



UNIVERSIDAD DE CHILE

# Inteligencia Artificial Generativa

Let's talk about hype stuff

---

Valentin Barriere // Clemente Henriquez

Universidad de Chile – DCC

Diplomado de Postítulo en Inteligencia Artificial, Primavera 2025

# **Modelos de Audio**

# Outline : Datos de Audio

## Datos de Audio

Introducción

Representaciones

Codificadores de Voz

LLMs de Voz

Benchmarks

Conjuntos de datos de  
pre-entrenamiento

Aplicaciones

# Especificidades del Audio

- Las entradas de voz tienen un número variable de unidades léxicas por secuencia.
- La voz es una secuencia larga que no tiene límites de segmentos.
- La voz es continua sin un diccionario predefinido de unidades para modelar explícitamente en el entorno auto-supervisado.
- Las tareas de procesamiento de voz pueden requerir información ortogonal, ej., ASR e identificación de hablante.

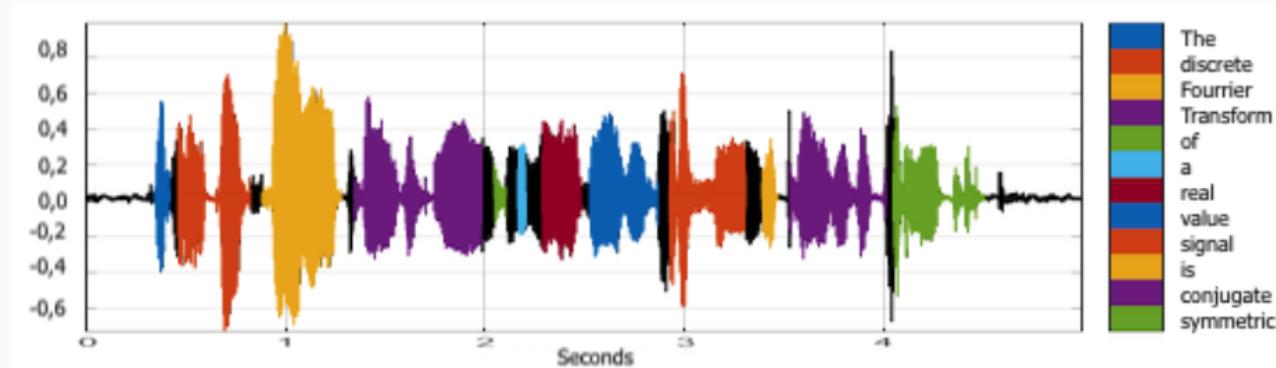
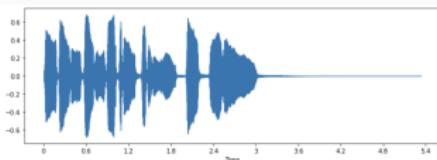
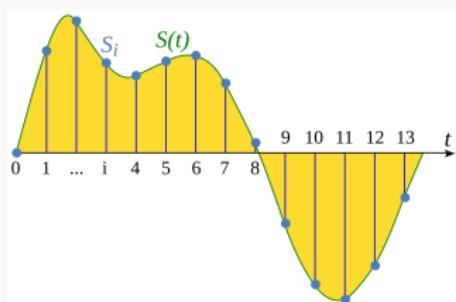


Figure 1: La voz es continua mientras que el texto es discreto

# ¿Qué son los Datos de Audio?

- El sonido es una onda continua — las computadoras lo almacenan como una serie de números (muestras).
- La **frecuencia de muestreo** define cuántas veces por segundo capturamos la señal.
- El arreglo resultante de valores forma una **forma de onda**.
- Cada punto representa amplitud — qué tan "fuerte" es el sonido en un instante dado.
- Esta representación digital permite a los modelos de IA **analizar, generar o entender** el sonido.

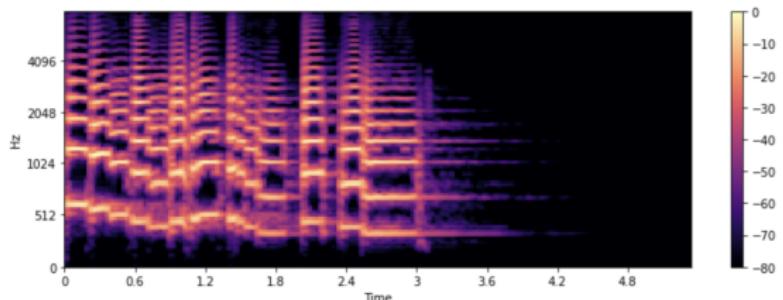


**Figure 3:** Forma de onda: tiempo vs amplitud

# Espectrograma: Forma clásica de entender los Sonidos

## ¿Por qué procesar audio?

- Los modelos no pueden interpretar el sonido crudo directamente — lo convertimos en **características**.
- La vista más común: el **espectrograma** — tiempo en un eje, frecuencia en el otro.
- Un **espectrograma mel** remolda las frecuencias para coincidir con la audición humana.
- Estas representaciones hacen que la voz, la música y los sonidos ambientales sean medibles y aprendibles.



# Tareas Clásicas de Audio

Tarea	Entrada	Salida	Descripción
<b>Clasificación de Audio</b>	Audio	Etiqueta	Clasificar sonidos, géneros musicales, sonidos ambientales
<b>Reconocimiento de Voz</b>	Voz	Texto	Convertir lenguaje hablado a texto escrito
<b>Identificación de Hablante</b>	Audio	ID de Hablante	Identificar qué persona está hablando
<b>Diarización de Hablante</b>	Audio	Segmentos + IDs	"¿Quién habló cuándo?" - identificar hablantes a lo largo del tiempo
<b>Texto a Voz</b>	Texto	Voz	Convertir texto escrito a audio hablado
<b>Conversión de Voz</b>	Audio + Objetivo	Audio	Cambiar características de voz (hablante, emoción)
<b>Generación de Música</b>	Texto/Audio	Audio	Generar música a partir de prompts o continuaciones
<b>Mejora de Audio</b>	Audio ruidoso	Audio limpio	Eliminar ruido, mejorar calidad

## Dos Paradigmas Principales

Comprensión (Audio → Información): Clasificación, ASR, Diarización, Identificación

# Enfoques de Entrada del Modelo: Audio Crudo vs Espectrogramas

## Entrada de Audio Crudo

- Procesamiento directo de forma de onda
- Señal temporal 1D
- Frecuencia de muestreo: 16kHz - 48kHz
- Aprendizaje de características directamente de la señal cruda

**Ejemplos:** HuBERT [13], wav2vec2 [3], wavLM [5], EnCodec [?], ...

## Tendencia

Los modelos modernos usan cada vez más **audio crudo** para aprendizaje extremo a extremo, pero los espectrogramas siguen siendo

## Entrada de Espectrograma

- Representación de frecuencia pre-calculada
- Imagen 2D tiempo-frecuencia
- Escala mel o escala lineal
- Aprovecha técnicas de procesamiento de imágenes

**Ejemplos:** Whisper [20], AST [11], CLAP [23], BYOL-A [12], ...

# Outline : Representaciones

Datos de Audio

**Representaciones**

Codificadores de Voz

LLMs de Voz

Benchmarks

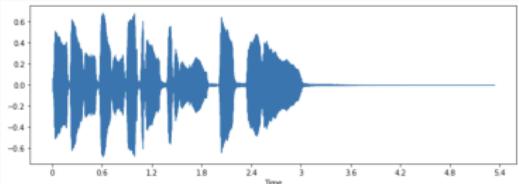
Conjuntos de datos de  
pre-entrenamiento

Aplicaciones

# Modelos de Audio Crudo: Aprendizaje desde la Forma de Onda

## Características Clave

- **Entrada:** Forma de onda cruda (señal 1D)
- **Procesamiento:** CNN 1D
- **Arquitectura:** El codificador aprende representaciones directamente



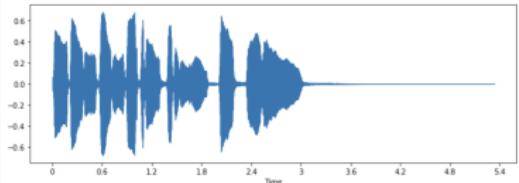
**Figure 4:** Forma de onda de audio crudo

**Ventajas:**

# Modelos de Audio Crudo: Aprendizaje desde la Forma de Onda

## Características Clave

- **Entrada:** Forma de onda cruda (señal 1D)
- **Procesamiento:** CNN 1D
- **Arquitectura:** El codificador aprende representaciones directamente



**Figure 4:** Forma de onda de audio crudo

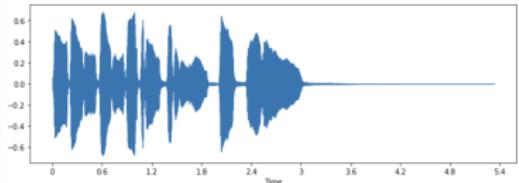
## Ventajas:

- Aprendizaje extremo a extremo

# Modelos de Audio Crudo: Aprendizaje desde la Forma de Onda

## Características Clave

- **Entrada:** Forma de onda cruda (señal 1D)
- **Procesamiento:** CNN 1D
- **Arquitectura:** El codificador aprende representaciones directamente



**Figure 4:** Forma de onda de audio crudo

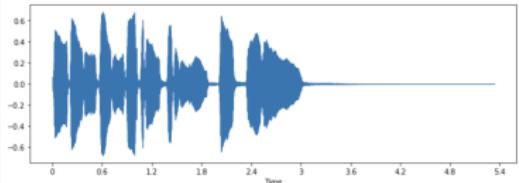
## Ventajas:

- Aprendizaje extremo a extremo
- Sin pérdida de información por extracción de características

# Modelos de Audio Crudo: Aprendizaje desde la Forma de Onda

## Características Clave

- **Entrada:** Forma de onda cruda (señal 1D)
- **Procesamiento:** CNN 1D
- **Arquitectura:** El codificador aprende representaciones directamente



**Figure 4:** Forma de onda de audio crudo

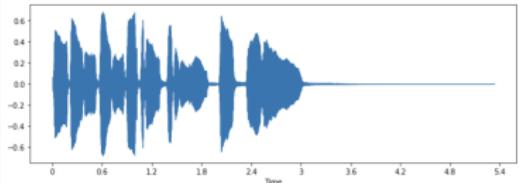
## Ventajas:

- Aprendizaje extremo a extremo
- Sin pérdida de información por extracción de características
- Aprende características óptimas para la tarea

# Modelos de Audio Crudo: Aprendizaje desde la Forma de Onda

## Características Clave

- **Entrada:** Forma de onda cruda (señal 1D)
- **Procesamiento:** CNN 1D
- **Arquitectura:** El codificador aprende representaciones directamente



**Figure 4:** Forma de onda de audio crudo

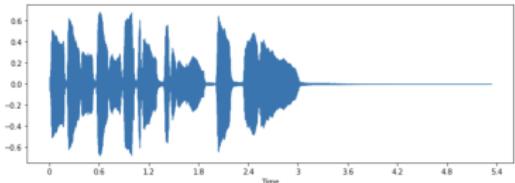
## Ventajas:

- Aprendizaje extremo a extremo
- Sin pérdida de información por extracción de características
- Aprende características óptimas para la tarea
- Funciona con diferentes

# Modelos de Audio Crudo: Aprendizaje desde la Forma de Onda

## Características Clave

- **Entrada:** Forma de onda cruda (señal 1D)
- **Procesamiento:** CNN 1D
- **Arquitectura:** El codificador aprende representaciones directamente



**Figure 4:** Forma de onda de audio crudo

## Ventajas:

- Aprendizaje extremo a extremo
- Sin pérdida de información por extracción de características
- Aprende características óptimas para la tarea
- Funciona con diferentes

