos.	En el proceso de entrenamiento del modelo de síntesis de texto a voz, se utilizaron dos conjuntos de datos. Un conjunto de dato fue obtenido de internet y se trata del conjunto de datos: ChqCSsdt, y de este, una porción se usó como otro sub conjunto de datos; el otro conjunto de datos fue creado para el propósito específico de este proyecto. Los dos

	<p>conjuntos de datos contienen grabaciones de habla en español con dialecto colombiano, junto con una lista de transcripciones. Estos conjuntos de datos fueron seleccionados porque el dialecto colombiano es el acento de interés para este proyecto.</p> <p>Ver <a href="#">Sección 4.4</a>.</p>
Entrenar un modelo de aprendizaje profundo para la síntesis de texto a voz, sobre el conjunto de datos seleccionado.	<p>Se entrenó el modelo previamente seleccionado cuatro veces de forma independiente. Cada entrenamiento se realizó con un conjunto de datos diferente (en un caso con una configuración diferente).</p> <p>Ver <a href="#">Sección 4.5</a>.</p>
Usar métricas de desempeño para determinar la precisión del modelo.	<p>Se llevó a cabo una encuesta con 55 participantes, a cada uno se le solicitó calificar la calidad de voz según las métricas MOS (Mean Opinion Score) para 10 audios de voz sintetizada. Esto se realizó para dos instancias del modelo que habían sido previamente entrenadas con diferentes conjuntos de datos. Además, se les pidió a los participantes de la encuesta que, a partir de cinco frases diferentes, eligieran cuál de los dos audios</p>

	<p>sintetizados les resultaba más entendible según la frase correspondiente.</p> <p>Ver <a href="#">Sección 5.3</a>.</p>
<p>Desplegar en la web el modelo ya entrenado para su uso de manera pública.</p>	<p>El modelo fue desplegado en la web a través de Google Colab, ofrece la opción de elegir entre las cuatro voces previamente entrenadas para sintetizar el texto ingresado a voz, ofreciendo algo de flexibilidad al usuario para seleccionar la voz que mejor se adapte a sus preferencias.</p> <p>Ver <a href="#">Sección 4.6</a>.</p>

**Tabla 1.** Resultados obtenidos.

#### 1.4. Alcances de la propuesta

Este trabajo pretende implementar un sintetizador de texto a voz que sintetice un audio de voz partiendo de un texto escrito. Estos modelos están basados en aprendizaje profundo y son entrenados con un conjunto de datos en español. El tipo de modelo texto a voz que se trabajará será de dos modelos, un modelo texto a espectrograma conocido como modelo principal y un modelo espectrograma a forma de onda, es decir audio, a éste se le conoce como vocoder. No se trabajarán otros tipos de modelos texto a voz como los end-to-end o modelos que no sigan el flujo anteriormente mencionado. Tampoco se hará profundidad en las características lingüísticas y fonéticas del español.

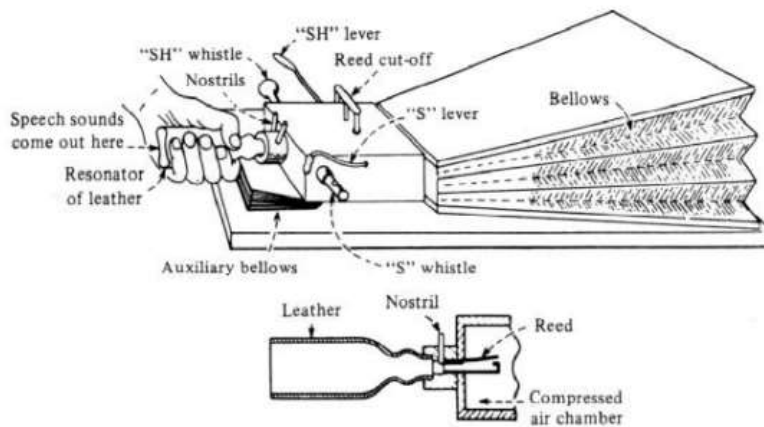
## **CAPÍTULO 2**

### **ESTADO DEL ARTE**

En este capítulo se presentan los antecedentes relevantes en el campo de la síntesis de texto a voz. Se abordan los desarrollos históricos y las investigaciones previas que sientan las bases para el desarrollo de sistemas de síntesis de texto a voz. También se muestran los enfoques tradicionales utilizados para generar voz artificial, como las síntesis basadas en un modelo articular o concatenación de unidades de sonido. Se exploran los avances recientes en el uso de técnicas de inteligencia artificial, como el aprendizaje profundo y las redes neuronales, para mejorar la calidad y la naturalidad de la síntesis de texto a voz. Se examina también el panorama actual de los sintetizadores de texto a voz. Por último, se aborda el estado de la síntesis de texto a voz en español con dialecto colombiano.

#### **2.1. Antecedentes de la síntesis de texto a voz**

A lo largo de la historia, se han realizado diversos intentos y métodos para lograr la síntesis de voz. En 1779, Christian Kratzenstein diseñó una máquina capaz de producir las cinco vocales largas, paralelamente Wolfgang von Kempelen trabajó y diseñó su propia máquina de síntesis de voz. En el siglo XIX, Charles Wheatstone construyó una versión mejorada de la máquina de von Kempelen, el modelo de Wheatstone que se muestra en Figura 1 era capaz de producir tanto vocales como la mayoría de las consonantes, e incluso algunas palabras. Alexander Graham Bell también realizó estudios en la síntesis de voz, pero al igual que sus predecesores los enfoques mecánicos para la síntesis de voz, no mostraron resultados destacables hasta la llegada de las computadoras [11].



**Figura 1.** Reconstrucción de la máquina de habla de Wolfgang von Kempelen hecha por Wheatstone [12].

Se considera que Homer Dudley fue el desarrollador del primer dispositivo sintetizador de voz conocido como VODER (Voice Operating Demonstrator), el cual se basó en el dispositivo VOCODER (Voice Coder), desarrollado en los años 30 en los laboratorios Bell [13]. El VOCODER original era un dispositivo diseñado para analizar parámetros acústicos de la voz en una variación lenta, y posteriormente el sintetizador reconstruiría una aproximación de la señal de voz original. El VODER consistía en una barra de muñeca que permitía seleccionar una fuente de voz o ruido, y un pedal para controlar la frecuencia fundamental. La calidad del habla y la inteligibilidad estaban lejos de ser buenas, pero el potencial para producir habla artificial quedó demostrado [11].

## 2.2. Primeros modelos de síntesis de texto a voz

Fue con la introducción de las computadoras y los avances en la tecnología cuando se lograron avances significativos en la síntesis de texto a voz. Con la capacidad de procesar y manipular datos de manera más compleja, se pudieron desarrollar algoritmos y modelos más sofisticados para generar voz artificial. Esto dio paso a enfoques basados en técnicas como la síntesis por concatenación, la síntesis por reglas fonéticas y, más recientemente, las redes neuronales y el

aprendizaje profundo. La primera demostración de un sistema de texto a voz completo se puede rastrear hasta el año 1968, Noriko Umeda del Laboratorio Eléctrico de Japón, desarrolló y demostró un sistema de texto a voz completo para inglés basado en un modelo articular [14].

En 1979, el Instituto de Tecnología de Massachusetts (M.I.T.) presentó el sistema de texto a voz MITalk. Este sistema fue utilizado posteriormente por Telesensory Systems Inc. y se convirtió en uno de los primeros casos de uso comercial de sintetizadores de texto a voz. Dos años más tarde, Dennis Klatt introdujo Klattalk. Las tecnologías usadas en este sistema de síntesis de voz junto con las usadas en MITalk, sentaron las bases para muchos otros sistemas de síntesis de texto a voz que surgieron posteriormente como DECtalk y Prose-2000 [15].

### **2.3. Aprendizaje profundo en la síntesis de texto a voz**

Anteriores modelos de síntesis de texto a voz como los usados antiguamente por el asistente de voz de Apple, Siri, eran sistemas basados en concatenación de unidades, los cuales requerían grandes bibliotecas de fragmentos del habla que al momento de realizar la síntesis de texto a voz divide el discurso grabado en sus componentes elementales, como los difonos y luego los concatena de acuerdo con el texto de entrada para crear un discurso nuevo [16].

Sin embargo, en los últimos años, la síntesis de texto a voz ha experimentado una evolución significativa. Los modelos basados en técnicas tradicionales como los modelos ocultos de Markov (HMM) o modelos Gaussianos mixtos (GMM), han sido reemplazados por enfoques basados en el aprendizaje profundo. Estos nuevos modelos han demostrado una calidad superior en la síntesis de voz, lo que ha llevado a la pérdida de relevancia de los modelos anteriores [17].



En 2016 se diseñó el primer modelo de un sintetizador de texto a voz en el que se introdujeron técnicas de desarrollo de síntesis de texto a voz basados en aprendizaje profundo. Dicho modelo se llamó WaveNet y era capaz de modular directamente ondas de audio a partir de características lingüísticas en lugar de concatenar unidades de fragmentos de sonidos grabados [2], [18]. En 2017 le siguieron modelos como Tacotron [3] y Deep Voice [4], los cuales propusieron tomar directamente secuencias de letras y fonemas, para generar características acústicas como espectrogramas y modelos como Char2Wav [5] y Clarinet [6]. Los anteriores son modelos end-to-end, es decir, que pueden generar audio a partir de un texto sin necesidad de un modelo complementario. Desde entonces han surgido variantes o versiones mejoradas como Tacotron 2 [7], Deep Voice 2 [19] y 3 [8], o modelos nuevos como GlowTTS [9] y FastSpeech 1 y 2 [10].

Desde entonces, los modelos de redes neuronales basadas en aprendizaje profundo han sido el estándar de la industria de síntesis de texto a voz. Nuevos estudios han buscado asimilarse cada vez más a la voz humana natural, realizando avances en aspectos de los que previamente carecían estos sistemas, como en la síntesis de voz emocional y expresiva, en la que se busca generar voces que reflejen diferentes emociones como la felicidad, la tristeza, y el miedo, entre otras.

## **2.4. Síntesis de texto a voz en el mercado actual**

En el mercado actual, la síntesis de texto a voz ha experimentado avances significativos y se ha convertido en una tecnología ampliamente adoptada en diversas aplicaciones y servicios. La calidad y la naturalidad de la voz sintetizada han mejorado considerablemente, lo que ha llevado a una experiencia de usuario más inmersiva y realista.

En términos de disponibilidad, existen numerosas soluciones de síntesis de texto a voz en el mercado, que son utilizadas en una variedad de aplicaciones. Por

ejemplo, los asistentes virtuales como Siri, Google Assistant y Amazon Alexa, utilizan la síntesis de voz para proporcionar respuestas habladas a las consultas de los usuarios. Además, servicios como Apple VoiceOver [20], ofrecen accesibilidad para personas con discapacidades visuales, como lectores de pantalla y aplicaciones de lectura de texto, hacen uso de la síntesis de voz para convertir texto en voz, permitiendo que los usuarios accedan a la información de manera auditiva.

En aplicaciones educativas como Duolingo [21], la síntesis de texto voz es utilizada en el aprendizaje para proporcionar retroalimentación auditiva a los estudiantes. Además, se emplea en herramientas de traducción de texto a voz como Traductor de Google, facilitando la comprensión de diferentes idiomas. En el campo del entretenimiento, el uso de la inteligencia artificial (IA) sigue siendo objeto de debate. Algunas aplicaciones, como Audible de Amazon [22], han optado por no permitir audiolibros leídos por IA, mientras que otras plataformas, como Apple Books [23], Google Play [24] y Spotify [25], han integrado esta tecnología en sus audiolibros o podcast.

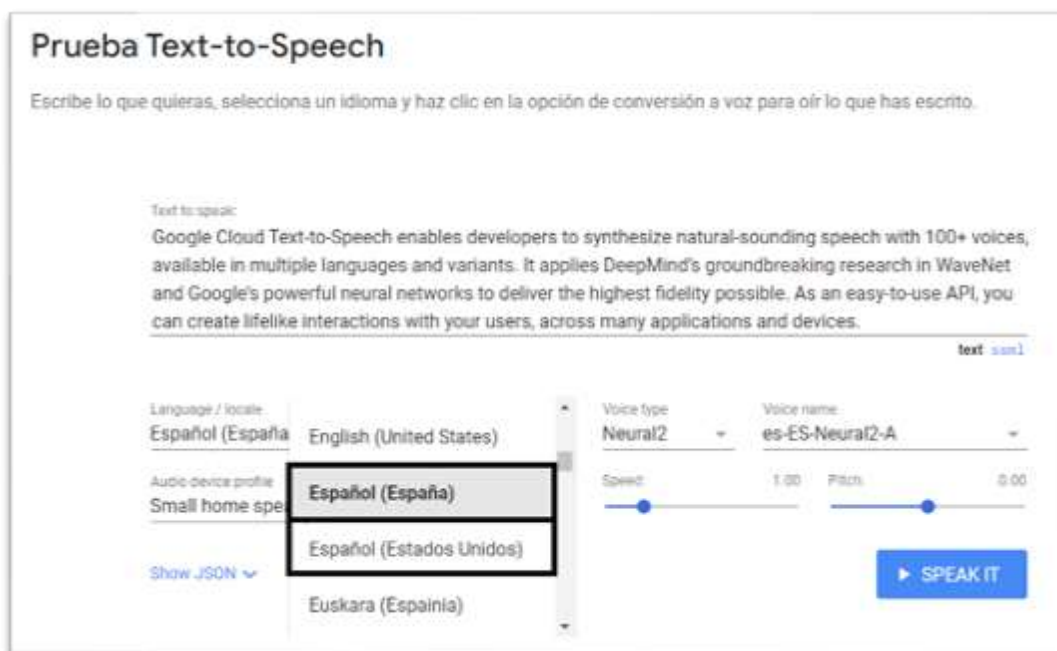
Existen varias aplicaciones que ofrecen servicios de texto a voz, brindando a los usuarios una amplia gama de opciones para generar voces sintetizadas. Algunas de estas aplicaciones, como [DeepZen](#), [Scribe Audio](#) y [Speechki](#), ofrecen galerías de voces pre-entrenadas. Además, algunas ofrecen la posibilidad de clonar la voz propia. Sin embargo, estos servicios suelen estar bajo modelos de pago, lo que puede limitar su accesibilidad para algunos usuarios.