## Ejemplo: wav2vec 2.0

- Codificador de características CNN 1D
- Convierte audio de 16kHz a representaciones latentes
- El Transformer procesa estas representaciones

# Modelos Basados en Espectrograma: Representación Visual del Audio

## Características Clave

- **Entrada:** Espectrograma mel (imagen 2D)
- **Procesamiento:** CNN 2D o Vision Transformers
- **Arquitectura:** Trata el audio como una imagen

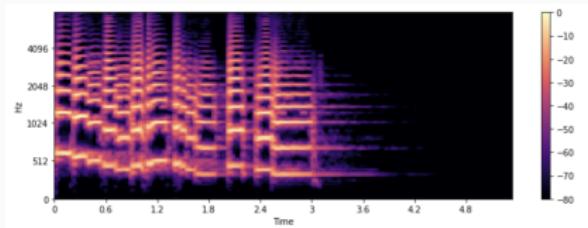


Figure 5: Espectrograma mel

Ventajas:

# Modelos Basados en Espectrograma: Representación Visual del Audio

## Características Clave

- **Entrada:** Espectrograma mel (imagen 2D)
- **Procesamiento:** CNN 2D o Vision Transformers
- **Arquitectura:** Trata el audio como una imagen

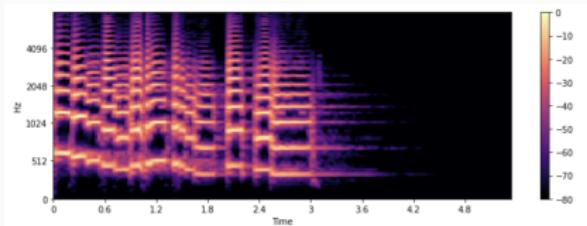


Figure 5: Espectrograma mel

## Ventajas:

- Aprovecha técnicas de visión por computadora

# Modelos Basados en Espectrograma: Representación Visual del Audio

## Características Clave

- **Entrada:** Espectrograma mel (imagen 2D)
- **Procesamiento:** CNN 2D o Vision Transformers
- **Arquitectura:** Trata el audio como una imagen

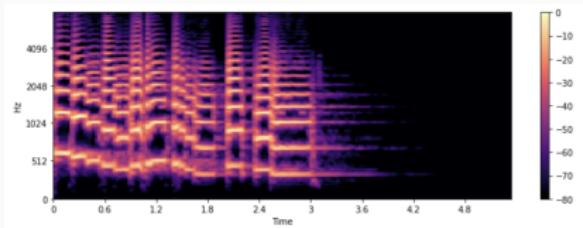


Figure 5: Espectrograma mel

## Ventajas:

- Aprovecha técnicas de visión por computadora
- Patrones tiempo-frecuencia interpretables

# Modelos Basados en Espectrograma: Representación Visual del Audio

## Características Clave

- **Entrada:** Espectrograma mel (imagen 2D)
- **Procesamiento:** CNN 2D o Vision Transformers
- **Arquitectura:** Trata el audio como una imagen

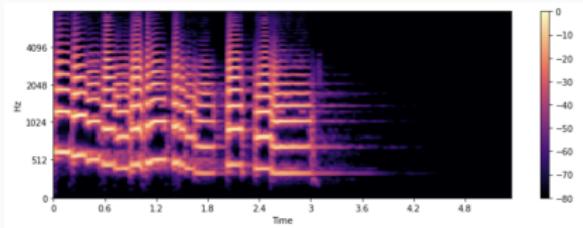


Figure 5: Espectrograma mel

## Ventajas:

- Aprovecha técnicas de visión por computadora
- Patrones tiempo-frecuencia interpretables
- Procesamiento eficiente (representación comprimida)

# Modelos Basados en Espectrograma: Representación Visual del Audio

## Características Clave

- **Entrada:** Espectrograma mel (imagen 2D)
- **Procesamiento:** CNN 2D o Vision Transformers
- **Arquitectura:** Trata el audio como una imagen

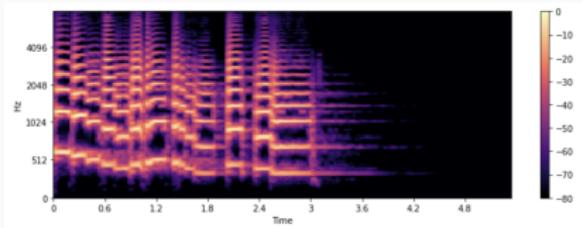


Figure 5: Espectrograma mel

## Ventajas:

- Aprovecha técnicas de visión por computadora
- Patrones tiempo-frecuencia interpretables
- Procesamiento eficiente (representación comprimida)

# Modelos Basados en Espectrograma: Representación Visual del Audio

## Características Clave

- Entrada:** Espectrograma mel (imagen 2D)
- Procesamiento:** CNN 2D o Vision Transformers
- Arquitectura:** Trata el audio como una imagen

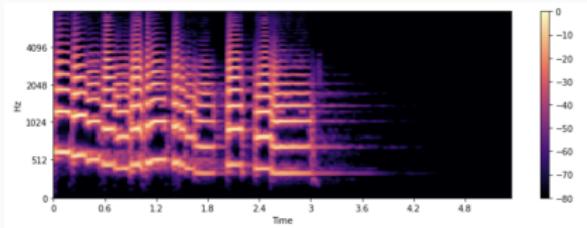


Figure 5: Espectrograma mel

## Ventajas:

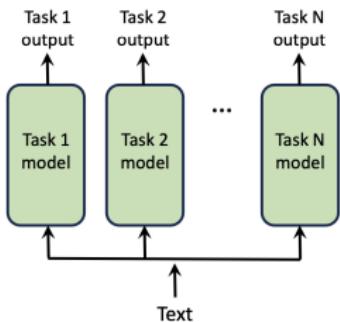
- Aprovecha técnicas de visión por computadora
- Patrones tiempo-frecuencia interpretables
- Procesamiento eficiente (representación comprimida)

## Ejemplo: Whisper

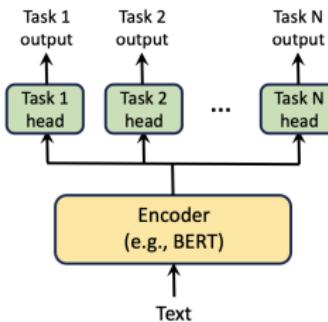
- Espectrograma log-mel (80 canales)
- Convoluciones 2D para extracción de características
- Arquitectura Transformer codificador-decodificador

# Evolución de los modelos fundamentales de texto y voz

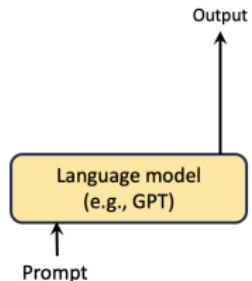
## The task-specific model era (- 2018)



## The encoder era (2018 - 2022)



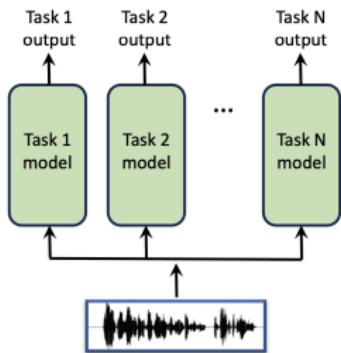
## The large language model era (2022 -)



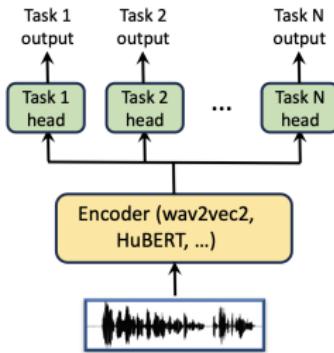
More task-universality, less human effort

# Evolución de los modelos fundamentales de texto y voz

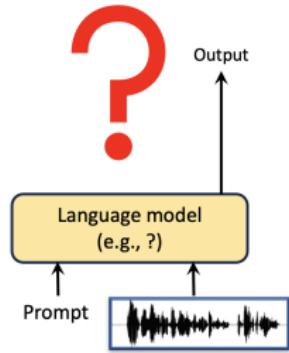
## The task-specific model era (- 2020)



## The speech encoder era (2020 -)

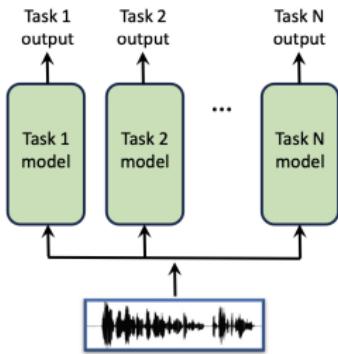


## The spoken large language model era (2024? -)

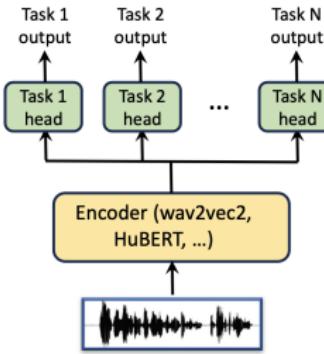


# Evolución de los modelos fundamentales de texto y voz

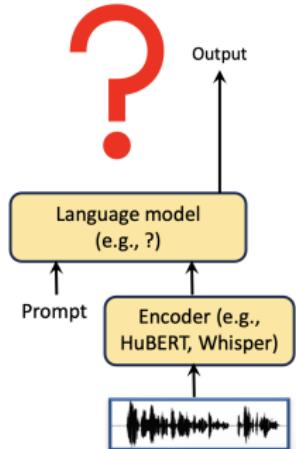
The task-specific model era (- 2020)



The speech encoder era (2020 -)



The spoken large language model era (2024? -)



# Outline : Codificadores de Voz

Datos de Audio

Representaciones

**Codificadores de Voz**

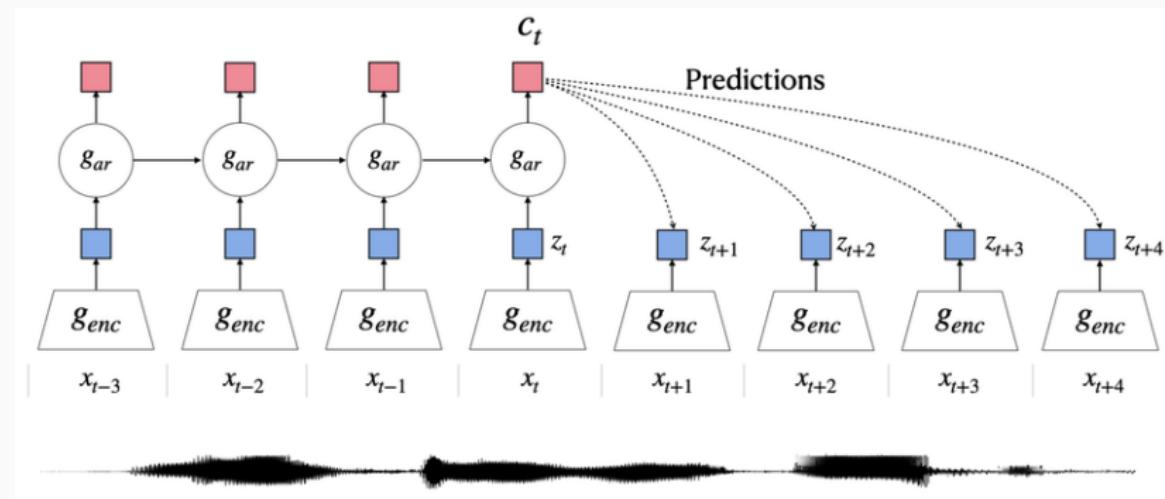
LLMs de Voz

Benchmarks

Conjuntos de datos de  
pre-entrenamiento

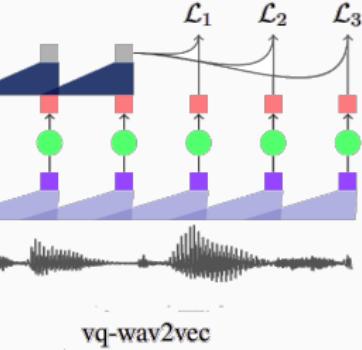
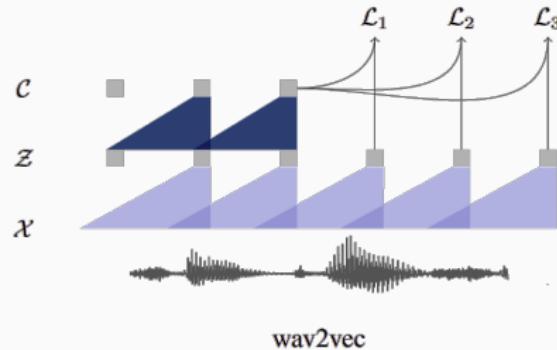
Aplicaciones

# wav2vec [?]



- Pre-entrenamiento no supervisado para representación de audio
- Usa un codificador CNN, luego predice los siguientes estados ocultos
- Basado en pérdida InfoNCE

## vq-wav2vec [2]



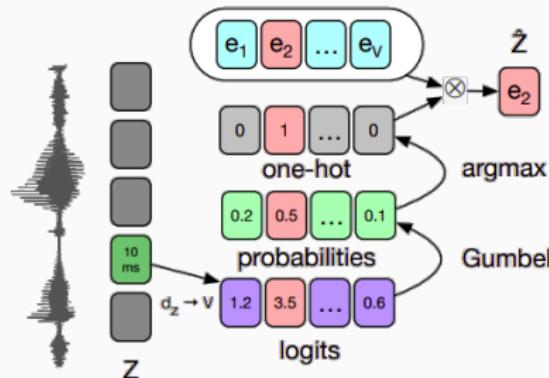
- Igual que wav2vec, pero procesando una cuantización del estado oculto.
- Usando la pérdida InfoNCE (Codificación Predictiva Contrastiva) como wav2vec y word2vec

## vq-wav2vec [2] Cuantización de estados ocultos

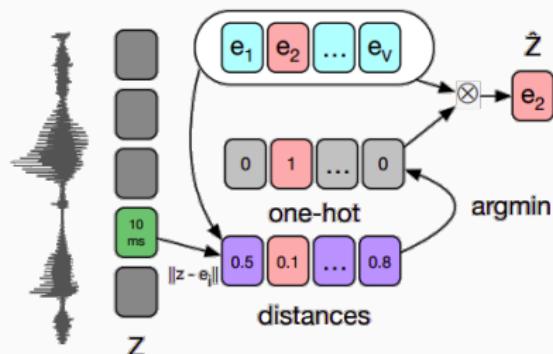
El módulo de cuantización reemplaza la representación continua original  $\mathcal{Z}$  por una representación discreta de tamaño fijo  $\hat{\mathcal{Z}} = e$ ; donde el **libro de códigos**  $e \in \mathbb{R}^{V \times d}$  contiene  $V$  representaciones de tamaño  $d$ .

## vq-wav2vec [2] Cuantización de estados ocultos

El módulo de cuantización reemplaza la representación continua original  $\mathcal{Z}$  por una representación discreta de tamaño fijo  $\hat{\mathcal{Z}} = e_i$  donde el **libro de códigos**  $e \in \mathbb{R}^{V \times d}$  contiene  $V$  representaciones de tamaño  $d$ .



(a) Gumbel-Softmax

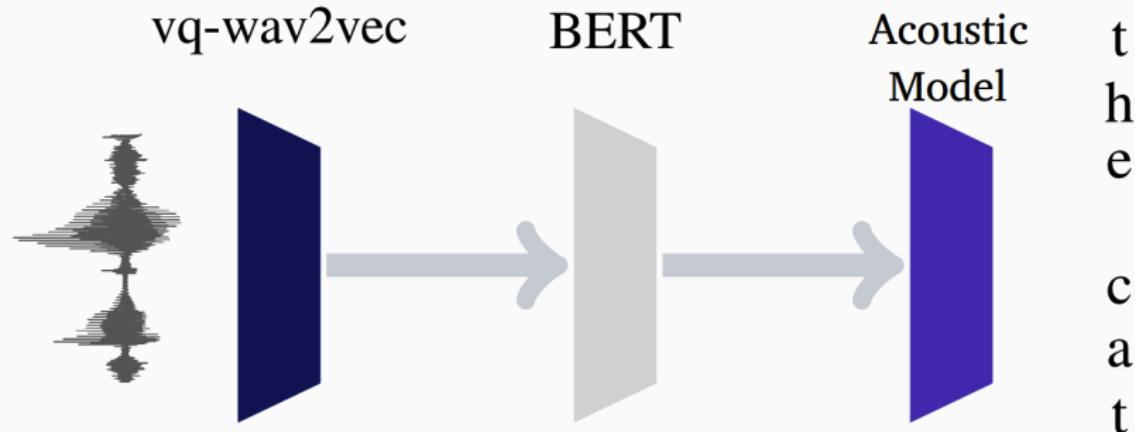


(b) K-means clustering.

Dos técnicas para pasar de vectores densos a cuantizados:

- **Gumbel-Softmax:** una aproximación diferenciable del arg max para calcular representaciones one-hot
- **K-means:** Similar a VQ-VAE [21]:  $||sg(z) - \hat{z}||^2 + \gamma * ||z - sg(\hat{z})||^2$

## vq-wav2vec [2]



La cuantización **hace posible pre-entrenar un transformer usando una arquitectura tipo BERT** y el objetivo MLM usando los valores cuantizados.

Más información [en este blogpost](#).

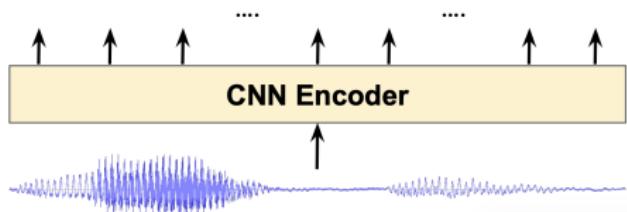
## wav2vec 2.0: [3]

- Predecir cuadros de voz  
enmascarados



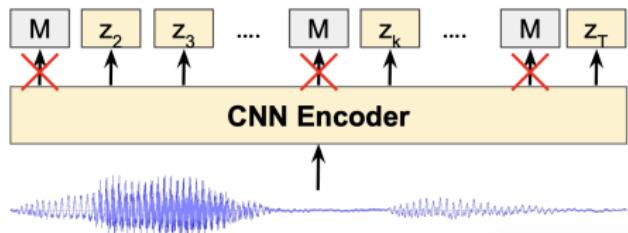
## wav2vec 2.0: [3]

- Predecir cuadros de voz  
enmascarados



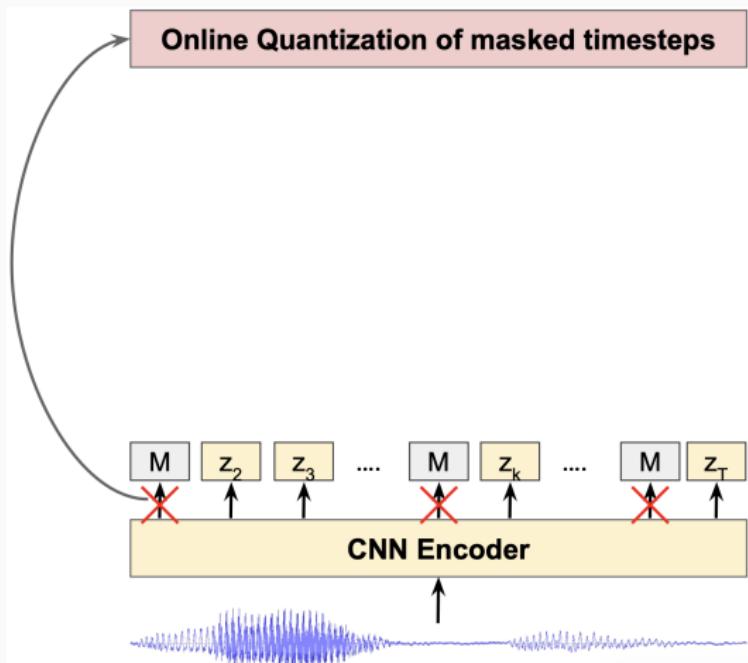
# wav2vec 2.0: [3]

- Predecir cuadros de voz  
enmascarados



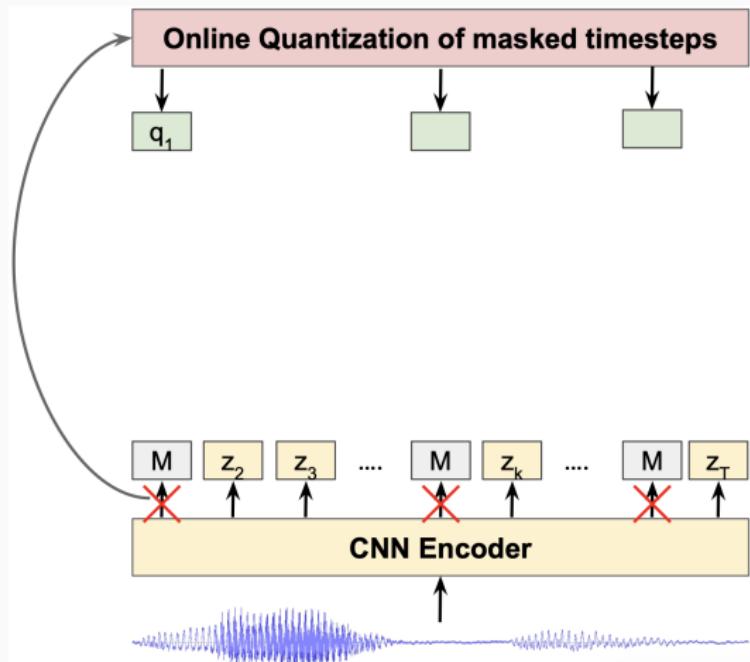
## wav2vec 2.0: [3]

- Predecir cuadros de voz enmascarados



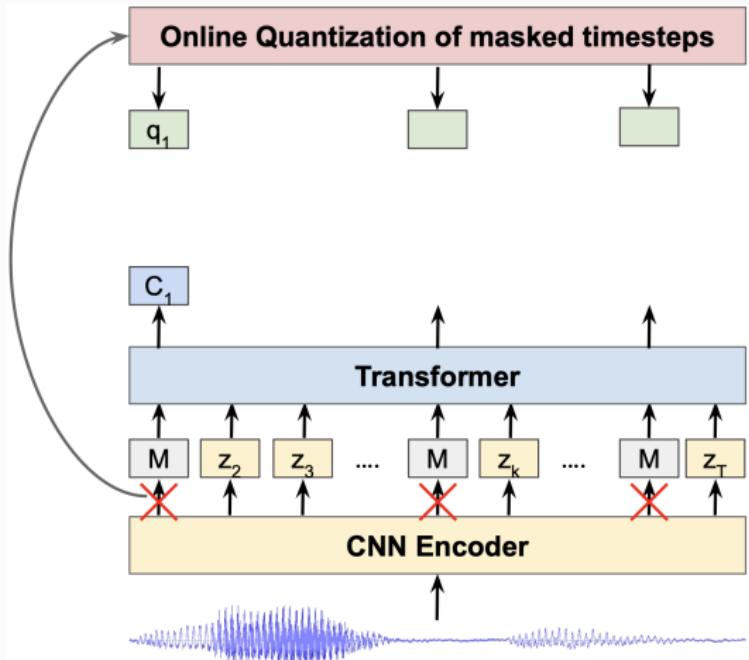
## wav2vec 2.0: [3]