Por otro lado, existen aplicaciones como [FakeYou](#) que, aunque también ofrecen servicios de pago, proporcionan opciones gratuitas algo limitadas. FakeYou cuenta con una comunidad de aficionados interesados en la inteligencia artificial y la clonación de voces, lo que les permite ofrecer galerías extensas de voces de personajes famosos. Es posible encontrar voces de personajes icónicos como Homero Simpson, El Chavo o incluso Barack Obama, lo que agrega un elemento

divertido y de entretenimiento a la síntesis de voz. Por otro lado, existen aplicaciones como [Coqui](#) que, aunque también requieren de un pago para acceder a ciertos servicios, ofrecen una interesante alternativa para aquellos usuarios con los conocimientos necesarios. Esta aplicación proporciona un [repositorio público en GitHub](#), lo que significa que aquellos que posean los conocimientos técnicos pueden utilizar servicios como la clonación de voz de forma gratuita. Esta iniciativa de código abierto brinda la posibilidad de experimentar y explorar las capacidades de clonación de voz sin incurrir en costos adicionales.

## 2.5. Síntesis de texto a voz en español con dialecto colombiano

Con la creciente popularidad de las inteligencias artificiales, las IA sintetizadoras de texto a voz están evolucionando rápidamente para adaptarse a diferentes idiomas y dialectos. Muchos modelos de texto a voz, como los ofrecidos por Amazon Polly [26] y Google Cloud [27], han sido adaptados para funcionar en varios idiomas, incluyendo el español. Sin embargo, es importante destacar que tanto Google Cloud como Amazon Polly no incluyen una versión con dialecto colombiano específico.



**Figura 2.** Google Cloud voces disponibles [28].

En la Figura 2 se puede observar que Google Cloud, ofrece las opciones de español ibérico y estadounidense, mientras que Amazon Polly, como se muestra en la Figura 3, proporciona sus propias versiones de los anteriores más el español mexicano.

Russian	ru-RU	Tatyana	Female
		Maxim	Male
Spanish (European)	es-ES	Conchita	Female
		Lucia	Female
		Enrique	Male
		Sergio	Male
Spanish (Mexican)	es-MX	Mia	Female
		Andrés	Male
Spanish (US)	es-US	Lupe**	Female
		Penélope/Penelope	Female
		Miguel	Male
		Pedro	Male
Swedish	sv-SE	Astrid	Female

**Figura 3.** Amazon Polly voces disponibles [26].

Además, otra compañía que ha expandido sus servicios de sintetizadores de texto a voz para llegar a una amplia audiencia es Meta. Siguiendo su propuesta *No Language Left Behind* (NLLB), recientemente lanzaron Massively Multilingual Speech (MMS) [29]. Esta iniciativa demostró el compromiso de Meta por abarcar una amplia diversidad lingüística y cultural, brindando posibilidades de preservar y difundir los conocimientos de diferentes comunidades. Este servicio tuvo en cuenta más de 25 dialectos colombianos, aunque no específicamente del español, sino de lenguas indígenas de América [30], [31].

## **CAPÍTULO 3**

### **MARCO TEÓRICO**

En este capítulo se presentan los conceptos fundamentales relacionados con la síntesis de texto a voz utilizando técnicas de aprendizaje automático y profundo. Además, se explican en detalle las redes neuronales y diferentes arquitecturas que se utilizan en la síntesis de texto a voz. También se abordan conceptos como espectrogramas, los cuales son herramientas fundamentales en el análisis del sonido, y se describen los modelos acústicos y vocoders, que son los bloques de construcción principales de los sistemas de síntesis de texto a voz. Con la finalidad de que el lector tenga una comprensión completa de la tecnología detrás de la síntesis de texto a voz, se detallan los principios fundamentales detrás de cada uno de estos conceptos y cómo se aplican en los sistemas de síntesis de texto a voz.

#### **3.1. Aprendizaje automático**

El aprendizaje automático (o *machine learning*) es un campo de investigación dedicado a comprender y crear métodos que "aprendan", es decir, métodos que aprovechan los datos para mejorar el rendimiento en algún conjunto de tareas [32]. A diferencia de algoritmos convencionales que siguen instrucciones programadas, los algoritmos de aprendizaje automático identifican patrones y relaciones en los datos a través del uso de técnicas estadísticas, para luego generalizar este conocimiento para tomar decisiones o hacer predicciones sobre nuevos datos [33]. Al entrenar (alimentar) un algoritmo de aprendizaje automático con un conjunto de datos, se obtiene un modelo que puede hacer predicciones o decisiones sobre nuevas observaciones [34].

El proceso de aprendizaje automático suele implicar los siguientes pasos: recopilación de datos relevantes, depuración de los datos de entrenamiento, elección del algoritmo apropiado, entrenamiento del modelo utilizando los datos de

entrenamiento, evaluación del rendimiento del modelo y, finalmente, hacer uso del modelo entrenado para hacer predicciones o decisiones en nuevas situaciones. El aprendizaje automático ha demostrado su utilidad en aplicaciones, como clasificación de imágenes, recomendación de productos, detección de fraudes, traducción automática y muchas otras áreas donde es necesario el procesamiento y análisis de grandes cantidades de datos.

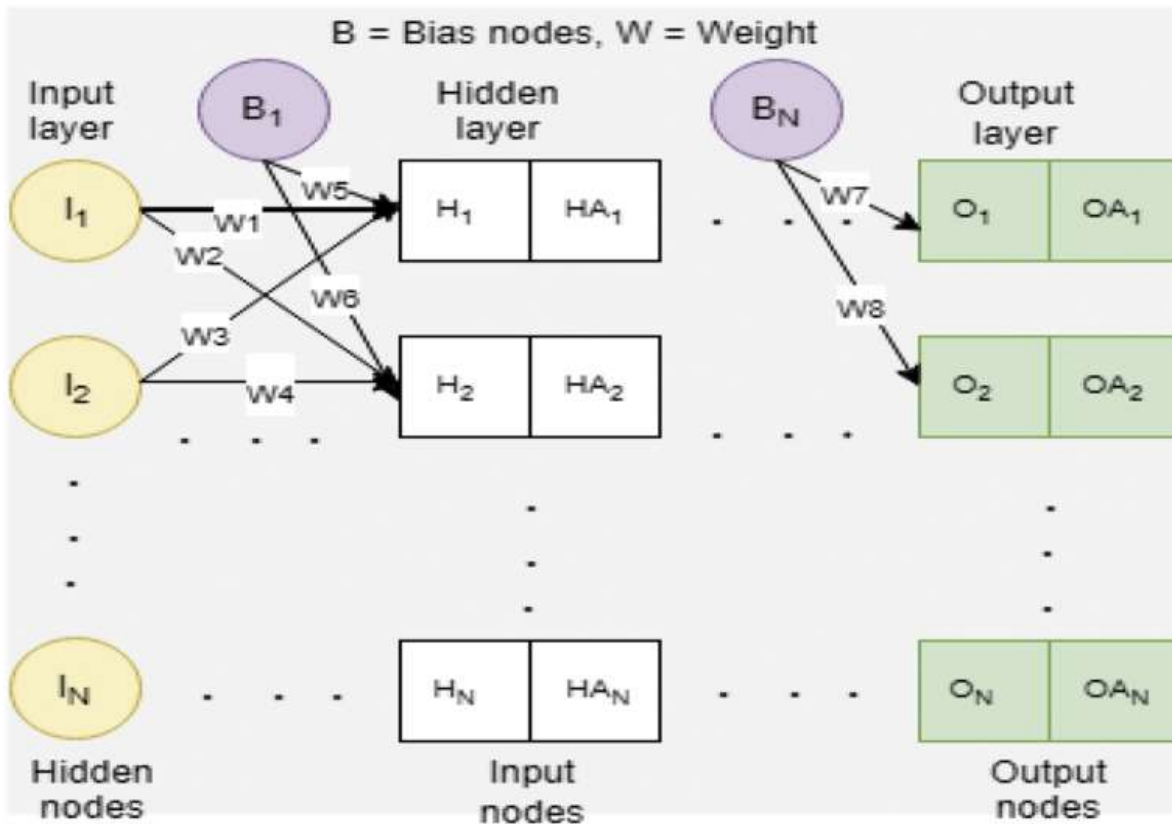
### **3.2. Aprendizaje profundo**

El aprendizaje profundo (o *deep learning*) es un conjunto de algoritmos de aprendizaje automático que están formados por la composición de múltiples transformaciones no lineales, con el objetivo de producir representaciones más abstractas y, en última instancia, más útiles [35]. El aprendizaje profundo descubre estructuras intrincadas en grandes conjuntos de datos mediante el uso del algoritmo de retro propagación para indicar cómo una máquina debe cambiar sus parámetros internos, lo cuales se utilizan para calcular la representación en cada capa a partir de la representación en la capa anterior [36]. A diferencia de los enfoques de aprendizaje automático tradicionales, que se basan en datos estructurados con características específicas diseñadas por expertos, el aprendizaje profundo permite que los modelos aprendan automáticamente representaciones de los datos a través de la extracción de características en capas sucesivas. Esto significa que los modelos de aprendizaje profundo pueden aprender directamente de datos no estructurados como texto e imágenes, y automatizan la extracción de características, eliminando parte de la dependencia de expertos humanos. A través de procesos como pendiente de gradiente o propagación inversa, los algoritmos de aprendizaje profundo se ajustan y se adaptan a sí mismos para ganar precisión, lo que le permite realizar mejores predicciones sobre nuevos datos [37].

A través de técnicas como las redes neuronales convolucionales y las redes neuronales recurrentes, el aprendizaje profundo ha demostrado ser especialmente efectivo en tareas relacionadas con el procesamiento de datos no estructurados, como el reconocimiento de imágenes, detección de objetos, el procesamiento del lenguaje natural, síntesis de texto a voz, descubrimiento de fármacos y en la genómica.

### **3.3. Redes neuronales**

Las redes neuronales son un modelo computacional de aprendizaje automático en el que una computadora aprende a realizar alguna tarea analizando ejemplos de entrenamiento. Estos ejemplos para el entrenamiento suelen ser depurados a mano con anticipación. Basadas en las neuronas humanas, una red neuronal consta de miles o incluso millones de nodos de procesamiento simples que están densamente interconectados. En la Figura 4 se muestra la arquitectura de la mayoría de las redes neuronales actuales, las cuales están organizadas en capas de nodos y los datos se suelen mover a través de ellas en una sola dirección. Un nodo individual puede estar conectado a varios nodos en la capa anterior, de la que recibe datos y varios nodos en la capa posterior, a la que envía datos. A cada una de sus conexiones entrantes, un nodo le asignará un peso. Cuando la red está activa, el nodo recibe un elemento de datos diferente por cada una de sus conexiones y lo multiplica por el peso asociado. Luego, suma los productos resultantes, obteniendo un solo valor que tendrá que superar un umbral para que el valor del nodo se propague a las capas conectadas [38].



**Figura 4.** Arquitectura de una red neuronal típica [39].

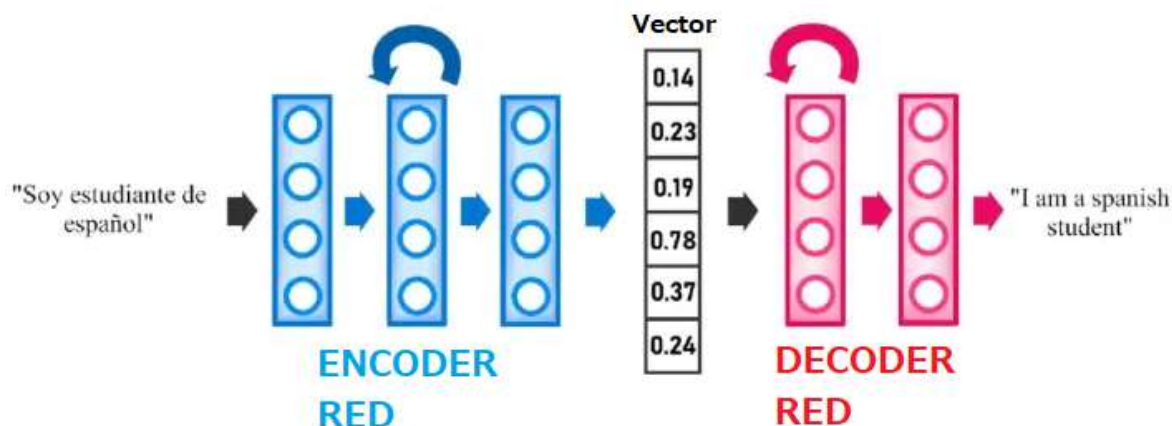
### 3.4. Seq2Seq

Sequence-to-Sequence es una arquitectura de redes neuronales que permite la generación de secuencias de salida a partir de secuencias de entrada [40]. Es un tipo de arquitectura de redes neuronales que se utiliza comúnmente para tareas de PLN (Procesamiento de Lenguaje Natural), como la traducción automática o la generación de resumen automáticamente.

La arquitectura consiste en dos componentes principales: el encoder y el decoder. El encoder se encarga de procesar la entrada y genera un vector en estado oculto que representa la información relevante de la entrada. Luego, el decoder utiliza ese vector oculto para generar la secuencia de salida apropiada. En la Figura 5 se muestra un ejemplo de un sistema Seq2Seq consistente de dos RNNs, una de



encoder que transforma la frase en español a un vector, y otra de decoder, que transforma el vector a una cadena de texto en inglés.

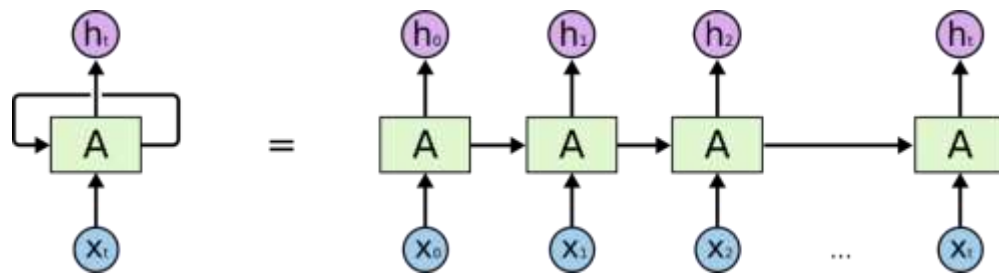


**Figura 5.** Sistema Seq2Seq basado en RNN [41].

### 3.5. Redes neuronales recurrentes

Las RNN (del inglés *Recurrent Neural Network*) son un tipo de redes neuronales artificiales que utilizan datos de series temporales o datos que involucran secuencias. La característica distintiva de una RNN es la existencia de conexiones recurrentes entre las neuronas, que permiten que la salida de una neurona se retroalimente como entrada en pasos posteriores. Esta propiedad de retroalimentación le permite a las RNN tener el concepto de “memoria” que les ayuda a almacenar los estados o la información de las entradas anteriores para aprender a reconocer patrones y dependencias en secuencias de datos y generar la siguiente salida de la secuencia [42].

Una RNN se puede pensar como múltiples copias de la misma red neuronal, en la Figura 6, el nodo A es un bucle de la red neuronal que analiza una entrada  $X_t$  y genera un valor  $h_t$ . El bucle A permite pasar información de un paso de la red al siguiente. En la figura el lado derecho de la ecuación representa la RNN del lado izquierdo vista como múltiples copias de la misma red neuronal A.



**Figura 6.** RNN desenrollada [43].

Las RNN se han utilizado con éxito en una amplia gama de áreas que involucran datos secuenciales. En el área de PLN en especial las RNN son usadas para modelos de reconocimiento del habla, traducción automática, generación de texto, análisis de sentimientos y síntesis de texto a voz.

### 3.6. Red neuronal convolucional

Una red neuronal convolucional o CNN (Convolutional Neural Network) es una arquitectura de redes profundas que se caracterizan estar compuesta por capas convolucionales, capas de pooling y capas completamente conectadas. Las capas de convolución [44] aplican filtros a la entrada de datos, extrayendo características importantes. Luego son seguidas por las capas de agrupamiento o pooling, que reducen la dimensión espacial de los datos disminuyendo la cantidad de parámetros y la complejidad computacional del modelo para finalmente pasar por las capas completamente conectadas que realizan clasificación o predicciones para producir la salida deseada.

La arquitectura CNN es utilizada principalmente en procesamiento y análisis de datos de tipo espacial, como imágenes o señales, por lo que son altamente efectivas en tareas de visión artificial como el reconocimiento de objetos, la detección de rostros, la segmentación de imágenes y la clasificación de imágenes. Aunque también ha tenido un papel en otros campos como en PLN.

### 3.7. Atención

En las redes neuronales, la atención es una técnica que asigna un peso o "prestar atención" a los estados específicos en el pasado que son más relevantes para producir el siguiente elemento en la secuencia de salida. Aprender qué parte de los datos es más importante que otra, depende del contexto, pero este proceso es diferenciable, por lo que el proceso de "prestar atención" se puede aprender durante el entrenamiento [45].

### 3.8. Transformer

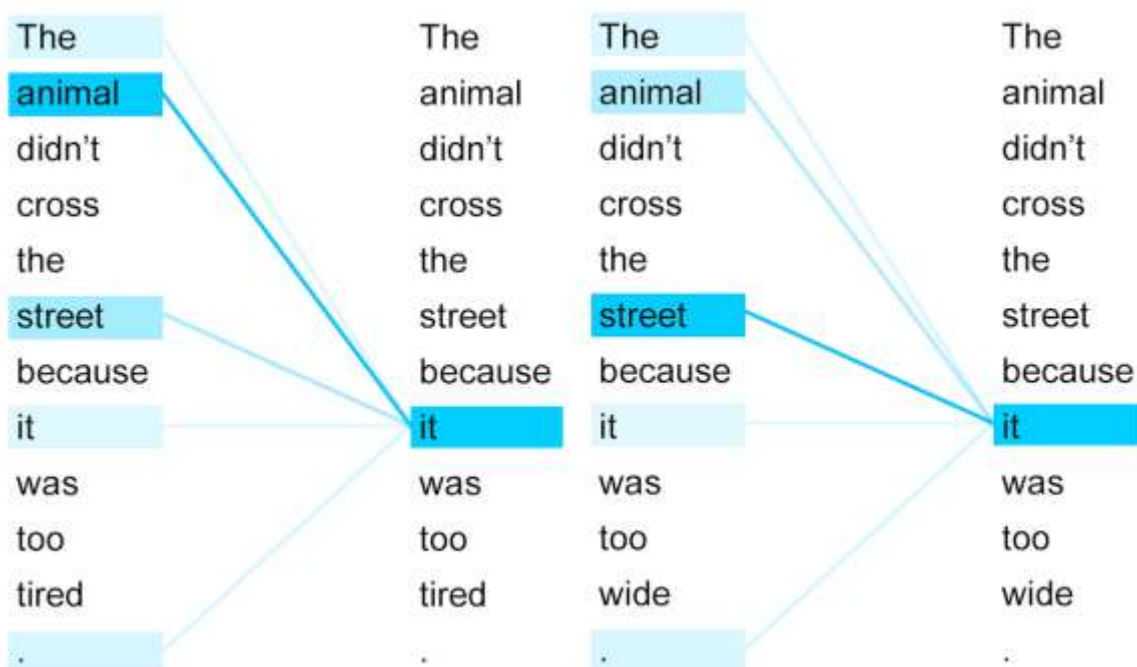
En el área de PLN el transformador es un modelo de aprendizaje profundo basado únicamente en mecanismos de atención, prescindiendo por completo de la recurrencia y la convolución, reemplazando las capas recurrentes más comúnmente utilizadas en arquitecturas de codificador-decodificador, con autoatención (*self attention*) de múltiples cabezas [46].

En cada paso, se aplica un mecanismo de autoatención que modela directamente las relaciones entre todas las palabras de una oración, independientemente de su posición respectiva. En cada etapa es aplicado la autoatención para modelar las relaciones entre todas las palabras de una oración directamente, sin importar la posición. Para ilustrar cómo se realiza este proceso, se muestra un ejemplo extraído del blog Google AI, acerca de la atención en la resolución de correferencia, un problema común en la traducción automática. A continuación, se presentan dos frases en inglés y su respectiva traducción al español usando el traductor de Google:

The animal didn't cross the street because **it** was too tired.  
El animal no cruzo la calle porque estaba demasiado cansado**o**

The animal didn't cross the street because **it** was too wide.  
El animal no cruzo la calle porque era muy anch**a**

Para la primera oración "it" se refiere al animal y en la segunda se refiere a la calle. En español "animal" y "street" tienen géneros diferentes, entonces la traducción del adjetivo en estas oraciones al español depende del género del sustantivo al que "it", hace referencia.



**Figura 7.** Distribución de autoatención del codificador para la palabra "it" [47].

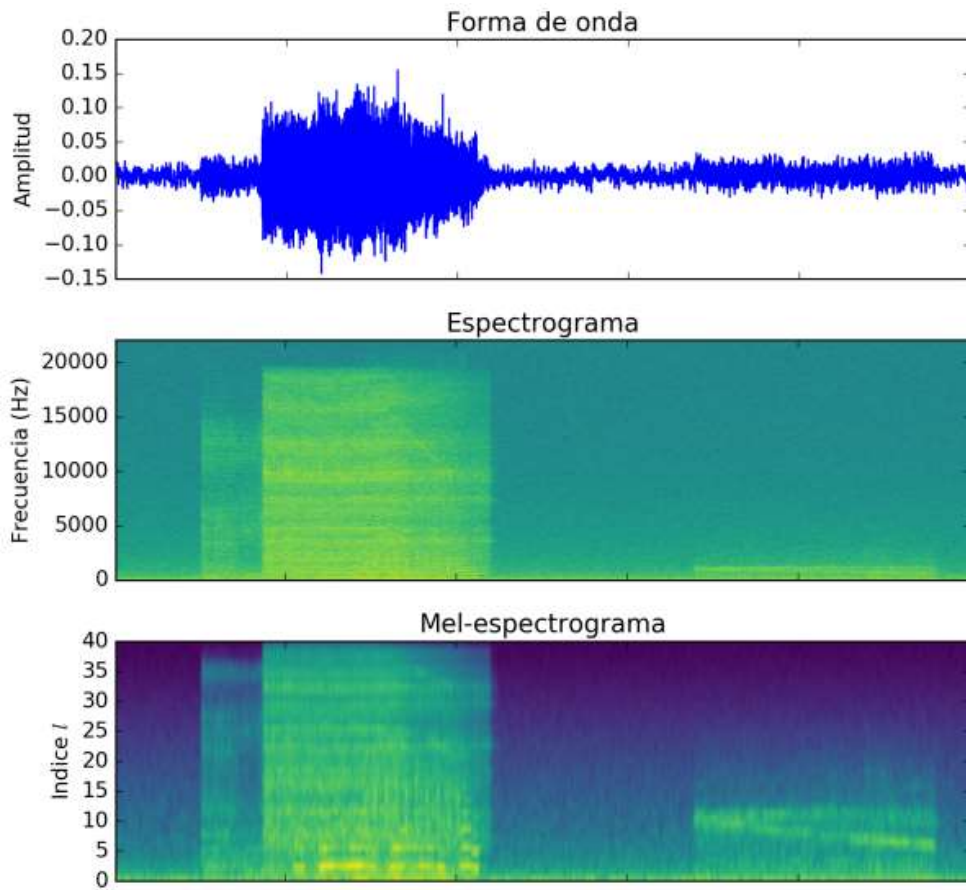
En la Figura 7 se visualiza a qué palabras prestó atención el codificador al calcular la representación final de la palabra "it" y arroja algo de luz sobre cómo la red tomó la decisión. En uno de sus pasos, el Traductor de Google identificó claramente los dos sustantivos a los cuales "it" podría referirse, y mostró una clara distinción en la atención prestada a cada uno, reflejando su capacidad de elección en diferentes contextos [47].

### **3.9. Espectrograma**

Un espectrograma es una representación visual del espectro de frecuencias de una señal a medida que varía con el tiempo. Aplicando varias transformaciones rápidas de Fourier (FFT), en varios segmentos de la señal de audio, este proceso se conoce como transformación de Fourier de tiempo corto (STFT).

#### **3.9.1. Espectrograma de Mel**

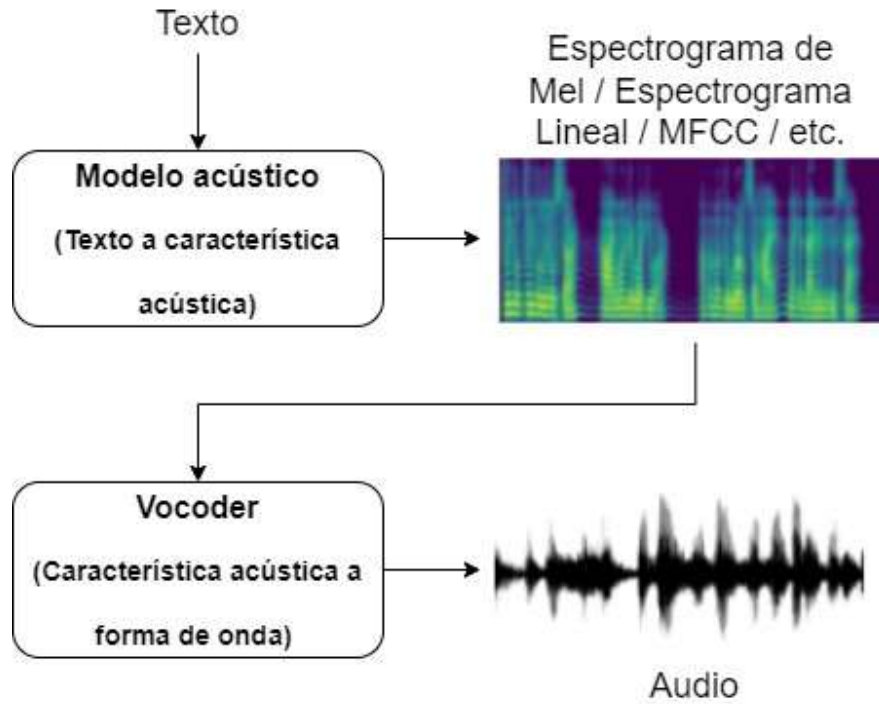
Un espectrograma de Mel es un espectrograma en el que las frecuencias se convierten a la escala de Mel. La escala de Mel fue propuesta en 1937 por Stevens, Volkman, y Newman, debido a que los humanos no percibimos frecuencias en una escala lineal y somos pésimos en detectar diferencias en frecuencias altas. Tal es el caso que no notaremos una diferencia si a 10000 Hz, se tiene una variación de más o menos 500 Hz, pero fácilmente notamos una diferencia si a 500 Hz se aumenta o disminuye unos 200 Hz, a pesar de que la diferencia sea menor. La escala Mel propone una unidad de tono, tal que distancias iguales en el tono sonarán igualmente distantes para el oyente [48]. En la Figura 8 se muestra la forma de onda de la señal de audio y el correspondiente espectrograma y espectrograma de Mel.



**Figura 8.** Ilustración de la forma de onda de la señal, espectrograma, y espectrograma de Mel [49].

### 3.10. Modelo acústico

Como se muestra en la Figura 9, los modelos acústicos tienen como objetivo generar características acústicas que luego el vocoder convertirá en forma de onda, es decir, en audio. Se han probado diferentes tipos de características acústicas, como coeficientes cepstrales en las frecuencias de Mel (MFCC), coeficientes Mel generalizados (MGC), aperiodicidad de banda (BAP), frecuencia fundamental (F0), sonoro/ sordo (V/UV), coeficientes cepstrales de frecuencia de corteza (BFCC) y los más utilizados, los espectrogramas de Mel [50].



**Figura 9.** Pipeline de sintetizadores de texto a voz [51] [50].

Existen una gran variedad de modelo acústicos con diferentes arquitecturas y enfoques. En la Tabla 2 se muestran algunos modelos acústicos que se percibieron como populares basándose en el número de citaciones en Google Académico con un umbral mínimo de 200 citaciones.