- Predecir cuadros de voz enmascarados



## wav2vec 2.0: [3]

- Predecir cuadros de voz enmascarados
- **Pérdida Contrastiva:**  
Las representaciones de cuadros predichos deben ser similares a las características de entrada cuantizadas en el mismo cuadro
- ...y diferentes de las entradas en cuadros diferentes

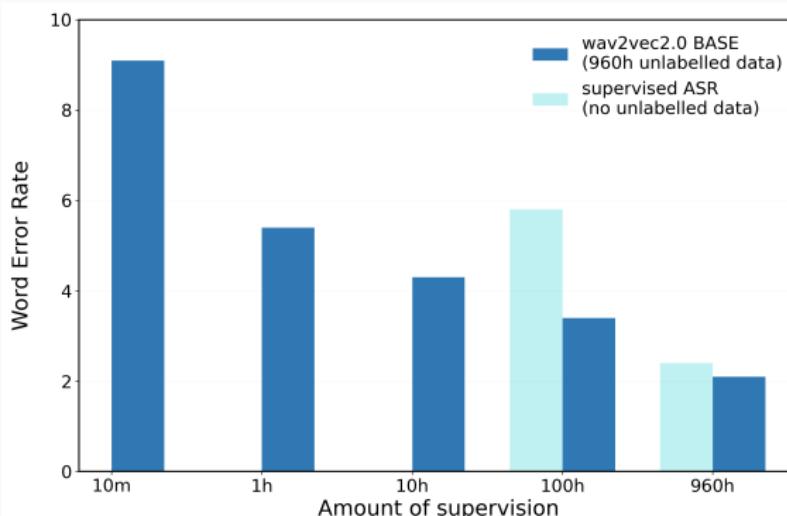


$$\mathcal{L}_m = -\log \frac{\exp(sim(\mathbf{c}_t, \mathbf{q}_t)/\kappa)}{\sum_{\mathbf{q} \sim \mathbf{Q}_t} \exp(sim(\mathbf{c}_t, \mathbf{q})/\kappa)}$$

# wav2vec 2.0: Resultados

## Primeras mejoras importantes en ASR usando aprendizaje auto-supervisado

- Rendimiento mejorado y eficiencia de datos etiquetados en el benchmark LibriSpeech
- Iguala un modelo supervisado usando solo el 1% de los datos etiquetados (100 horas → 1 hora)



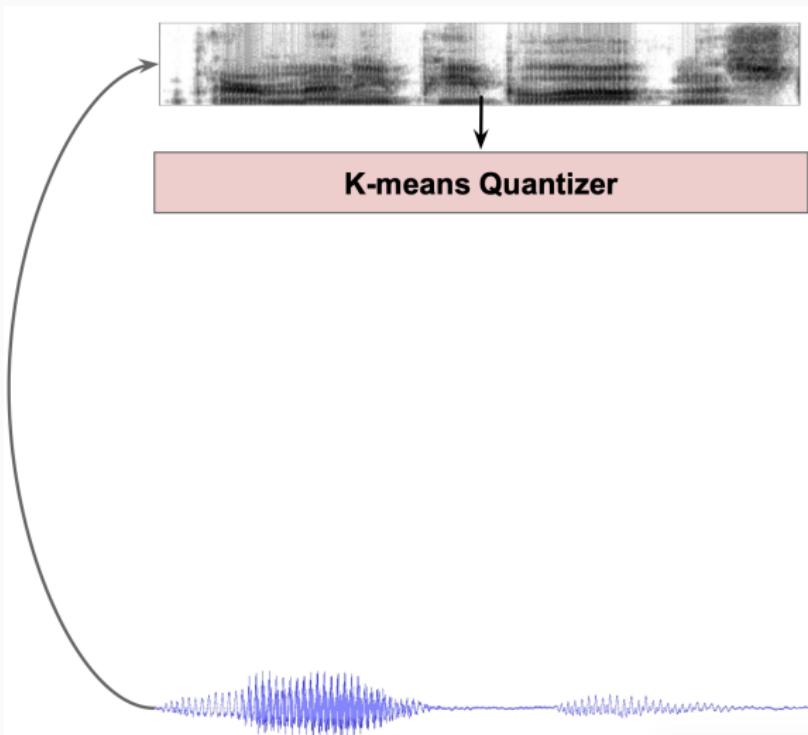
## HuBERT: Hidden-unit BERT [13]

- Un método simple para aplicar aprendizaje de representaciones estilo BERT para voz.
- Igualó o superó el estado del arte en ASR siendo el mejor para muchas tareas de voz.
- Con sus unidades discretas de alta calidad, HuBERT facilitó la investigación de NLP sin texto.

## HuBERT: Hidden-unit BERT [13]

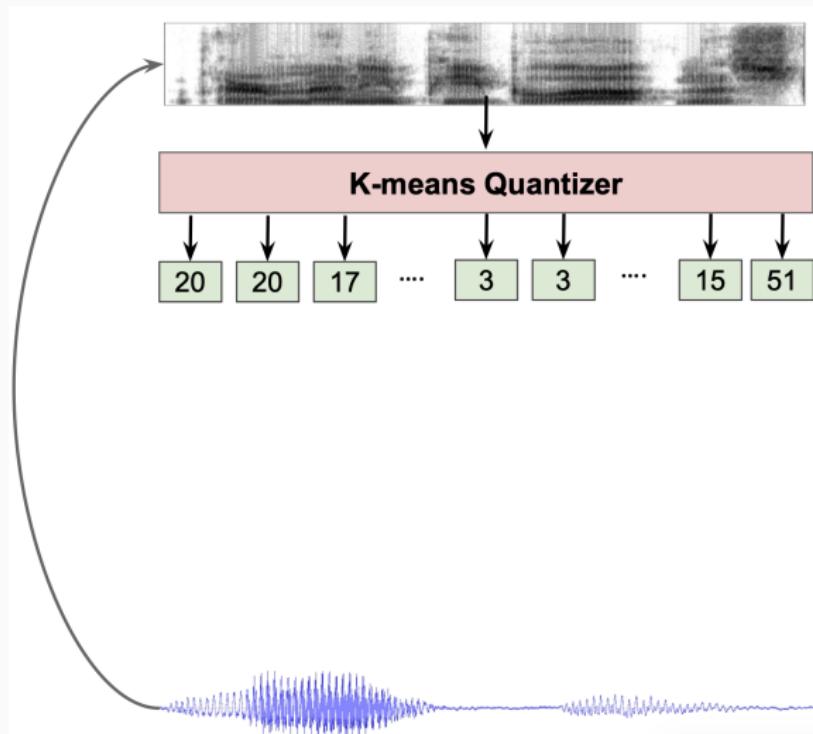


# HuBERT: Hidden-unit BERT [13]



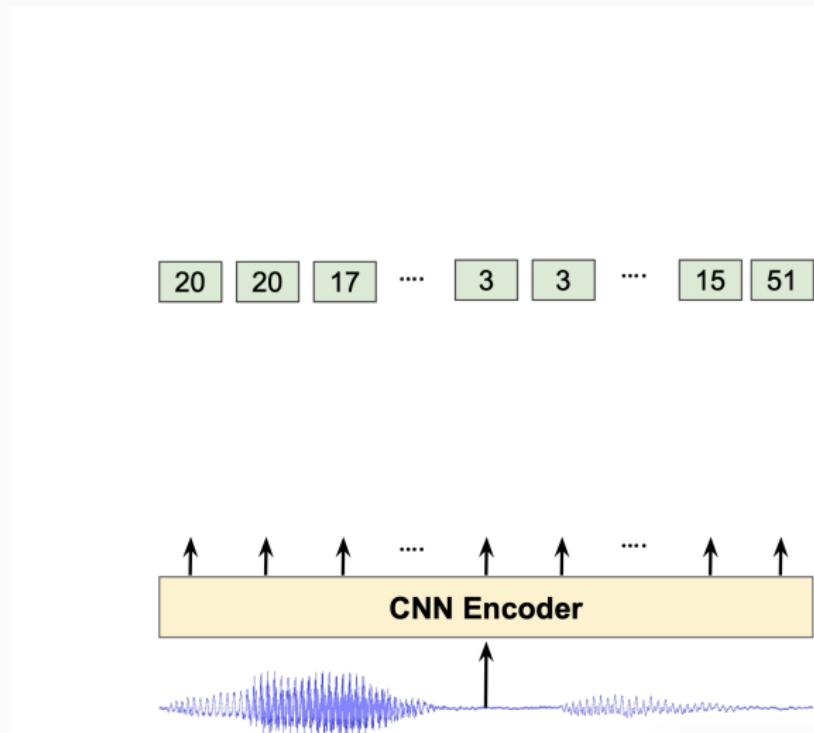
# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce etiquetas a nivel de cuadro.



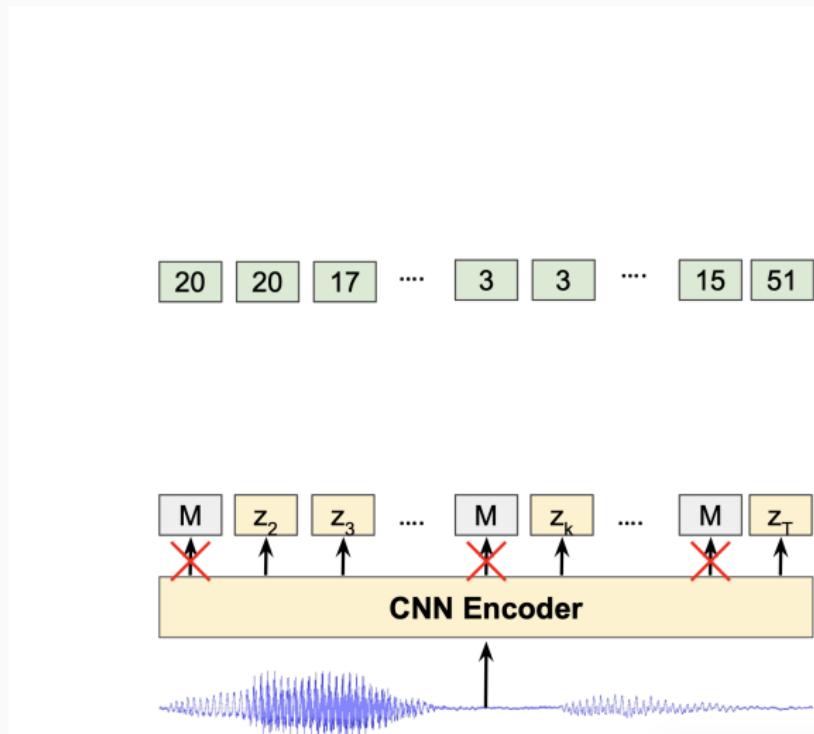
# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce  
etiquetas a nivel de  
cuadro.