<b><i>Sintetizadore texto a voz</i></b>	<b>Representación acústica (Salida)</b>	<b>Enfoque</b>	<b>Estructura</b>	<b>Citaciones (30/10/22)</b>
<i>Tacotron</i>	Espectrograma Lineal	Seq2Seq	Hibrido / RNN	1415
<i>Tacotron 2</i>	Espectrograma de Mel	Seq2Seq	RNN	1935
<i>Deep Voice</i>	Espectrograma de Mel	/	CNN	592
<i>Deep Voice 2</i>	Espectrograma de Mel	/	CNN	258
<i>Deep Voice 3</i>	Espectrograma de Mel	Seq2Seq	CNN	299
<i>DCTTS</i>	Espectrograma de Mel	Seq2Seq	CNN	294
<i>FastSpeech</i>	Espectrograma de Mel	Seq2Seq	Transformer	562

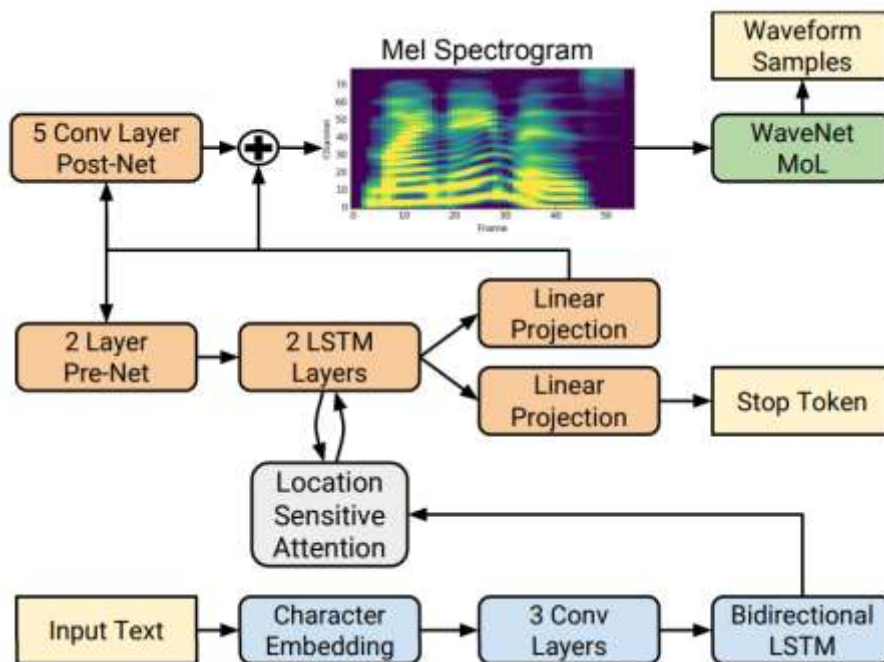
<i>FastSpeech 2</i>	Espectrograma de Mel	Seq2Seq	Transformer	457
<i>AlignTTS</i>	Espectrograma de Mel	Seq2Seq	Transformer	446

**Tabla 2.** Comparativa de diferentes modelos acústicos de texto a voz [50].

A continuación, se profundizará en un modelo por estructura (RNN, CNN, y Transformer). Los modelos escogidos para su explicación detallada son los más citados por cada estructura y entre dichos modelos se presentan las versiones más recientes, por ejemplo, Deep Voice 3 frente a Deep Voice 2 y Deep Voice.

### 3.10.1. Tacotron 2

Tal como se muestra en la Figura 10, la arquitectura propuesta para Tacotron 2 se compone de dos componentes: primero, un modelo compuesto de una red neuronal recurrente Seq2Seq con atención, que predice una secuencia de *frames* de espectrograma de Mel a partir de una secuencia de caracteres. La red está compuesta por un encoder y un decoder con atención, el encoder convierte una secuencia de caracteres en una representación de características ocultas que el decoder consume para predecir un espectrograma.



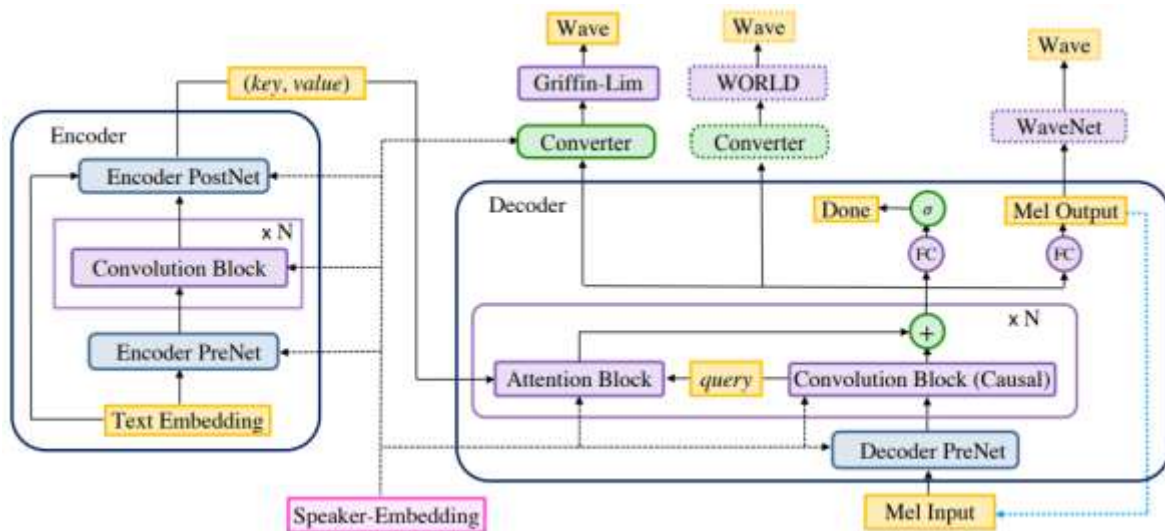
**Figura 10.** Arquitectura de Tacotron 2 [7].



Una vez el decoder predice el espectrograma es pasado a la Post-Net, la cual contiene cinco capas convolucionales que predicen un “residuo” que se sumará al espectrograma del decoder para mejorar la reconstrucción final. El espectrograma final es pasado a la segunda parte de la arquitectura, una versión modificada de WaveNet para generar audio a partir de un espectrograma de Mel [7].

### 3.10.2. Deep Voice 3

La arquitectura de Deep Voice 3 como se muestra en la Figura 11, utiliza un encoder con capas convolucionales residuales para codificar texto en una clave por intervalo de tiempo y un vector de valor para un decoder basado en atención. El decoder es completamente convolucional causal, utiliza la representación aprendida con un mecanismo de atención convolucional de saltos múltiples para predecir los espectrogramas de magnitud logarítmica en escala de Mel, de manera autorregresiva que corresponden al audio de salida.

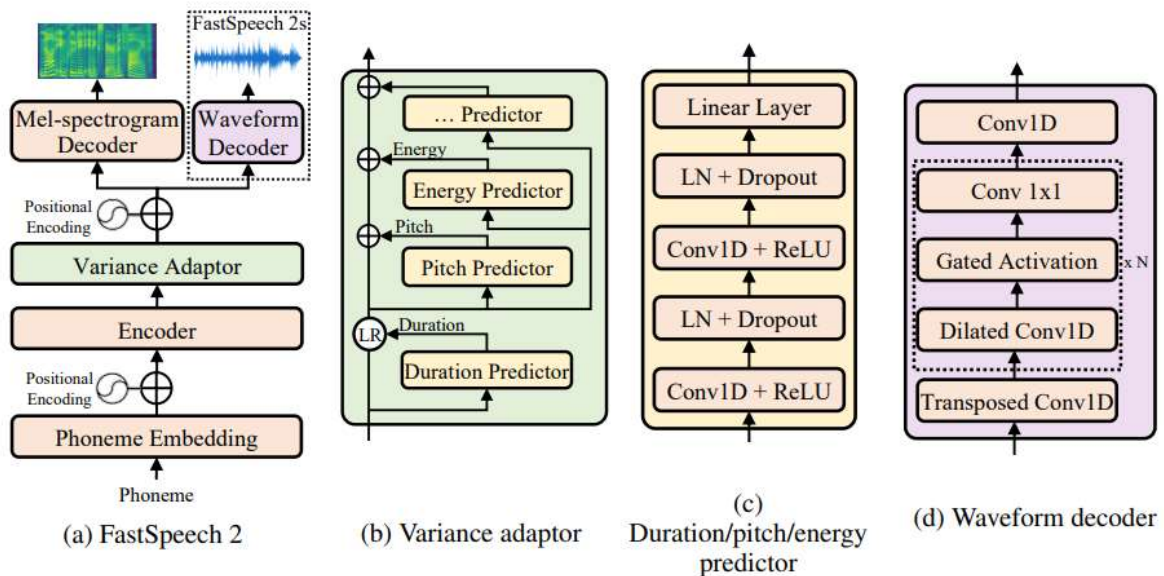


**Figura 11.** Arquitectura de Deep Voice 3 [8].

Los estados ocultos del decoder luego se alimentan a un convertidor totalmente convolucional que predice los parámetros necesarios del vocoder para la síntesis de formas de onda, a partir de los estados ocultos del decoder [8].

### 3.10.3. FastSpeech 2

Debido a que FastSpeech 2 requiere una secuencia de fonemas de entrada, es necesario primero convertir la secuencia de caracteres del texto a una secuencia de fonemas con la ayuda de una herramienta externa. La arquitectura general de FastSpeech 2 se muestra en la Figura 12(a). El encoder convierte la secuencia de fonemas en una secuencia oculta de fonemas y después el adaptador de varianza agrega diferente información como la duración, el tono, y la energía a la secuencia oculta. Finalmente, el decoder convierte la secuencia oculta ya adaptada a secuencia de espectrograma de Mel. Esto se realiza en paralelo.



**Figura 12.** Arquitectura de FastSpeech 2 [10].

FastSpeech 2 utiliza un bloque feed-forward Transformer, que es un stack de autoatención y convolución de una dimensión como estructura básica para el codificador y el decodificador del espectrograma de Mel [10].

### 3.11. Vocoder

En el sentido tradicional de la palabra un vocoder es un analizador y sintetizador de voz humana inventado en 1938 por Homer Dudley en Bell Labs [52]. También se ha utilizado ampliamente como instrumento musical electrónico. En la síntesis de texto a voz basada en el aprendizaje profundo, los vocoders neuronales son utilizados para convertir las representaciones acústicas de una señal de audio en formas de onda. Un vocoder se centra en sintetizar formas de onda a partir de representaciones de baja dimensión, como los espectrogramas Mel [53]. Dentro de los diferentes modelos que han surgido, Wavenet es probablemente el más conocido por la increíble calidad de audios que puede generar y que incluso es capaz de generar música [2], lamentablemente es muy ineficiente y lento para aplicaciones de tiempo real.

Al igual que los modelos acústicos, hay una gran variedad de vocoder disponibles, cada uno con diferentes diseños de arquitecturas o enfoques, pero por simplicidad en la Tabla 3 se muestran algunos vocoders que se percibieron como populares basándose en el número de citas en Google Académico con un umbral mínimo de 200 citas.

<b><i>Sintetizadore texto a voz</i></b>	<b>Representación acústica (Entrada)</b>	<b>Enfoque</b>	<b>Estructura</b>	<b>Citaciones (30/10/22)</b>
<i>WaveNet</i>	Característica lingüística	/	CNN	4796
<i>Parallel WaveNet</i>	Característica lingüística	Flow	CNN	730
<i>WaveGAN</i>	/	GAN	CNN	480
<i>Parallel WaveGAN</i>	Espectrograma de Mel	GAN	CNN	486
<i>MelGAN</i>	Espectrograma de Mel	GAN	CNN	534
<i>SampleRNN</i>	/	/	RNN	533
<i>LPCNet</i>	BFCC	/	RNN	334

<i>WaveRNN</i>	Característica lingüística	/	RNN	709
<i>WaveGlow</i>	Espectrograma de Mel	Flow	Hibrido / CNN	777
<i>HiFi-GAN</i>	Espectrograma de Mel	GAN	Hibrido / CNN	433

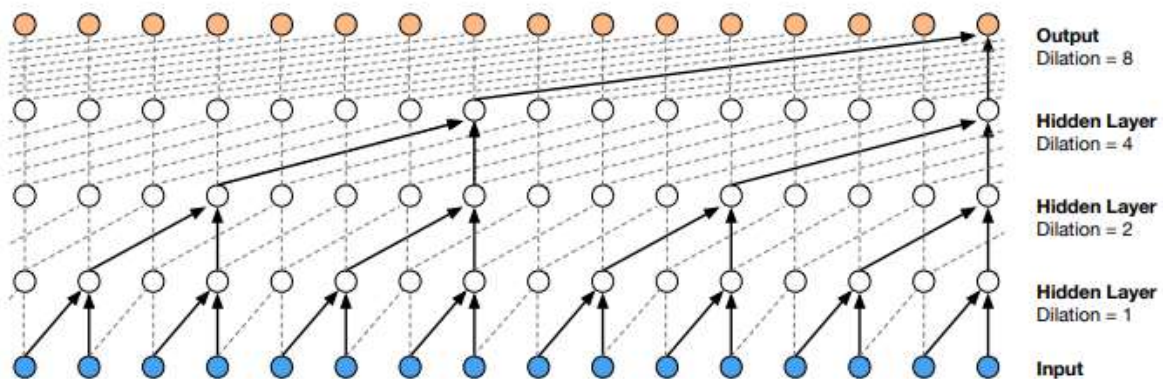
**Tabla 3.** Comparativa de diferentes vocoder para texto a voz [50].

A continuación, se profundizará en un modelo por arquitectura (RNN, CNN, e híbrido Flow y GAN). Los modelos escogidos para ver a más detalle serán los más citados por cada estructura.

Wavenet y WaveRNN originalmente fueron propuestos como modelos que directamente convertían características lingüísticas a forma de onda por lo que con la ayuda de un analizador de texto que generara estas características podrían considerarse modelos end-to-end, pero en trabajos posteriores se limitan a usarlos solo como vocoders.

### 3.11.1. WaveNet

WaveNet fue desarrollada por DeepMind inspirándose en la arquitectura de PixelCNN y PixelRNN, siendo estos últimos modelos para la generación de imágenes, pero a diferencia de los modelos PixelNet, WaveNet es adaptada para ser unidimensional.



**Figura 13.** Stack de capas convolucionales causales dilatadas [2].

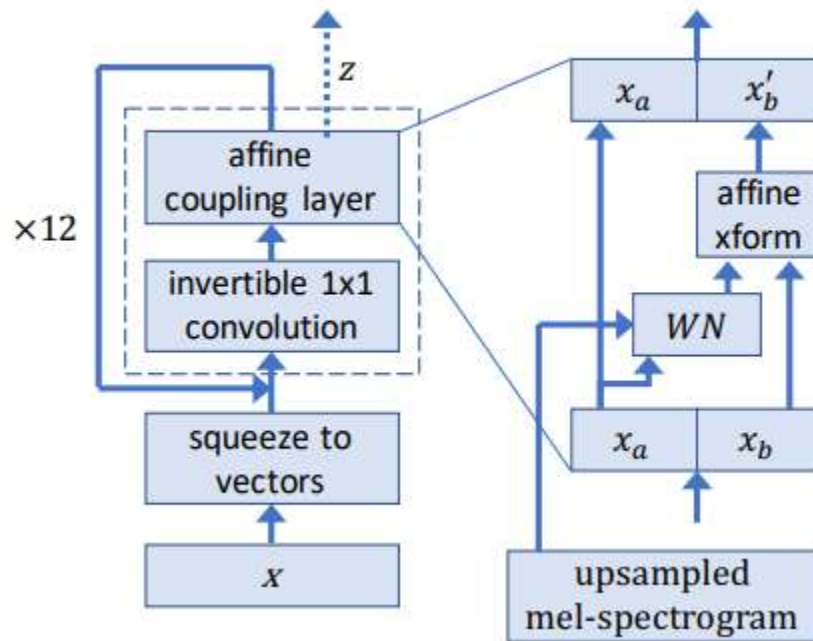
WaveNet es una red neuronal completamente convolucional como la mostrada en la Figura 13, donde las capas convolucionales tienen varios factores de dilatación que permiten que su campo receptivo crezca exponencialmente con la profundidad y cubra miles de pasos de tiempo. Durante la síntesis de voz en cada paso se extrae un valor de la distribución de probabilidad calculada por la red. Luego, este valor se retroalimenta a la entrada y se realiza una nueva predicción para el siguiente paso. La muestra generada es un audio complejo y realista, pero construir la muestra paso a paso es computacionalmente costoso [2].

### **3.11.2. WaveRNN**

WaveRNN es un modelo de síntesis de texto a voz basado en redes neuronales recurrentes [54]. El modelo propuesto utiliza una RNN para modelar la distribución condicional de probabilidad de la forma de onda de audio, y luego un modelo de muestreo inverso que genera la forma de onda de audio a partir de la salida de la red neuronal recurrente.

### **3.11.3. WaveGlow**

WaveGlow es una red basada en Flow (es decir en un modelo generativo basado en flujos autoregresivos), capaz de generar voz de alta calidad a partir de espectrogramas de Mel, mediante el muestreo de una distribución. Específicamente, se toman muestras de una Gaussiana esférica de promedio cero con el mismo número de dimensiones que la salida deseada. Dichas muestras se pasan por una serie de capas que transforman la distribución simple en una que tiene la distribución deseada.



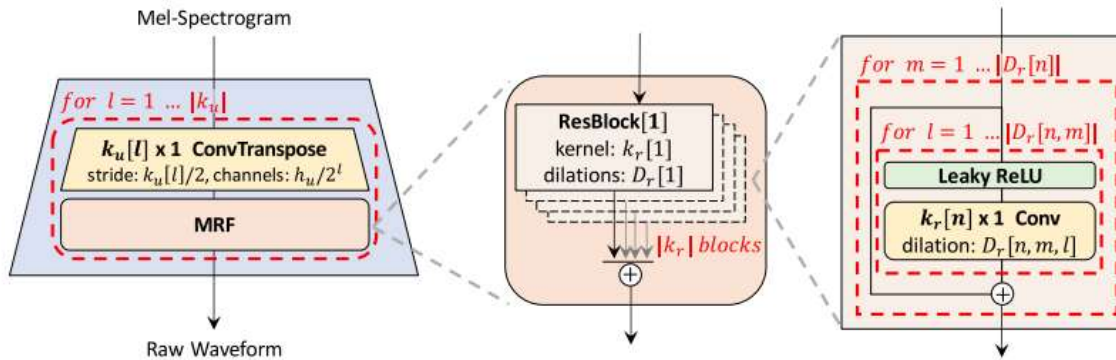
**Figura 14.** Arquitectura de WaveGlow [55].

En la Figura 14 se muestra WaveGlow, el cual implementa una sola red entrenada en una única función de costo: maximizar la probabilidad de los datos de entrenamiento, lo que hace que el procedimiento de entrenamiento sea simple y estable [55]. También WaveGlow aprovecha el flujo generativo para entrenar e inferir en paralelo. Sin embargo, por lo general se necesitan apilar varias iteraciones  $T$  ( $T$  es el número de pasos o iteraciones en modelos basados en Flow), para garantizar la calidad del mapeo entre los datos y las distribuciones anteriores.

#### 3.11.4. HiFi-GAN

HiFi-GAN es un modelo de síntesis de audio de alta fidelidad basado en redes generativas adversariales (GAN, por sus siglas en inglés). La arquitectura de HiFi-GAN se compone de dos partes principales: un generador y un discriminador.

El generador que se puede ver en la Figura 15, se compone de una red neuronal totalmente convolucional. Utiliza espectrogramas de Mel como entrada y lo aumenta a través de convoluciones transpuestas hasta que la longitud de la secuencia de salida coincide con la resolución temporal de la forma de onda.



**Figura 15.** Arquitectura del generador de HiFi-GAN [56].

Para el discriminador, se utiliza un discriminador de periodos múltiples (MPD) que consta de varios subdiscriminadores, cada uno de los cuales maneja una parte de las señales periódicas del audio de entrada [56].

### 3.12. Conjunto de datos

Un conjunto de datos (o dataset), es una colección de datos. En el caso de datos tabulados, un conjunto de datos contiene los valores para cada una de las variables organizadas como columnas, por ejemplo, la transcripción de un clip de voz que corresponde a cada audio del conjunto de datos que están organizados en filas. El conjunto de datos también puede consistir en una colección de documentos o de archivos [57]. En el ámbito de texto a voz, un buen conjunto de datos debe cumplir con los siguientes requisitos [58]:

- Distribución gaussiana en las longitudes de los clips de audio y el texto. Permite asegurarse que haya suficientes clips de voz tanto cortos como largos.

- Libre de errores, bugs, o archivos corruptos. Permite asegurarse que el identificador y las transcripciones corresponden correctamente al clip de voz.
- Sin ruido de fondo. El ruido de fondo puede hacer que el modelo presente problemas y es probable que el resultado final sea subóptimo.
- Tono y acentos compatibles entre los clips de voz. En los casos que se estén entrenando con solo una voz, es necesario mantener una consistencia entre los tonos y el acento, ya que estas diferencias entre muestras degradan el rendimiento del modelo.
- Buena cobertura de fonemas. Asegura que el conjunto de datos cubra una buena parte de los fonemas. En español no todas las variedades de lengua tienen la misma cantidad de fonemas. En la mayoría de las variedades se distinguen por lo menos 18 fonemas consonánticos, pero por ejemplo, en la versión castellana se pueden presentar algunos fonemas extras como /θ/ [59].



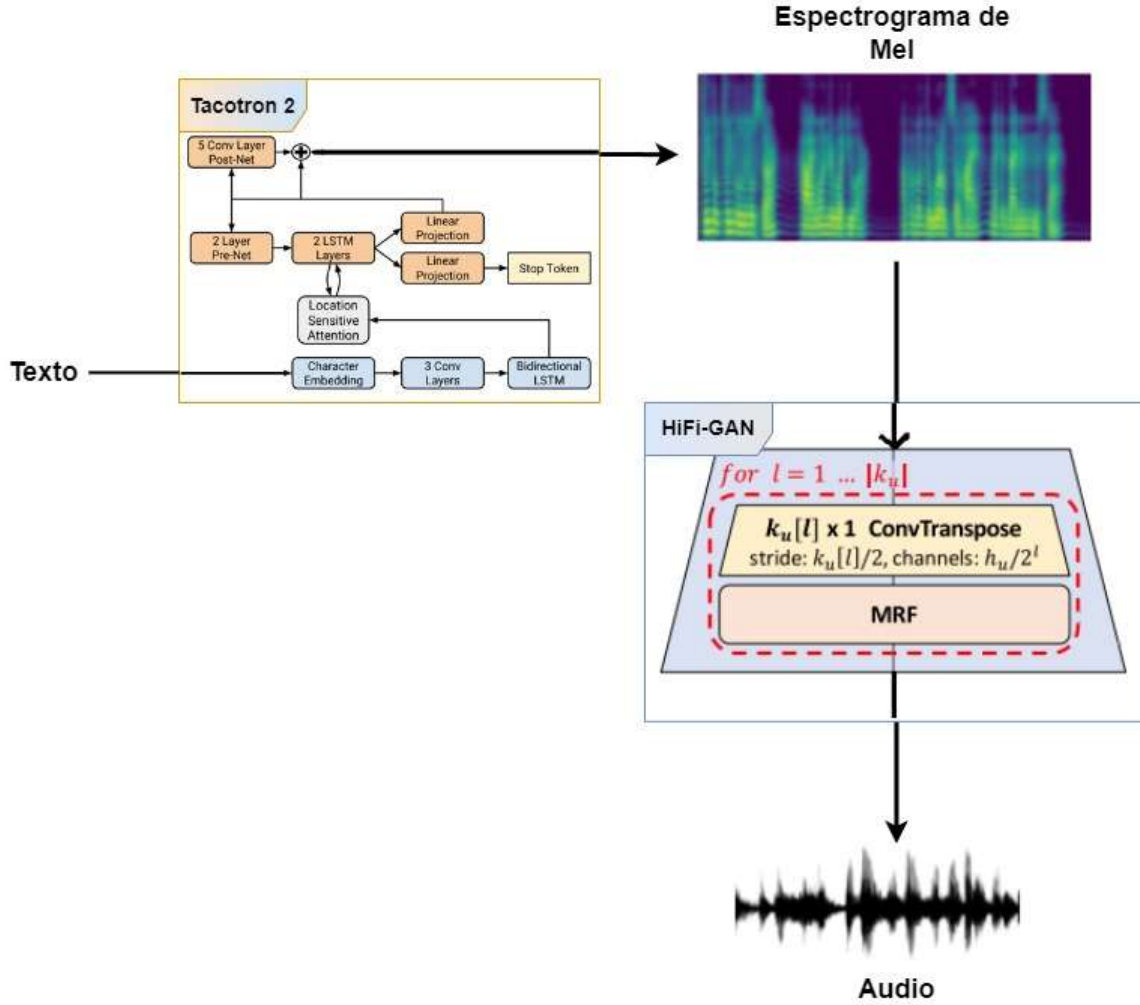
## **CAPÍTULO 4**

### **MODELO PROPUESTO**

En este capítulo se presenta la arquitectura del modelo de síntesis de texto a voz propuesto en este trabajo, así como los criterios utilizados para elegir el modelo acústico y el vocoder. Se mostrará cómo se seleccionaron los conjuntos de datos utilizados teniendo en cuenta factores como el dialecto utilizado y la disponibilidad en línea. Además, se explicará cómo fue entrenado el modelo y cómo fue su despliegue en la web.

#### **4.1. Arquitectura del modelo propuesto**

Para el desarrollo de este trabajo, se ha seleccionado el modelo de síntesis de voz Tacotron 2 para el modelo acústico, y HiFi-GAN para el vocoder. En la Figura 16 se muestra un diagrama de la arquitectura del modelo de sintetizador de texto a voz propuesto. En este proceso, el texto de entrada pasa por el modelo acústico, Tacotron 2, que lo convierte a la característica acústica, espectrograma de Mel. La información que busca representar el espectrograma son las propiedades de la señal de audio de habla, tales como la entonación, la duración de los fonemas, las transiciones entre ellos, etc. Una vez obtenido el espectrograma, es utilizado como entrada para el vocoder, HiFi-GAN, que se encarga de transformarlas en señales de audio de la voz generada.



**Figura 16.** Diagrama del modelo propuesto.

## 4.2. Selección del modelo acústico

En la Tabla 4 se encuentra una comparación del *Mean Opinion Score* (MOS), que es una métrica utilizada para evaluar la calidad de los audios de voz, así como de la complejidad temporal en entrenamiento e inferencia con respecto a la longitud de secuencia  $N$  para los modelos acústicos previamente seleccionados. Tacotron 2 tiene una complejidad de entrenamiento e inferencia de tiempo lineal  $O(N)$  debido a su estructura recursiva, mientras que DeepVoice 3 y FastSpeech 2 tienen una complejidad de tiempo constante  $O(1)$  en el entrenamiento y, en el caso de

FastSpeech 2, también en la inferencia. La inferencia hace referencia al tiempo necesario para realizar una predicción sobre nuevos datos de entrada.

En este trabajo de grado se escogió el modelo Tacotron 2 debido a que, a pesar de ser más lento en entrenar que Deep Voice 3 y FastSpeech 2, y más lento en inferir que FastSpeech 2, es el modelo que al sintetizar los audios de voz suenan más realistas o de mejor calidad. Esto se puede verificar usando como referencia la puntuación MOS reportada en la Tabla 4 [60]. De acuerdo con dichos valores, Tacotron 2 obtuvo una puntuación de 4.25 frente a 3.93 y 1.90 de DeepVoice 3 y FastSpeech 2, respectivamente. Además, para tomar esta decisión se tuvo en cuenta que Tacotron 2 cuenta con vastas implementaciones disponibles en la red.

<b>Modelo</b>	<b>Entrenando</b>	<b>Inferencia</b>	<b>MOS</b>
Tacotron 2	$O(N)$	$O(N)$	$4.25 \pm 0.17$
DeepVoice 3	$O(1)$	$O(N)$	$3.93 \pm 0.19$
FastSpeech 2	$O(1)$	$O(1)$	$1.90 \pm 0.43$

**Tabla 4.** Comparación de modelos acústicos [50] [60].

### 4.3. Selección del vocoder

A pesar de que WaveNet suele presentar la mejor calidad de voz generada entre todos los modelos disponibles, sufre de una velocidad para inferir lenta y requiere demasiada potencia computacional para aplicaciones de la vida real. De acuerdo con la Tabla 5, WaveGlow usa una diferente variable para la medición de la complejidad temporal de entrenamiento e inferencia, para WaveGlow la complejidad depende del número de iteraciones  $T$ . Por lo tanto, se dificulta la comparación de este vocoder con otros de complejidad temporal lineal como WaveRNN. Afortunadamente, HiFi-GAN cuenta con una complejidad temporal constante de entrenamiento e inferencia por lo que se podría asumir que es más rápido que WaveRNN y WaveGlow. Además, como factor adicional se tomó en cuenta la existencia y disponibilidad de éstos en la red y se encontró un modelo de

HiFi-GAN integrado con Tacotron 2, el modelo acústico previamente seleccionado. Por estas razones en este trabajo se usó el vocoder HiFi-GAN.