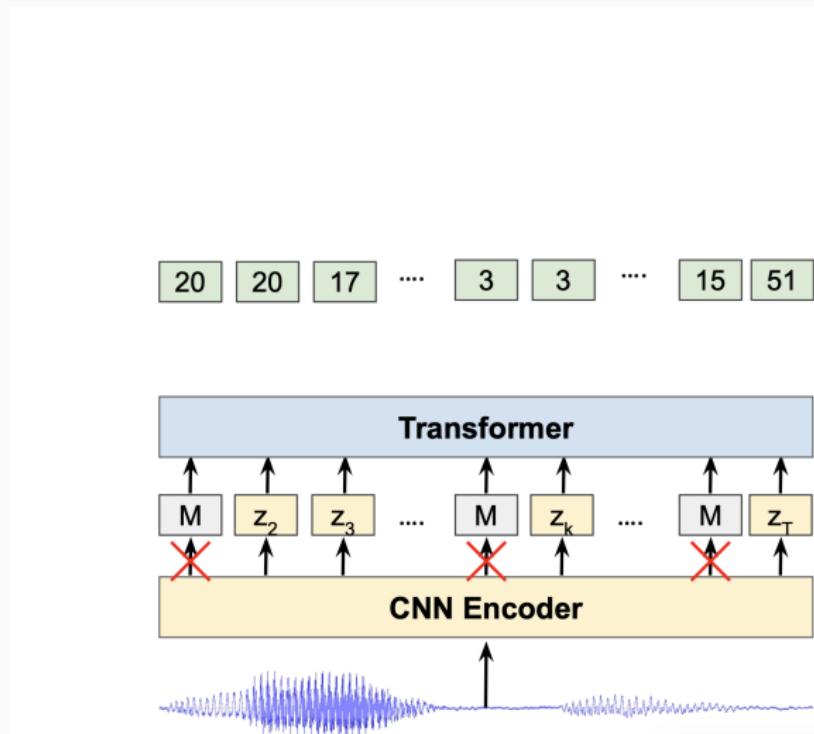
# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce  
etiquetas a nivel de  
cuadro.



# HuBERT: Hidden-unit BERT [13]

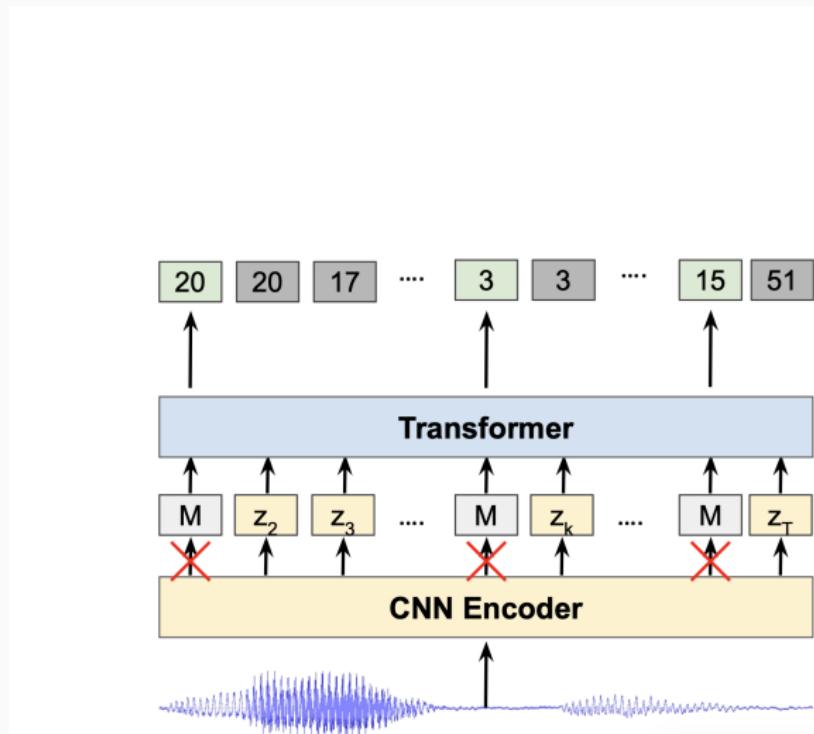
- El cuantizador  
K-means produce etiquetas a nivel de cuadro.



# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce etiquetas a nivel de cuadro.
- Pérdida de predicción enmascarada tipo BERT:

$$\mathcal{L}_m = \sum_{t \in M} -\log p(y_t | X)$$

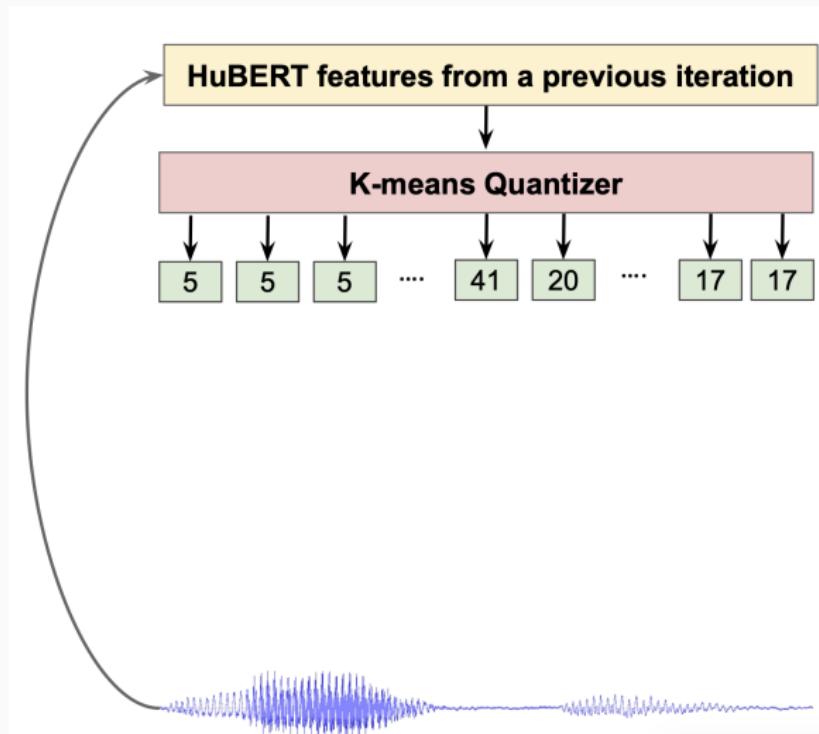


# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce etiquetas a nivel de cuadro.
- Pérdida de predicción enmascarada tipo BERT:

$$\mathcal{L}_m = \sum_{t \in M} -\log p(y_t | X)$$

- La primera iteración usa espectrograma cuantizado, luego características HuBERT de la iteración anterior

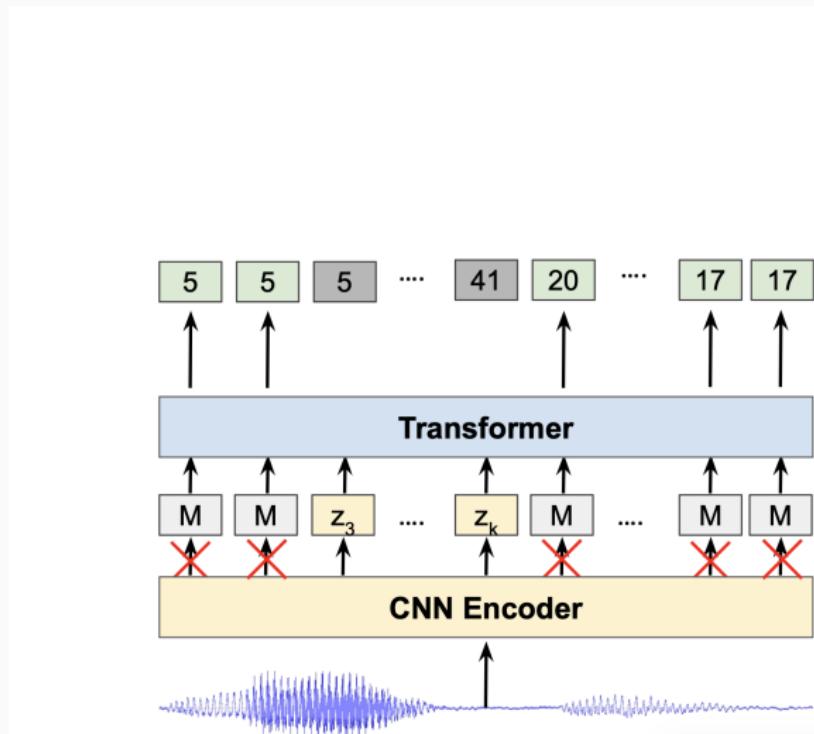


# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce etiquetas a nivel de cuadro.
- Pérdida de predicción enmascarada tipo BERT:

$$\mathcal{L}_m = \sum_{t \in M} -\log p(y_t | X)$$

- La primera iteración usa espectrograma cuantizado, luego características HuBERT de la iteración anterior

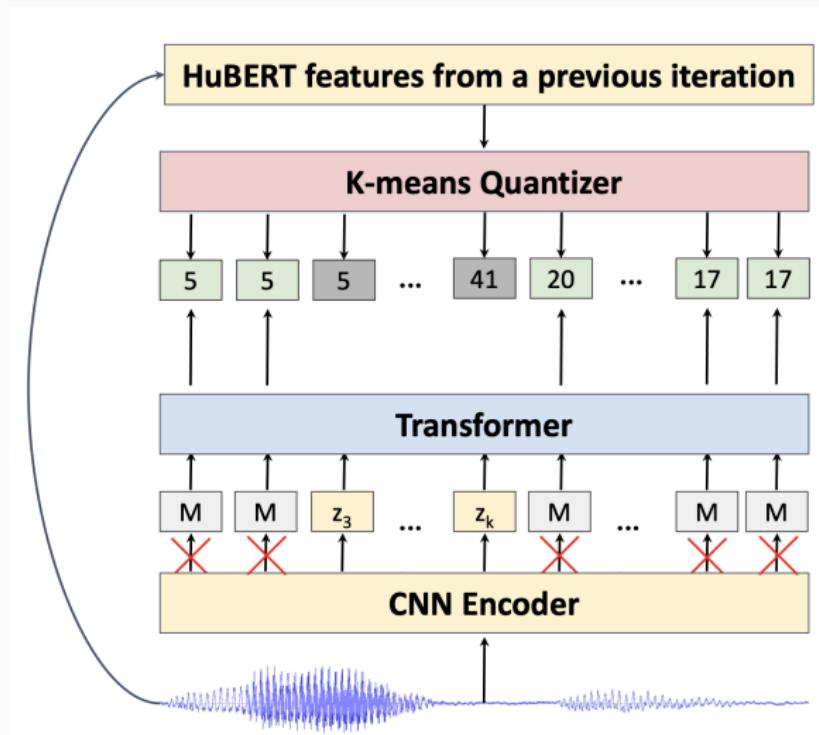


# HuBERT: Hidden-unit BERT [13]

- El cuantizador  
K-means produce etiquetas a nivel de cuadro.
- Pérdida de predicción enmascarada tipo BERT:

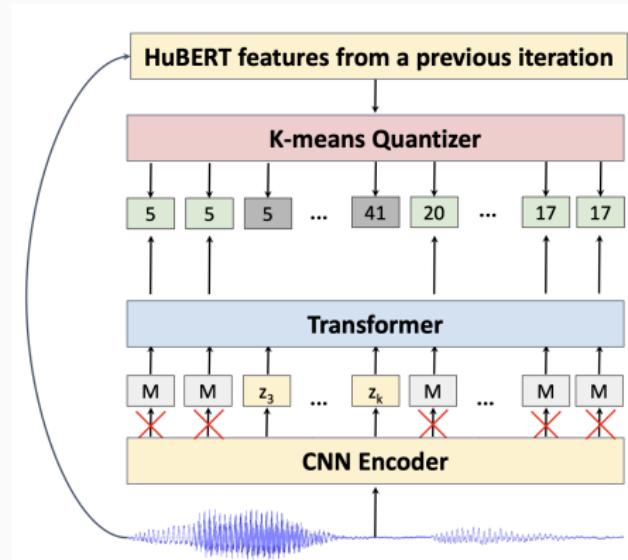
$$\mathcal{L}_m = \sum_{t \in M} -\log p(y_t | X)$$

- La primera iteración usa espectrograma cuantizado, luego características HuBERT de la iteración anterior



# HuBERT: Hidden-unit BERT [13]

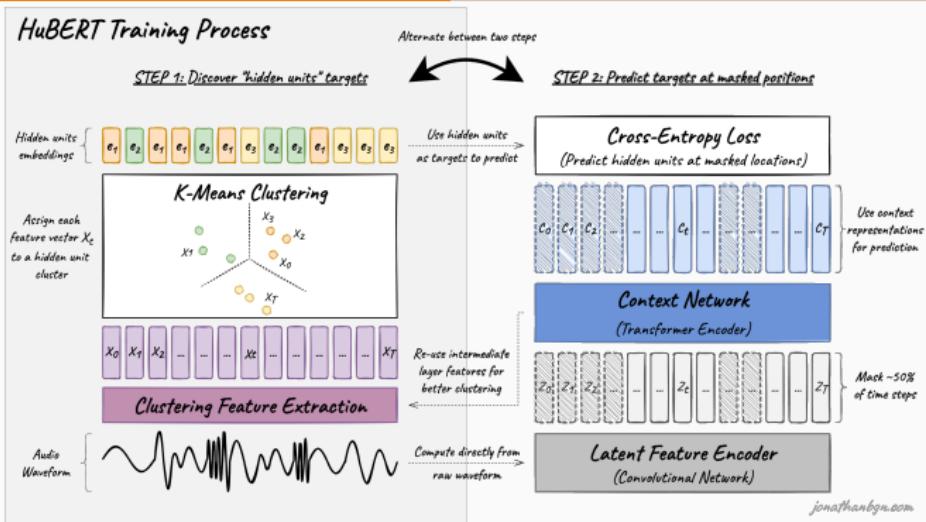
- Un tamaño de libro de códigos pequeño, ej., 50, 100, se usa para la iteración de entrenamiento inicial para enfocarse en diferencias fonéticas en lugar de hablante y estilo.
- Capa 6 para iteración 1, capa 9 para iteración 2 usadas para los pasos de agrupamiento.  
Encontraron empíricamente que contienen características de mayor calidad para muchas tareas de voz.



## Resultados

- Igualó o superó el estado del arte en ASR
- Mejores representaciones para múltiples tareas posteriores: ASR,

# HuBERT: Una Explicación Visual



## Innovación Clave

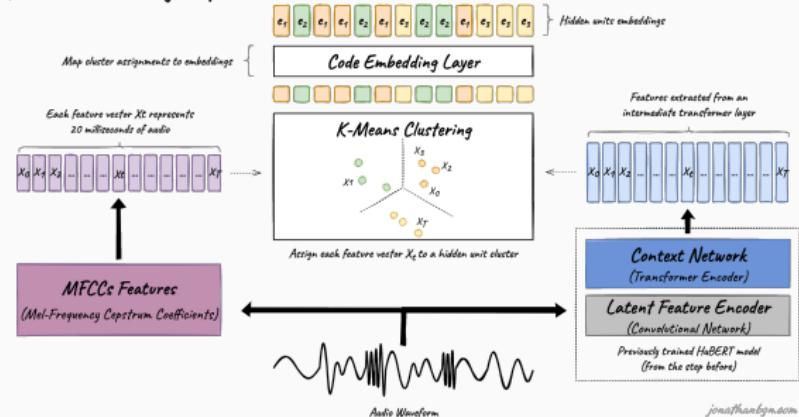
Aprender representaciones significativas de voz **sin datos etiquetados**

- **Problema:** La voz es continua, no discreta como el texto
- **Solución:** Crear unidades discretas mediante agrupamiento
- **Entrenamiento:** Usar predicción enmascarada estilo BERT
- **Resultado:** Representaciones ricas para cualquier tarea de voz

# HuBERT Paso 1: Agrupamiento de Segmentos de Voz

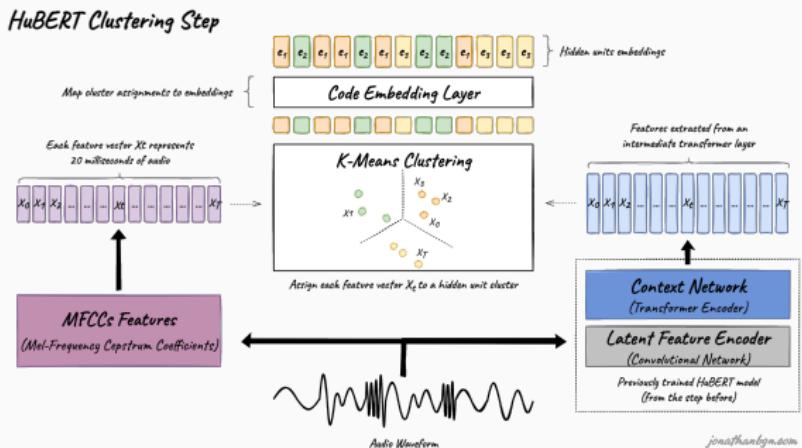
- Audio dividido en segmentos de 25ms

*HubERT Clustering Step*



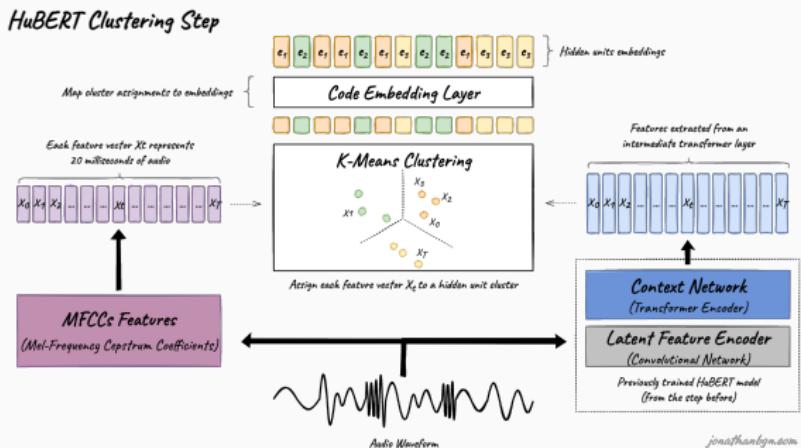
# HuBERT Paso 1: Agrupamiento de Segmentos de Voz

- Audio dividido en segmentos de 25ms
- Extraer características MFCC de cada segmento



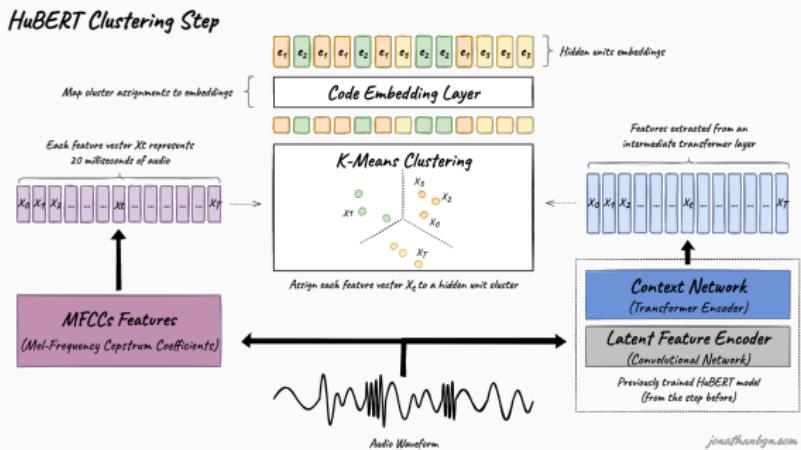
# HuBERT Paso 1: Agrupamiento de Segmentos de Voz

- Audio dividido en segmentos de 25ms
- Extraer características MFCC de cada segmento
- **Agrupamiento K-means** agrupa segmentos similares



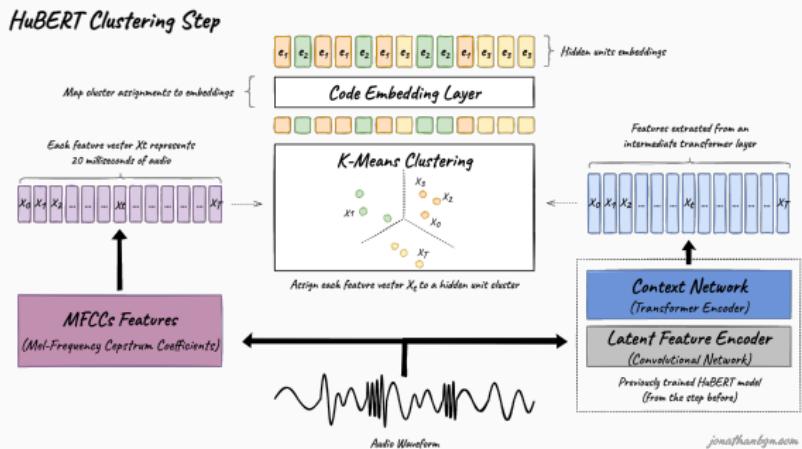
# HuBERT Paso 1: Agrupamiento de Segmentos de Voz

- Audio dividido en segmentos de 25ms
- Extraer características MFCC de cada segmento
- **Agrupamiento K-means** agrupa segmentos similares
- Cada segmento recibe un **ID de cluster**



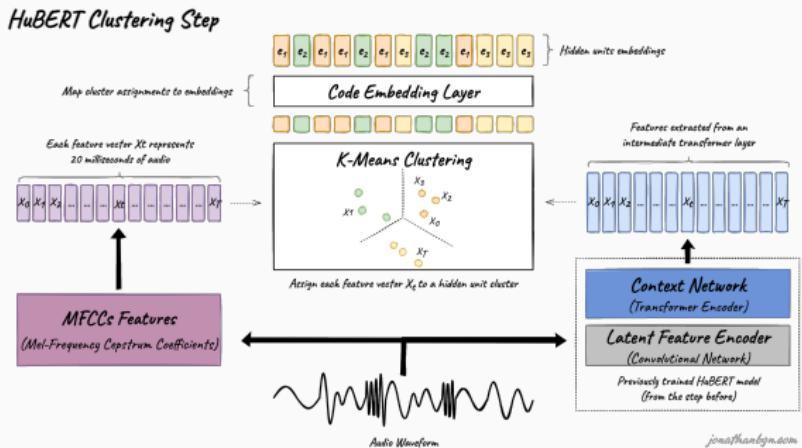
# HuBERT Paso 1: Agrupamiento de Segmentos de Voz

- Audio dividido en segmentos de 25ms
- Extraer características MFCC de cada segmento
- **Agrupamiento K-means** agrupa segmentos similares
- Cada segmento recibe un **ID de cluster**
- Estos IDs se convierten en "pseudo-etiquetas" para entrenamiento



# HuBERT Paso 1: Agrupamiento de Segmentos de Voz

- Audio dividido en segmentos de 25ms
- Extraer características MFCC de cada segmento
- **Agrupamiento K-means** agrupa segmentos similares
- Cada segmento recibe un **ID de cluster**
- Estos IDs se convierten en "pseudo-etiquetas" para entrenamiento