Modelo	Entrenando	Inferencia
WaveNet	$O(1)$	$O(N)$
WaveRNN	$O(N)$	$O(N)$
WaveGlow	$O(T)$	$O(T)$
HiFi-GAN	$O(1)$	$O(1)$

**Tabla 5.** Comparación de vocoders [50].

#### 4.4. Selección del conjunto de datos

Se buscaron diferentes conjuntos de datos que cumplieran con los requisitos descritos en la sección 3.12. Como requisito adicional, los conjuntos de datos deben estar compuestos por audios .wav en lenguaje español y un archivo .txt, .tsv o .csv, con el nombre o identificador del audio y la transcripción de este. De los conjuntos de datos encontrados se preseleccionaron los siguientes diez.

Dataset	Idioma	Numero de audios	Tiempo total (h)	Licencia
LJSpeech	Inglés	13100	23h 55m	Dominio público
CSS10 Spanish	Español (Ibérico)	11111	23h 49m	Dominio público
120h Spanish Speech	Español (Ibérico)	112845	120h	Dominio público
The M-AILABS Speech Dataset	Español (Ibérico, Mexicano y Argentino)	59297	108h 34m	M-AILABS

Crowdsourced high-quality Chilean/Colombian/Peruvian/Puerto Rico/Venezuelan Spanish speech data set.	Español (Chileno, Colombiano, Peruano, Puertorriqueño o Venezolano)	Chileno: 4374 Colombiano : 4903 Peruano: 5447 Puertorriqueño: 617 Venezolano: 3357	Chileno: 7h 8m Colombiano : 7h 34m Peruano: 9h 13m Puertorriqueño: / Venezolano: 4h 48m	CC-BY-4.0-SA
Spanish TTS Speech Corpus (Appen)	Español (Ibérico)	1787	1h 45m	ELRA END USER

**Tabla 6.** Diferentes datasets usados para texto a voz.

A continuación, se describe en detalle algunos de los conjuntos de datos.

- LJSpeech. Es el conjunto de datos más popularmente usado en el ámbito de texto a voz. Cuenta con 13100 audios de 1 a 10 segundos que acumulan cerca de 24 horas totales de audios y transcripciones depuradas, además de ser de dominio público. Sin embargo, este conjunto presenta el problema que está en inglés. Por lo que los fonemas con los cuales se entrena el modelo no se pueden usar para un modelo que se piensa hacer en español.
- 120h Spanish Speech, The M-AILABS Speech Dataset y CSS10 Spanish. Estos conjuntos de datos no tienen costo y contiene una buena cantidad y tiempo total de audios, por lo que podrían ser una buena opción para ser usados. Sin embargo, se busca preferiblemente un conjunto de datos en dialecto colombiano para el desarrollo del proyecto, esto debido a que el dialecto español y el dialecto colombiano tienen

algunas diferencias fonéticas. Una de las diferencias más notables es el seseo en Colombia, que es la pronunciación de ⟨c⟩ ante ⟨e⟩ o ⟨i⟩, ⟨s⟩ y ⟨z⟩ no presentan distinción asimilándose a la consonante fricativa alveolar sorda /s/, mientras que en el dialecto español sí presentan una distinción, pronunciando ⟨c/z⟩ como la consonante fricativa dental sorda /θ/ y ⟨s⟩ como /s/.

- Spanish TTS Speech Corpus (Appen). Cuenta con un total de 1 hora y 45 minutos de audio, los cuales fueron grabados por un único hablante y corresponden al dialecto español. Es importante destacar que se prioriza el uso del dialecto colombiano en este trabajo. Además, se debe tener en cuenta que la utilización de este conjunto de datos requiere la adquisición de una licencia para su uso.
- Crowdsourced high-quality Spanish speech data set. Estos conjuntos de datos ofrecen el beneficio de estar en múltiples dialectos como el dialecto colombiano. A pesar de que cuenta con 7 horas y 34 minutos de audios, tras una inspección adicional de las 7 horas y media, solo es posible usar alrededor de 10 minutos de audio total para este proyecto debido a que el resto de audios suelen ser las mismas frases pero con diferentes lectores.

Finalmente, para este trabajo se usó el conjunto de datos ChqCSsdt (Crowdsourced high-quality Colombian Spanish speech data set). Sin embargo, debido a que el tiempo total de audios usados es de solamente 11 minutos, lo cual puede resultar insuficiente, en este proyecto se creó además un conjunto de audios específico para entrenar el modelo de síntesis de texto a voz con el objetivo de probar y simular la calidad de una voz sintetizada a partir de grabaciones caseras con una mínima edición de los audios.

#### **4.4.1. Creación del conjunto de datos propio**

Se grabaron 250 clips de audio con una duración total de alrededor de 20 minutos, cada uno con su transcripción respectiva de la frase. Durante la creación del conjunto de datos se tuvieron en cuenta los requisitos previamente descritos en la sección 3.12. Para mejorar la calidad de los clips de audio, se les aplicó un proceso de edición utilizando herramientas como Audacity y Adobe Audition. Durante la edición, se eliminaron elementos no deseados como el ruido de fondo, los ecos, y los sonidos que produce la boca al hablar. El objetivo de este proceso de edición fue mejorar la calidad del audio en general, para que el modelo pudiera aprender de manera más efectiva a partir de estos datos de entrada.

El conjunto de datos creado se puede encontrar en el siguiente enlace:

- [Conjunto de datos](#).

#### **4.5. Entrenamiento del modelo propuesto**

En el entrenamiento del modelo se decidió experimentar realizando cuatro entrenamientos por separado, buscando evaluar la calidad de voz producida por el modelo de síntesis en las cuatro variaciones. En los cuatro experimentos se utilizó el mismo modelo de síntesis. En tres de los experimentos se usó la configuración por defecto de los hiperparámetros. La diferencia entre los tres experimentos fue el conjunto de datos. En el experimento restante se decidió usar una configuración diferente en la tasa de aprendizaje.

Los hiperparámetros son parámetros configurables que se establecen antes del proceso de entrenamiento y afectan la forma en que el modelo aprende. El modelo disponía de los tres siguientes hiperparametros para su configuración:

- Tamaño de lote: determina cuántos ejemplos de entrenamiento se utilizan antes de realizar una actualización de los parámetros. Por defecto el modelo usa la siguiente fórmula para calcular el tamaño de lote a usar:

```
Tamaño_lote = int(audios/32)
```

- Taza de aprendizaje: determina qué tan rápido o qué tan lento se ajustan los parámetros del modelo en respuesta al error calculado durante el entrenamiento. Por defecto el modelo usa 0.0003
- Épocas de entrenamiento: se refiere a un recorrido completo de todo el conjunto de datos de entrenamiento a través del modelo durante el proceso de entrenamiento. En otras palabras, una época significa que todos los ejemplos de entrenamiento han sido presentados al modelo una vez. Por defecto el modelo usa 250

Los diferentes experimentos en adelante serán identificados según la cantidad de audios utilizados en cada uno de ellos, siendo nombrados como "Voz 149", "Voz 250", "Voz 2534" y "Voz 250b".

El cuaderno en el que se entrenó el modelo se encontró en la comunidad de [Discord de FakeYou](#).

#### 4.5.1. Voz 149

En el experimento Voz 149, se utilizó para el entrenamiento la parte de ChqCSsdt (Crowdsourced high-quality Colombian Spanish speech data set), seleccionada previamente en la sección 4.4.

Conjunto de datos:

- Cantidad de audios = 149
- Duración total de los audios = 11 min 20 seg



Hiperparámetros:

- Tamaño de lote = 4
- Taza de aprendizaje = 0.0003
- Épocas de entrenamiento = 250

Se puede escuchar audios de esta voz en el siguiente enlace [Audios](#).

#### **4.5.2. Voz 250**

En el experimento Voz 250, se usó el conjunto de datos propio presentado en la sección 4.4.1.

Conjunto de datos:

- Cantidad de audios = 250
- Duración total de los audios = 20 min 26 seg

Hiperparámetros:

- Tamaño de lote = 7
- Taza de aprendizaje = 0.0003
- Épocas de entrenamiento = 250

Se puede escuchar audios de esta voz en el siguiente enlace [Audios](#).

#### **4.5.3. Voz 2534**

En el experimento Voz 2534, se realizó un experimento con la totalidad del conjunto de datos ChqCSsdt, con el propósito principal de evaluar el rendimiento del modelo, al entrenar con audios de voz provenientes de múltiples voces diferentes, 16 voces para ser exactos.

Conjunto de datos:

- Cantidad de audios = 2534
- Duración total de los audios = 3 h 50 min 19 seg

Dado que un tamaño de lote grande implica una mayor demanda de memoria para almacenar los cálculos intermedios, fue recomendado no utilizar un tamaño de lote superior a 18, ya que podría sobrecargar la memoria de la GPU en Google Colab. Por lo tanto, se optó por un tamaño de lote de 10 para garantizar un funcionamiento seguro.

Hiperparámetros:

- Tamaño de lote = 10
- Taza de aprendizaje = 0.0003
- Épocas de entrenamiento = 250

Se puede escuchar audios de esta voz en el siguiente enlace [Audios](#).

#### 4.5.4. Voz 250b

En el experimento Voz 250, se usó el conjunto de datos propio presentado en la sección 4.4.1. A diferencia de Voz 250, se decidió usar diferente tasa de aprendizaje.

Conjunto de datos:

- Cantidad de audios = 250
- Duración total de los audios = 20 min 26 seg

Con el fin de obtener la tasa de aprendizaje óptima de manera teórica, se empleó la siguiente fórmula que se basa en el artículo original de Tacotron 2:

$$\text{Taza\_aprendizaje} = 0.001 * (\text{Tamaño\_lote} / 256) ** 0.5$$

Hiperparámetros:

- Tamaño de lote = 7
- Taza de aprendizaje = 0.00016
- Épocas de entrenamiento = 180

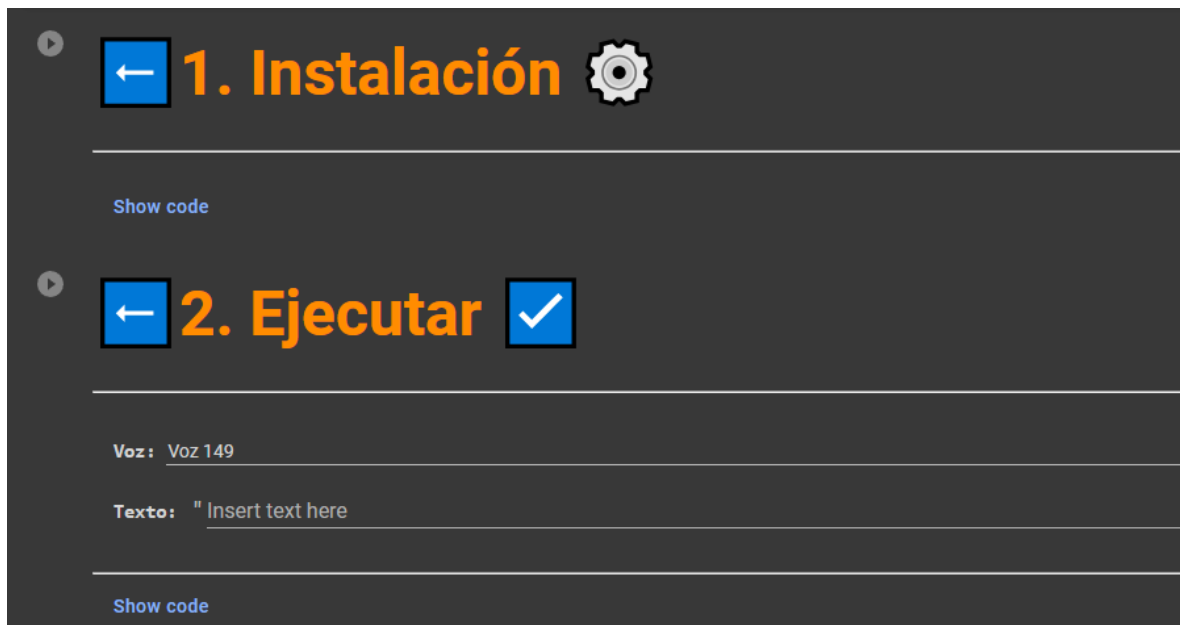
Se puede escuchar audios de esta voz en el siguiente enlace [Audios](#).

#### **4.6. Modelo desplegado**

El modelo fue desplegado en la web a través de Google Colab, utilizando el lenguaje de programación Python, el cual es compatible con esta plataforma. Google Colab cuenta con servicio en la nube que brinda acceso a recursos computacionales y proporciona la ventaja de poder relegar los requisitos de infraestructura, como la necesidad de tener una unidad de procesamiento gráfico (GPU).

Otra ventaja significativa de utilizar Google Colab, es que los usuarios no necesitan descargar ni instalar nada en sus dispositivos. El entorno de Colab es completamente basado en la nube, lo que significa que los modelos y las bibliotecas requeridas se ejecutan directamente en línea. Esto facilita el acceso y el uso del modelo sin la necesidad de preocuparse por softwares o los requisitos de hardware.

Con el objetivo de lograr una interfaz de fácil uso, especialmente para personas con dificultades para leer, se optó por un diseño minimalista en la aplicación. Se redujo la cantidad de elementos visuales a solo dos botones principales, una lista desplegable y un campo de texto para ingresar el texto a sintetizar. Se evitó utilizar explicaciones escritas extensas y en su lugar se utilizaron imágenes y gifs animados para proporcionar instrucciones visuales claras sobre cómo utilizar la aplicación. La Figura 17 muestra la apariencia de la aplicación, donde se puede apreciar la sencillez y la intuitividad del diseño.



**Figura 17.** Cuaderno de Google Colab con los modelos entrenados.

El modelo ofrece la opción de elegir entre las cuatro voces previamente entrenadas, para sintetizar el texto ingresado a voz. Esta flexibilidad permite al usuario seleccionar la voz que mejor se adapte a sus preferencias. Se puede acceder al modelo a través del siguiente [enlace](#).

## **CAPÍTULO 5**

### **PRUEBAS Y RESULTADOS**

En este capítulo se presentan las pruebas realizadas para evaluar el rendimiento del modelo de síntesis de texto a voz seleccionado. Además, se mostrará la metodología utilizada en la encuesta realizada a 55 personas, donde se midió la calidad de la voz sintetizada a través de la métrica MOS para cada versión del modelo. Finalmente, se realiza una comparación de audios de voz de ambas versiones.

#### **5.1. Pruebas**

Se llevaron a cabo múltiples pruebas para evaluar el rendimiento de los modelos en la síntesis de voz, donde se generaron y evaluaron diferentes audios. Estas pruebas tuvieron como objetivo analizar y comparar el desempeño de cada modelo en términos de la calidad de voz, de qué tan entendible es, y cómo se pronuncian los acentos. Para ello se recopilieron opiniones de evaluadores humanos.