## ¿Por qué Agrupamiento?

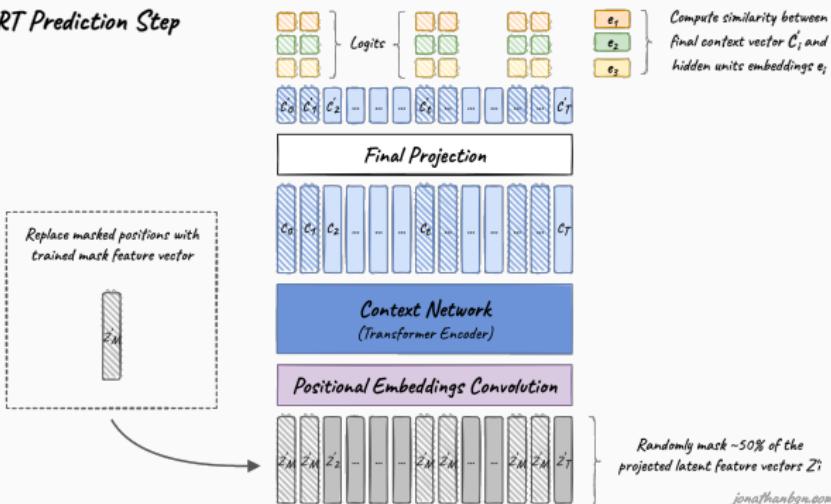
Crea objetivos discretos a partir de audio continuo, permitiendo entrenamiento estilo BERT

# HuBERT Paso 2: Entrenamiento de Predicción Enmascarada

- Enmascarar

aleatoriamente  
~50% de los  
segmentos de audio

*HuBERT Prediction Step*

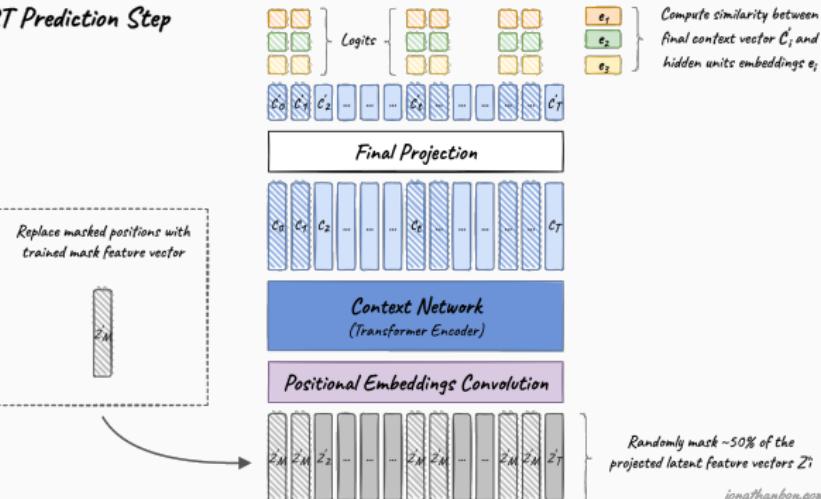


[jonathanbgm.com](http://jonathanbgm.com)

# HuBERT Paso 2: Entrenamiento de Predicción Enmascarada

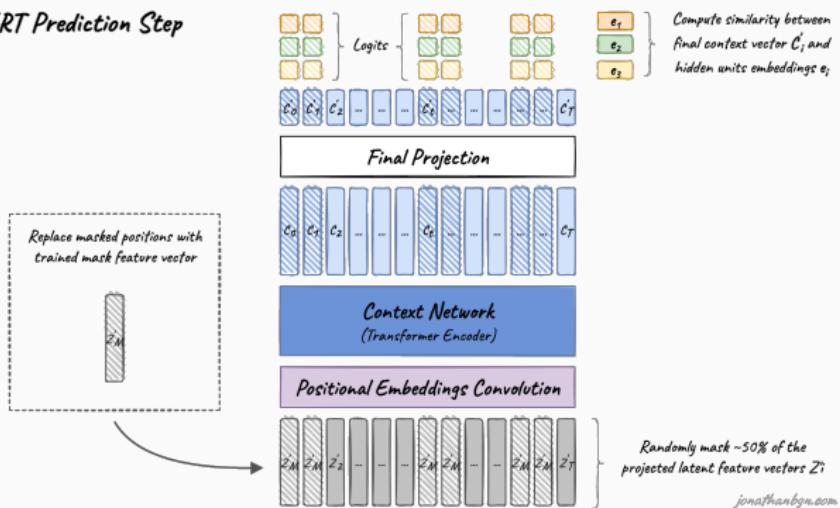
- Enmascarar aleatoriamente ~50% de los segmentos de audio
- El codificador transformer procesa la secuencia

*HuBERT Prediction Step*



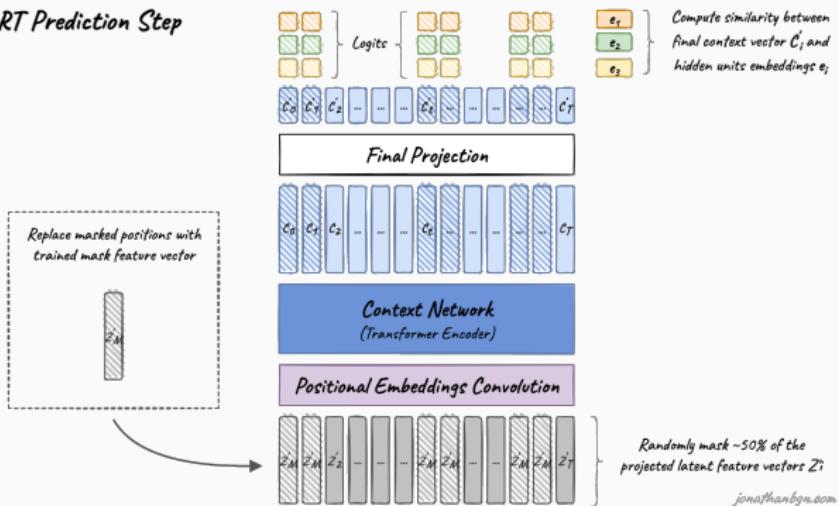
# HuBERT Paso 2: Entrenamiento de Predicción Enmascarada

- Enmascarar aleatoriamente ~50% de los segmentos de audio
- El codificador transformer procesa la secuencia
- El modelo predice **IDs de cluster** de segmentos enmascarados



# HuBERT Paso 2: Entrenamiento de Predicción Enmascarada

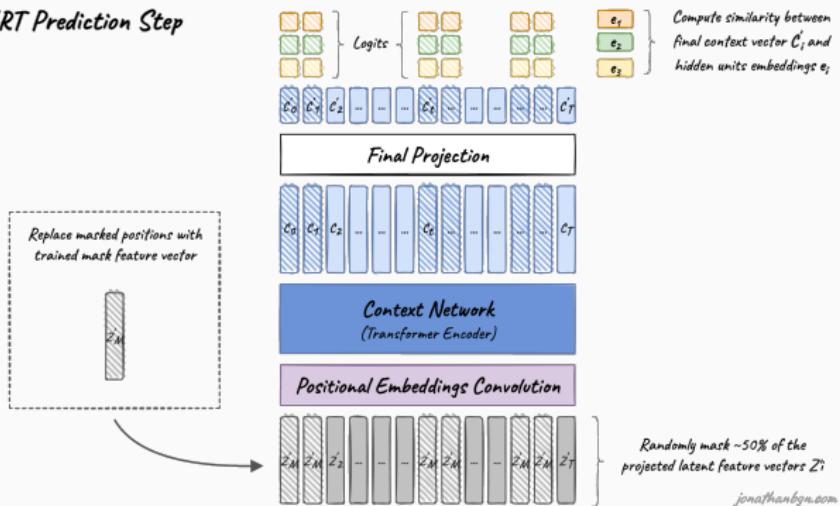
- Enmascarar aleatoriamente ~50% de los segmentos de audio
- El codificador transformer procesa la secuencia
- El modelo predice **IDs de cluster** de segmentos enmascarados
- Usa **pérdida de entropía cruzada**



jonathanbgm.com

# HuBERT Paso 2: Entrenamiento de Predicción Enmascarada

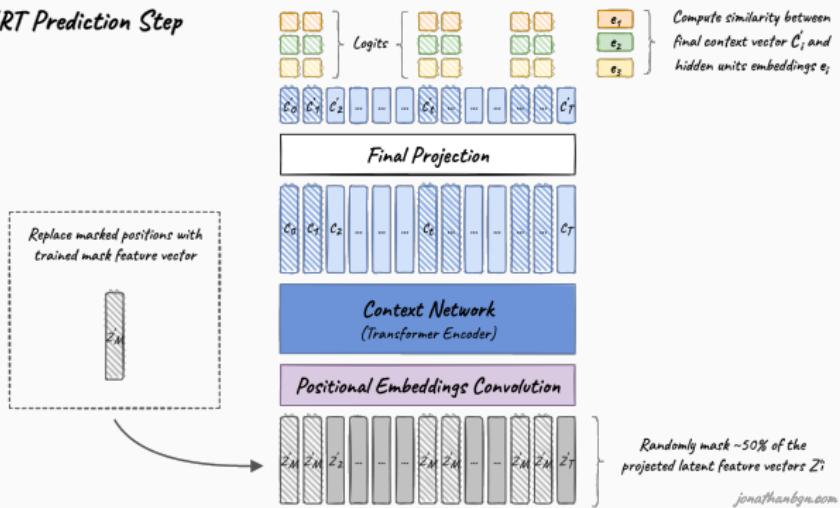
- Enmascarar aleatoriamente ~50% de los segmentos de audio
- El codificador transformer procesa la secuencia
- El modelo predice **IDs de cluster** de segmentos enmascarados
- Usa **pérdida de entropía cruzada**
- El modelo aprende representaciones contextualizadas



jonathanbgm.com

# HuBERT Paso 2: Entrenamiento de Predicción Enmascarada

- Enmascarar aleatoriamente ~50% de los segmentos de audio
- El codificador transformer procesa la secuencia
- El modelo predice **IDs de cluster** de segmentos enmascarados
- Usa **pérdida de entropía cruzada**
- El modelo aprende representaciones contextualizadas



## Diferencia Clave con wav2vec 2.0

Pérdida más simple: predecir objetivos discretos en lugar de aprendizaje contrastivo

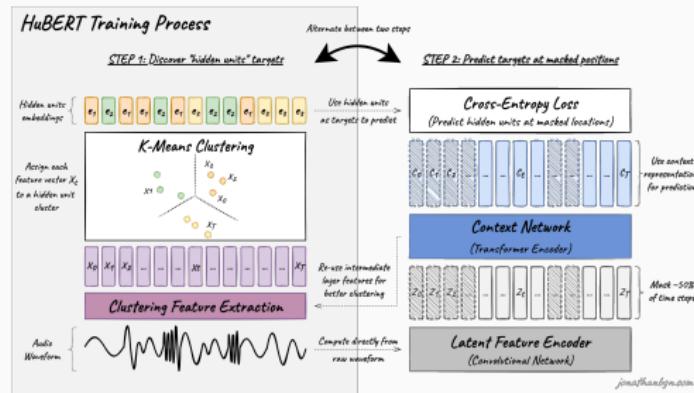
# HuBERT: Refinamiento Iterativo

## Iteración 1

- Agrupar usando **características MFCC**
- Entrenar modelo con estas etiquetas
- Extraer características de la **capa 6**

## Iteración 2

- Re-agrupar usando **características de capa 6**
- Entrenar nuevo modelo con etiquetas refinadas
- Extraer características de la **capa 9**



## ¿Por qué Iterar?

Cada iteración produce características de **mayor calidad** que capturan más información semántica