##### **5.1.1. Pruebas y observaciones iniciales**

Se notó durante las pruebas iniciales realizadas que los audios sintetizados presentaron inconsistencias en algunos casos. Se observó que en los acentos que usan tildes, como en la sílaba "cé" de "océano", la pronunciación puede ser deficiente en ocasiones. Además, en Voz 149 se encontró que en algunas ocasiones se omitieron sílabas, lo cual puede alterar completamente el sentido de una frase como en el caso de la frase, "No estoy seguro de cómo proceder en esta situación.", donde se omitió la palabra "No", dando lugar a la interpretación opuesta.

Por otro lado, se observó que Voz 250, suele ser más consistente en términos de claridad de pronunciación y no se encontraron casos de palabras omitidas. Sin embargo, se encontraron algunas frases que no pudieron ser sintetizadas satisfactoriamente como la frase, "Por las tardes se escucha música en la Plaza", lo cual en ocasiones también ocurre con frases de longitud de entre 3 y 4 segundos. Se buscaron posibles razones para esto, como la duración de la frase, las palabras utilizadas o la frase en sí, pero no se encontró una razón clara para estas fallas.

En el caso de Voz 2534, al sintetizar voces entrenadas con este conjunto de datos, se encontró que no se obtenían resultados aceptables. La gran mayoría de audios sintetizados presentaban errores como largos silencios, sonidos ininteligibles o repeticiones de palabras en una misma oración, por esta razón se decidió excluirla de la encuesta con evaluadores humanos. Estos resultados indican que el modelo propuesto, entrenado con una sola voz se desempeña mejor en términos de calidad y consistencia en la síntesis de texto a voz.

El experimento Voz 250b se llevó a cabo con el propósito original de probar si el uso de la fórmula encontrada para calcular la tasa de aprendizaje óptima resultaría en una mejora notable en los audios sintetizados. Sin embargo, no se observaron mejoras significativas en los audios, ni se notaron cambios a partir de la época 40 del entrenamiento ([audios por cada 20 épocas](#)). Debido a estos resultados, se tomó la decisión de detener el entrenamiento del modelo en la época 180. Dado que Voz 250b se consideró más como un experimento para evaluar el impacto de la tasa de aprendizaje calculada, se decidió también excluirlo de la encuesta realizada con evaluadores humanos.

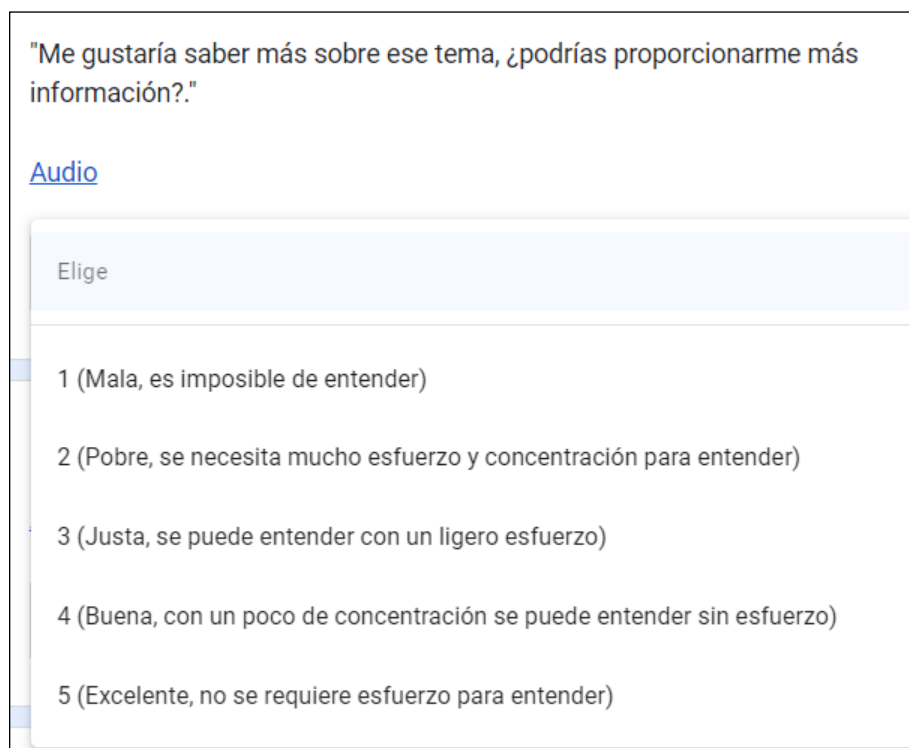
## **5.2. Métrica (MOS)**

El *Mean Opinion Score* (MOS) es una medida numérica utilizada para evaluar la calidad de audios de voz. Se basa en una escala de puntuación que va de 1 a 5,

donde 1 es la calidad más baja y 5 es la más alta. Los usuarios evalúan la calidad del audio en función de diferentes criterios, como la claridad, la distorsión, el eco, etc. El resultado final es el promedio aritmético de las puntuaciones obtenidas por los encuestados.

### 5.3. Resultados

La presente investigación tiene como objetivo analizar mediante el uso del *Mean Opinión Score* (MOS) la calidad de los audios de voz sintetizados. Para ello, se realizó una encuesta en la cual se seleccionaron 10 audios de Voz 149 y 10 de Voz 250. Se le solicitó a un grupo de 55 encuestados que evaluaran la calidad de cada uno de los audios de voz utilizando la escala de MOS. La encuesta utilizó una escala donde 1 representaba una calidad de audio "Mala", es decir, imposible de entender, y 5 representaba una calidad de audio "Excelente". La Figura 18 muestra un ejemplo de las preguntas realizadas en la encuesta.



"Me gustaría saber más sobre ese tema, ¿podrías proporcionarme más información?."

[Audio](#)

Elige

- 1 (Mala, es imposible de entender)
- 2 (Pobre, se necesita mucho esfuerzo y concentración para entender)
- 3 (Justa, se puede entender con un ligero esfuerzo)
- 4 (Buena, con un poco de concentración se puede entender sin esfuerzo)
- 5 (Excelente, no se requiere esfuerzo para entender)

**Figura 18.** Pregunta de opción múltiple con escala tipo MOS.

A partir de las respuestas recibidas se procedió a promediar la calificación que cada encuestado dio a cada pregunta. De esta forma, se obtuvieron 55 puntuaciones MOS por cada encuestado. Posteriormente, se promediaron estas 55 puntuaciones, obteniendo así el MOS percibido en promedio por todos los encuestados. Esta puntuación se puede ver en la Tabla 7 junto con el margen de error. Para determinar el margen de error, se consideró un nivel de confianza del 95%, que es el estándar en encuestas de tipo MOS. Para el caso de la proporción de muestra se consideró una calificación de 3.5 o superior como aceptable. Los resultados obtenidos permitieron determinar la calidad de voz generada y su nivel de aceptación por parte de los encuestados.

<b>Conjunto de datos</b>	<b>MOS</b>
Voz 149	4.14 ± 0.08
Voz 250	4.15 ± 0.09

**Tabla 7.** MOS con 95% de nivel de confianza.

Además, se incluyó una sección comparativa en la encuesta, donde se presentaron dos voces generadas y se les pidió a los encuestados que indicaran cuál de las dos presentaba una mejor calidad, claridad de pronunciación, acentos y puntuaciones, entre otros aspectos relevantes. En la Figura 19 se muestra un ejemplo. Para evitar sesgos, las preguntas no ofrecían ninguna indicación de qué opción correspondía a qué voz y se cambió qué opción correspondía a qué voz en cada pregunta.



"A pesar de las dificultades, el equipo logró superar las expectativas y ganar el campeonato." \*

[Opción 1](#)

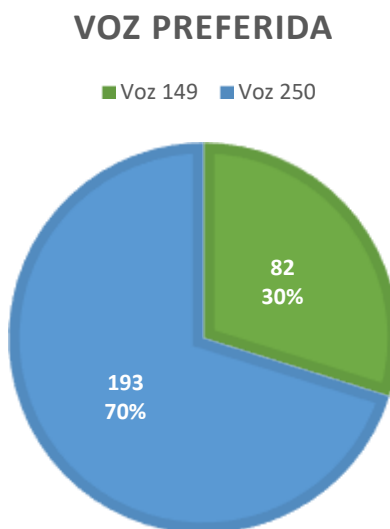
[Opción 2](#)

☐ Opción 1

☐ Opción 2

**Figura 19.** Pregunta comparativa.

De los 55 encuestados, se obtuvieron 275 respuestas sobre qué voz era superior teniendo en cuenta los criterios previamente establecidos. En la Figura 20 se muestra que, de las 275 respuestas, 193 encuestados que representan el 70%, eligieron "Voz 250", mientras que el resto, es decir, el 30%, respondieron que consideraban a "Voz 149" como superior.



**Figura 20.** Comparación entre las dos voces generadas.

La encuesta completa y los resultados se pueden encontrar en los enlaces:

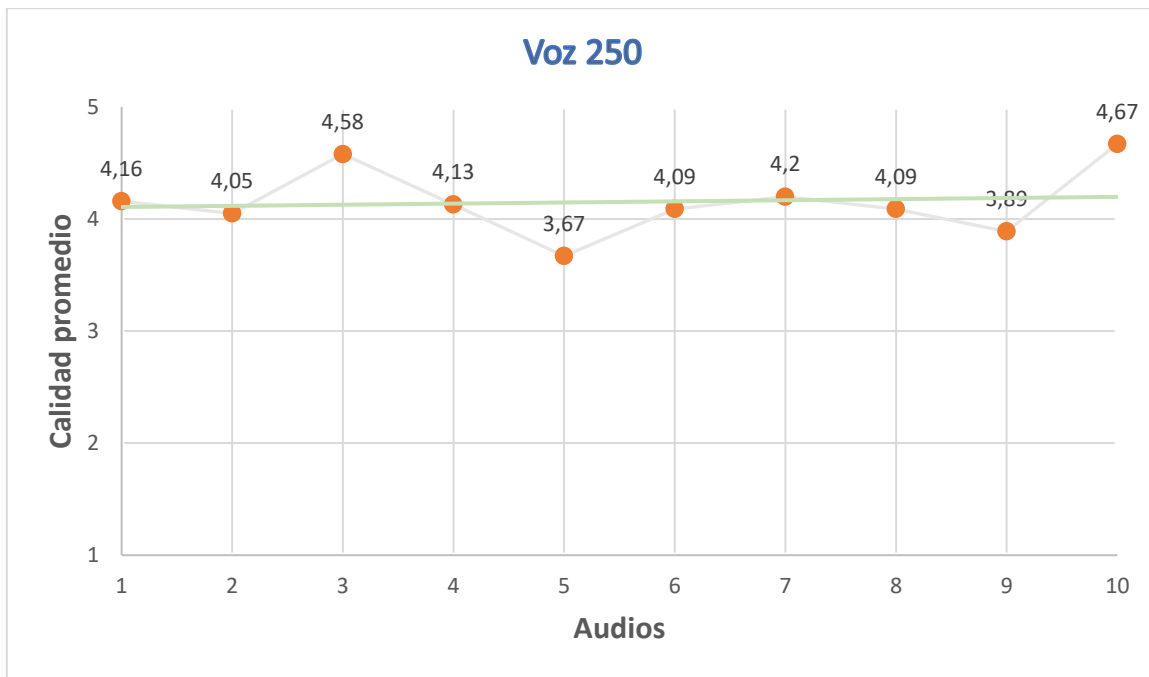
- [Encuesta.](#)
- [Respuestas.](#)

## 5.4. Análisis

En los resultados de la encuesta se observó que, en la sección de calificación del MOS, los valores obtenidos por ambos fueron muy similares, con una diferencia mínima de 0.01 en la calificación obtenida por Voz 250, en comparación con Voz 149. Se encontró que el audio mejor valorado por los encuestados fue el audio 8 de Voz 149 con un puntaje de 4.75, pero al mismo tiempo el audio peor valorado por los encuestados fue el audio 6 de Voz 149 con un puntaje de 3.15. Además, se observó que Voz 149, presenta una desviación estándar de 0.56, mientras que Voz 250 presenta una desviación de 0.29, lo que sugiere que Voz 250 es más consistente, como se puede apreciar en la comparación de los gráficos de dispersión de las figuras 21 y 22.



**Figura 21.** Gráfico de dispersión para Voz 149.



**Figura 22.** Gráfico de dispersión para Voz 250.

Según estos resultados, Voz 149 puede presentar tanto audios de alta calidad como también audios de peor calidad. Esto puede ser debido que Voz 149 no siempre presenta una correcta lectura del texto, pero cuando la presenta la voz sintetizada es nítida y entendible.

## 5.5. Comparación de resultados

Este trabajo de sintetizadores de texto a voz con enfoque en el dialecto colombiano presenta un valor agregado, ya que cubre una brecha existente en la investigación actual. A pesar de la creciente demanda de sistemas de síntesis de voz adaptados a diferentes dialectos y acentos, hay una falta de trabajos específicos que se centren en el dialecto colombiano. Este proyecto se destaca como uno de los primeros en abordar esta necesidad particular, proporcionando opciones que permitirá a los usuarios disfrutar de audios sintetizados en su lengua materna con la que tiene mayor familiaridad y naturalidad.

Los modelos Voz 149 y Voz 250 entrenados en este trabajo fueron evaluados y obtuvieron un puntaje MOS de 4.14 y 4.15, respectivamente. Una calificación MOS de 4.14 y 4.15, muestra que los modelos entrenados en este trabajo presentaron un desempeño superior a modelos como FastSpeech 2 y DeepVoice 3, que, en sus respectivos artículos originales, obtuvieron calificaciones MOS por debajo de 4. Es importante destacar que, en comparación con el artículo original de Tacotron 2, que obtuvo un puntaje MOS de 4.5, los modelos entrenados aún presentan limitaciones. Sin embargo, es relevante tener en cuenta que en el artículo original de Tacotron 2 se utilizó el vocoder WaveNet, que es conocido por su alto costo computacional y se considera uno de los mejores vocoders disponibles en la actualidad, actualmente usado en los servicios ofrecidos por Google. A pesar de esto, los resultados obtenidos en este trabajo demuestran un avance significativo en la síntesis de voz en el dialecto colombiano.

## **CAPÍTULO 6**

### **CONCLUSIONES**

En este trabajo se seleccionaron los modelos Tacotron 2 y HiFi-GAN, para implementar el sistema de síntesis de texto a voz. Esta elección se hizo con base a su rendimiento y capacidad para generar voces naturales en español. Se utilizaron dos conjuntos de datos para el entrenamiento del modelo. Estos conjuntos de datos contienen grabaciones de habla en español con dialecto colombiano, que es el acento de interés en este proyecto. El modelo seleccionado fue entrenado cuatro veces de forma independiente, utilizando conjuntos de datos diferente o variaciones de estos. Se realizó una encuesta con 55 participantes para evaluar la calidad de las voces sintetizadas. Estas métricas de desempeño proporcionaron una evaluación objetiva y cuantitativa del rendimiento del modelo. También el modelo entrenado se implementó en Google Colab, lo que permite su acceso y uso de manera gratuita. Los usuarios tienen la opción de elegir entre las cuatro voces entrenadas para sintetizar texto a voz.

Aunque ambos modelos obtuvieron valores de puntuación MOS similares con una diferencia de 0.01 que indica que los audios generados por ambos modelos son aceptables en cuanto a calidad (puntuación superior a 4), el modelo Voz 250 fue preferido por el 70% de los encuestados, mientras que el modelo Voz 149 obtuvo el 30% de preferencia. Al realizar una comparación directa se hacen más evidentes los problemas de calidad e inteligibilidad que presentan.

El modelo Voz 149 muestra una mayor variabilidad en las puntuaciones, se cree que esto es debido al conjunto de datos usados para Voz 149, el cual es más pequeño y por lo tanto menos audios para entrenar, pero al mismo tiempo los audios son de mejor calidad individual. Cuando se sintetizaba audio y salía una buena representación del texto, el audio presentaba mejor calidad que Voz 250, pero, cuando no salía una buena representación del texto debido a que se saltaba palabras o no pronunciaba bien algunos acentos, Voz 250 tomaba la ventaja

porque es consistente produciendo una buena representación del texto, a pesar de que la calidad de la voz fuera ligeramente inferior.

La ingeniería de sistemas, y en particular el campo de la inteligencia artificial, es un área en constante evolución. Esta dinámica añade dificultad a la investigación en este campo, ya que las tecnologías cambian rápidamente y nuevos estudios e innovaciones emergen constantemente. Un ejemplo de esto fue cuando Google Colab actualizó la versión de PyTorch, lo cual requirió actualizar el código del modelo para adaptarse a estos cambios. Esta necesidad de adaptación y actualización constante presenta desafíos a la investigación, ya que las bases pueden verse afectadas por cambios tecnológicos y nuevos descubrimientos.

### **6.1. Trabajos futuros**

El modelo utilizado en este trabajo se basa en un tono constante para la síntesis de voz. Sin embargo, se vislumbran interesantes posibilidades para modelos futuros que podrían ampliarse con el objetivo de acercarse aún más a la voz humana. Estos modelos podrían ser capaces de generar voces más expresivas y con emociones, capaces de reflejar diferentes estados de ánimo, como ira, miedo o alegría.

Además, se podría experimentar con conjuntos de datos que pongan un mayor énfasis en los acentos y tildes. Esto podría ayudar a abordar errores relacionados con los acentos y tildes, como los mencionados en la sección 5.1.1, y permitir una mejor pronunciación frente a estos.

## BIBLIOGRAFÍA

- [1] B. K. Das, *Twentieth century literary criticism*. Atlantic Publishers & Dist, 2005.
- [2] A. van den Oord *et al.*, “Wavenet: A generative model for raw audio”, *arXiv preprint arXiv:1609.03499*, 2016.
- [3] Y. Wang *et al.*, “Tacotron: Towards end-to-end speech synthesis”, *arXiv preprint arXiv:1703.10135*, 2017.
- [4] S. Ö. Arik *et al.*, “Deep voice: Real-time neural text-to-speech”, en *International Conference on Machine Learning*, PMLR, 2017, pp. 195–204.
- [5] J. Sotelo *et al.*, “Char2wav: End-to-end speech synthesis”, 2017.
- [6] W. Ping, K. Peng, y J. Chen, “Clarinet: Parallel wave generation in end-to-end text-to-speech”, *arXiv preprint arXiv:1807.07281*, 2018.
- [7] J. Shen *et al.*, “Natural tts synthesis by conditioning wavenet on mel spectrogram predictions”, en *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2018, pp. 4779–4783.
- [8] W. Ping *et al.*, “Deep voice 3: Scaling text-to-speech with convolutional sequence learning”, *arXiv preprint arXiv:1710.07654*, 2017.
- [9] J. Kim, S. Kim, J. Kong, y S. Yoon, “Glow-tts: A generative flow for text-to-speech via monotonic alignment search”, *Adv Neural Inf Process Syst*, vol. 33, pp. 8067–8077, 2020.
- [10] Y. Ren *et al.*, “Fastspeech 2: Fast and high-quality end-to-end text to speech”, *arXiv preprint arXiv:2006.04558*, 2020.
- [11] M. Karjalainen, “Review of speech synthesis technology”, *Helsinki University of Technology, Department of Electrical and Communications Engineering*, 1999.
- [12] J. L. Flanagan, *Speech analysis synthesis and perception*, vol. 3. Springer Science & Business Media, 2013.
- [13] L. J. Raphael, G. J. Borden, y K. S. Harris, *Speech science primer: Physiology, acoustics, and perception of speech*. Lippincott Williams & Wilkins, 2007.
- [14] D. H. Klatt, “Review of text-to-speech conversion for English”, *J Acoust Soc Am*, vol. 82, núm. 3, pp. 737–793, 1987.

- [15] J. Allen, M. S. Hunnicutt, D. H. Klatt, R. C. Armstrong, y D. B. Pisoni, *From text to speech: The MITalk system*. Cambridge University Press, 1987.
- [16] T. Siri, “Deep learning for Siri’s voice: On-device deep mixture density networks for hybrid unit selection synthesis”, *Mach. Learn. J*, vol. 1, núm. 4, 2014.
- [17] T. Chen, A. Kumar, P. Nagarsheth, G. Sivaraman, y E. Khoury, “Generalization of Audio Deepfake Detection.”, en *Odyssey*, 2020, pp. 132–137.
- [18] J. Kahn, “Apple engineers share behind-the-scenes evolution of Siri & more on Apple Machine Learning Journal”. 2017.
- [19] S. Arik *et al.*, “Deep voice 2: Multi-speaker neural text-to-speech”, *arXiv preprint arXiv:1705.08947*, 2017.
- [20] Apple, “Activar VoiceOver y practicar gestos en el iPhone”, *Manual de uso del iPhone*. <https://support.apple.com/es-es/guide/iphone/iph3e2e415f/ios> (consultado el 12 de junio de 2023).
- [21] E. 'Chiu, K. 'Lenzo, y G. 'Swecker, “Giving our characters voices”, *Duolingo Blog*, el 23 de agosto de 2021. <https://blog.duolingo.com/character-voices/> (consultado el 13 de junio de 2023).
- [22] ACX, “ACX Audio Submission Requirements”. <https://www.acx.com/help/acx-audio-submission-requirements/201456300> (consultado el 13 de junio de 2023).
- [23] J. 'Porter, “Apple Books quietly launches AI-narrated audiobooks”, *The Verge*, el 5 de enero de 2023. <https://www.theverge.com/2023/1/5/23540261/apple-text-to-speech-audiobooks-ebooks-artificial-intelligence-narrator-madison-jackson> (consultado el 13 de junio de 2023).
- [24] Google, “Auto-narrated audiobooks”. <https://play.google.com/books/publish/autonarrated/> (consultado el 13 de junio de 2023).
- [25] W. 'Gendron, “That podcast ad you’re listening to may soon be AI. Spotify is reportedly developing bots to mimic your favorite hosts.”, *Business Insider*, el 22 de mayo de 2023. Consultado: el 13 de junio de 2023. [En línea]. Disponible en: <https://www.businessinsider.com/spotify-developing-ai-bots-ads-based-on-podcast-hosts-report-2023-5>
- [26] Amazon Web Services, “Voices in Amazon Polly”. <https://docs.aws.amazon.com/polly/latest/dg/voicelist.html> (consultado el 13 de junio de 2023).



- [27] Google, “Idiomas y voces compatibles”. <https://cloud.google.com/text-to-speech/docs/voices?hl=es-419> (consultado el 13 de junio de 2023).
- [28] Google, “Text-to-Speech”. <https://cloud.google.com/text-to-speech?hl=es> (consultado el 13 de junio de 2023).
- [29] V. Pratap *et al.*, “Scaling Speech Technology to 1,000+ Languages”, *arXiv preprint arXiv:2305.13516*, 2023.
- [30] “MMS - Language Coverage”, *Meta AI*, el 22 de mayo de 2023. [https://dl.fbaipublicfiles.com/mms/misc/language\\_coverage\\_mms.html](https://dl.fbaipublicfiles.com/mms/misc/language_coverage_mms.html) (consultado el 13 de junio de 2023).
- [31] J. D. 'Cano, “Meta mejora conversión de voz a texto para más 1.100 idiomas, incluyendo dialectos”, *El Tiempo*, may 2023, Consultado: el 13 de junio de 2023. [En línea]. Disponible en: <https://www.eltiempo.com/tecnosfera/novedades-tecnologia/meta-crea-conversion-de-voz-a-texto-para-mas-1-100-idiommas-770862>
- [32] T. M. Mitchell y T. M. Mitchell, *Machine learning*, vol. 1, núm. 9. McGraw-hill New York, 1997.
- [33] S. 'Yee y T. 'Chu, “A visual introduction to machine learning”, *R2D3*. <http://www.r2d3.us/visual-intro-to-machine-learning-part-1/> (consultado el 19 de mayo de 2023).
- [34] Z.-H. Zhou, *Machine learning*. Springer Nature, 2021.
- [35] Y. Bengio, A. Courville, y P. Vincent, “Representation learning: A review and new perspectives”, *IEEE Trans Pattern Anal Mach Intell*, vol. 35, núm. 8, pp. 1798–1828, 2013.
- [36] Y. LeCun, Y. Bengio, y G. Hinton, “Deep learning”, *Nature*, vol. 521, núm. 7553, pp. 436–444, 2015.
- [37] IBM, “¿Qué es Deep Learning?”, *IBM*. <https://www.ibm.com/es-es/topics/deep-learning> (consultado el 17 de mayo de 2023).
- [38] L. Hardesty, “MIT News Office.‘Explained: Neural Networks.’” MIT News. April, 2017.
- [39] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, N. A. Mohamed, y H. Arshad, “State-of-the-art in artificial neural network applications: A survey”, *Heliyon*, vol. 4, núm. 11, p. e00938, 2018.
- [40] I. Sutskever, O. Vinyals, y Q. V Le, “Sequence to sequence learning with neural networks”, *Adv Neural Inf Process Syst*, vol. 27, 2014.

- [41] R. 'Rodríguez, "Seq2Seq: de secuencia a secuencia".  
<https://lamaquinaoraculo.com/computacion/sec2sec-de-secuencia-a-secuencia/> (consultado el 10 de junio de 2023).
- [42] Mehreen Saeed, "An Introduction To Recurrent Neural Networks And The Math That Powers Them", el 24 de septiembre de 2021.  
<https://machinelearningmastery.com/an-introduction-to-recurrent-neural-networks-and-the-math-that-powers-them/> (consultado el 28 de julio de 2022).
- [43] Christopher Olah, "Understanding LSTM Networks", *Colah's blog*, el 27 de julio de 2015. <http://colah.github.io/posts/2015-08-Understanding-LSTMs/> (consultado el 28 de julio de 2022).
- [44] I. Goodfellow, Y. Bengio, y A. Courville, *Deep learning*. MIT press, 2016.
- [45] L. Tunstall, L. von Werra, y T. Wolf, *Natural language processing with transformers*. " O'Reilly Media, Inc.", 2022.
- [46] A. Vaswani *et al.*, "Attention is all you need", *Adv Neural Inf Process Syst*, vol. 30, 2017.
- [47] Jakob Uszkoreit, "Transformer: A Novel Neural Network Architecture for Language Understanding", *Google AI blog*, el 31 de julio de 2017.  
<https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html> (consultado el 28 de julio de 2022).
- [48] L. Roberts, "Understanding the mel spectrogram-analytics vidhya-medium", *Medium*. url: <https://medium.com/analytics-vidhya/understanding-the-melspectrogram-fca2afa2ce53>, 2020.
- [49] P. Zinemanas, "Herramientas computacionales para el análisis del entorno sonoro urbano", 2019.
- [50] X. Tan, T. Qin, F. Soong, y T.-Y. Liu, "A survey on neural speech synthesis", *arXiv preprint arXiv:2106.15561*, 2021.
- [51] K. Kumar *et al.*, "Melgan: Generative adversarial networks for conditional waveform synthesis", *Adv Neural Inf Process Syst*, vol. 32, 2019.
- [52] W. H. Dudley, "The vocoder", *Bell. Labs. Rec.*, vol. 18, p. 122, 1939.
- [53] E. A. AlBadawy, A. Gibiansky, Q. He, J. Wu, M.-C. Chang, y S. Lyu, "Vocbench: A Neural Vocoder Benchmark for Speech Synthesis", en *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2022, pp. 881–885.
- [54] N. Kalchbrenner *et al.*, "Efficient neural audio synthesis", en *International Conference on Machine Learning*, PMLR, 2018, pp. 2410–2419.

- [55] R. Prenger, R. Valle, y B. Catanzaro, “Waveglow: A flow-based generative network for speech synthesis”, en *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 3617–3621.
- [56] J. Kong, J. Kim, y J. Bae, “Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis”, *Adv Neural Inf Process Syst*, vol. 33, pp. 17022–17033, 2020.
- [57] C. Snijders, U. Matzat, y U.-D. Reips, “‘ Big Data’: big gaps of knowledge in the field of internet science”, *International journal of internet science*, vol. 7, núm. 1, pp. 1–5, 2012.
- [58] G. Eren y The Coqui TTS Team, “Coqui TTS”, el 1 de enero de 2021. <https://www.coqui.ai> (consultado el 27 de julio de 2022).
- [59] J. I. Hualde, *The sounds of Spanish with audio CD*. Cambridge University Press, 2005.
- [60] T. Gopalakrishnan, S. A. Imam, y A. Aggarwal, “Fine Tuning and Comparing Tacotron 2, Deep Voice 3, and FastSpeech 2 TTS Models in a Low Resource Environment”, en *2022 IEEE International Conference on Data Science and Information System (ICDSIS)*, IEEE, 2022, pp. 1–6.