# HuBERT vs wav2vec 2.0: Diferencias Clave

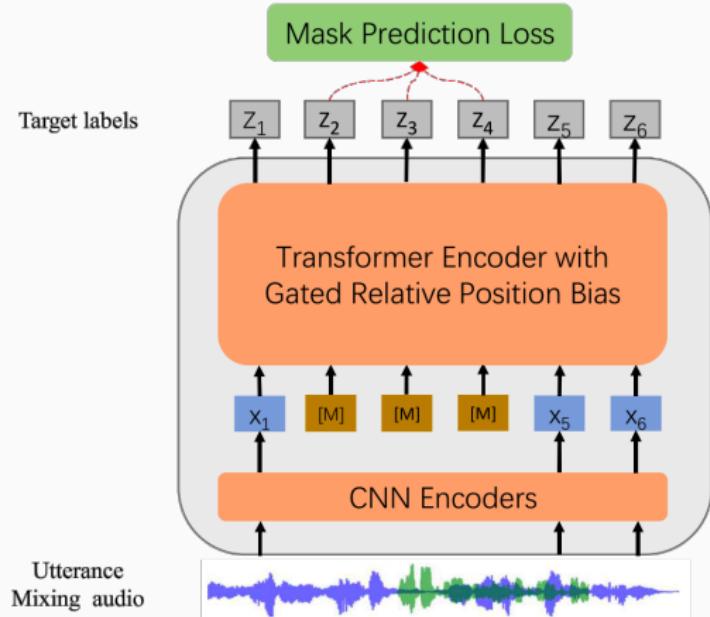
Aspecto	wav2vec 2.0	HuBERT
Objetivo	Latentes cuantizados	IDs de cluster
Pérdida	Contrastiva	Entropía cruzada
Entrenamiento	Pasada única	Refinamiento iterativo
Complejidad	Mayor	Más simple
Rendimiento	Excelente	<b>Mejor en la mayoría de tareas</b>

## Ventajas de HuBERT

- Objetivo de entrenamiento más simple (sin muestreo negativo)
- Mejor transferencia a tareas no-ASR (ID de hablante, emoción, etc.)
- Entrenamiento más estable
- El refinamiento iterativo mejora la calidad

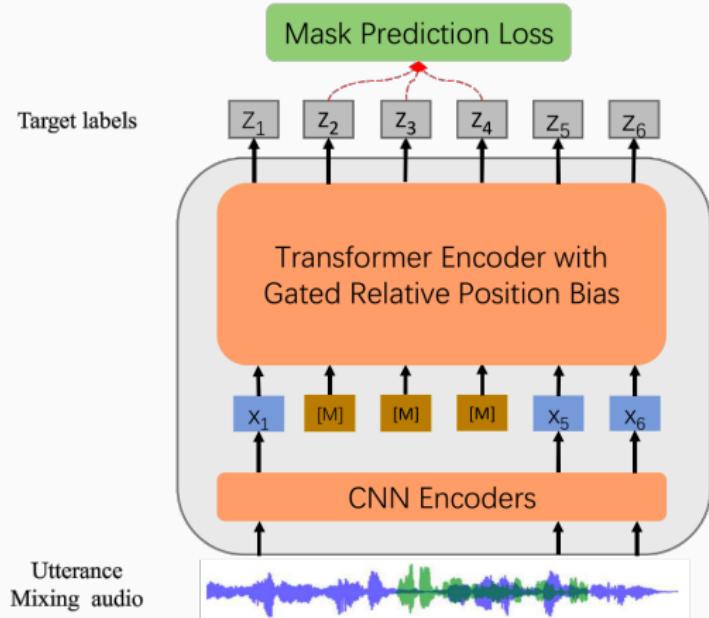
# wavLM [5]

- Igual que HuBERT pero con Ruido



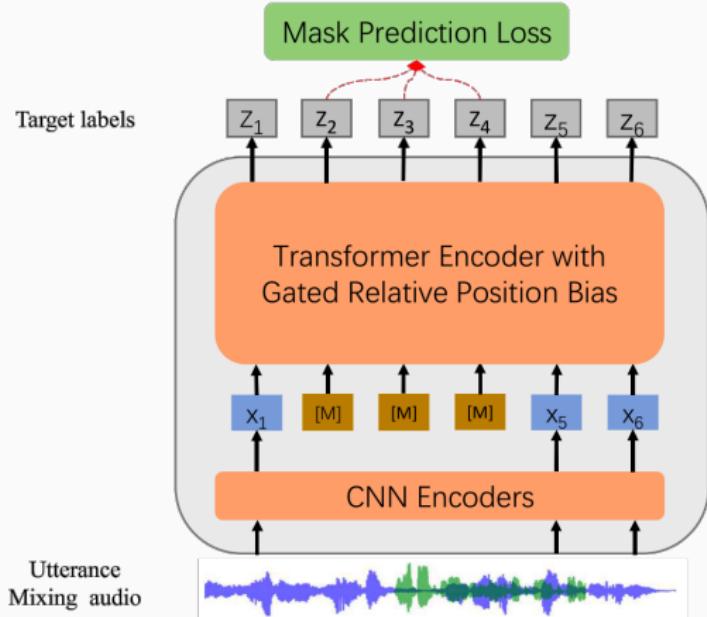
# wavLM [5]

- Igual que HuBERT pero con Ruido
- El modelo necesita encontrar la representación del audio original

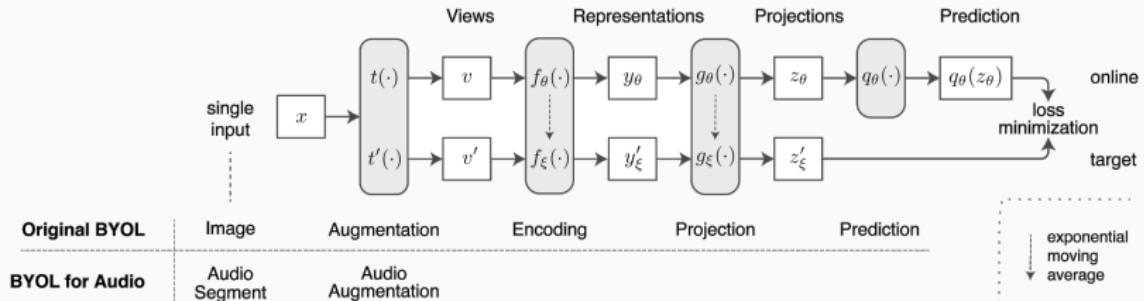


# wavLM [5]

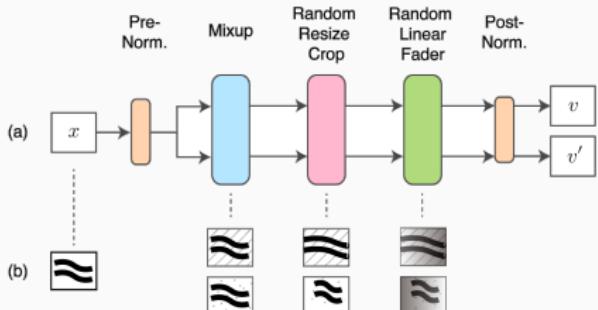
- Igual que HuBERT pero con Ruido
- El modelo necesita encontrar la representación del audio original
- Permite extender modelos de voz pre-entrenados a tareas no-ASR: modela información necesaria para identificación de hablante, separación o diarización



# Bootstrap Your Own Latent - Audio: BYOL-A [12]



- Mismo principio que BYOL [12] pero aplicando la aumentación en los espectrogramas
- CNN simple
- Obtener un vector por sonido
- Muy bueno para sonidos generales:
  - Reconocimiento de Eventos Sonoros
  - Voz No Semántica



# Auto-Codificador Enmascarado de Audio: AudioMAE [16]

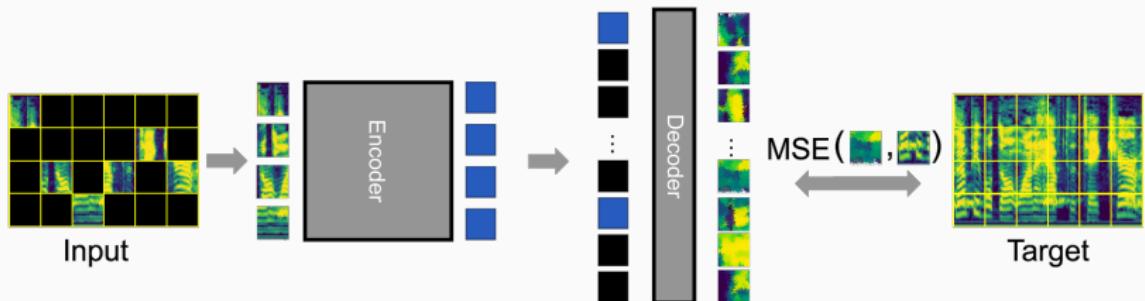


Figure 6: Tan simple como suena

- Aprendizaje auto-supervisado
- El espectrograma se divide en parches
- Enmascarar 80% de los parches
- Restaurar la entrada, minimizando MSE en la porción enmascarada
- Usar un ViT como backbone [8]

# Outline : LLMs de Voz

---

Datos de Audio

Representaciones

Codificadores de Voz

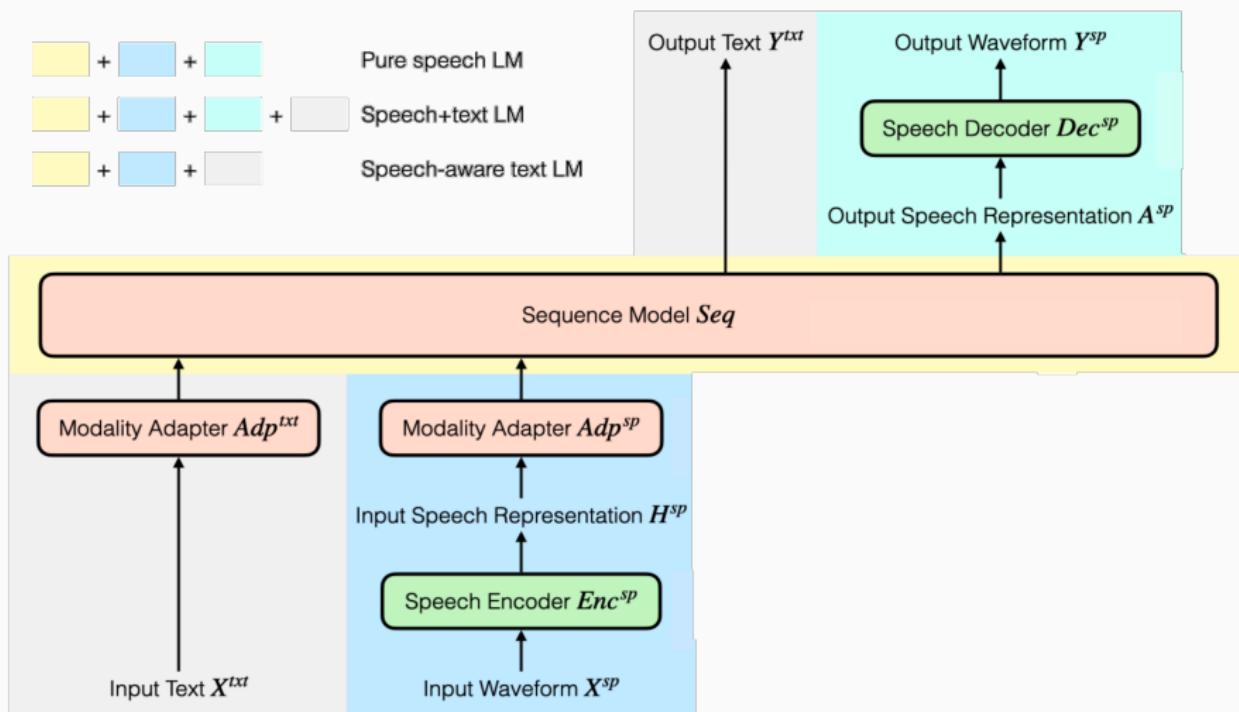
**LLMs de Voz**

Benchmarks

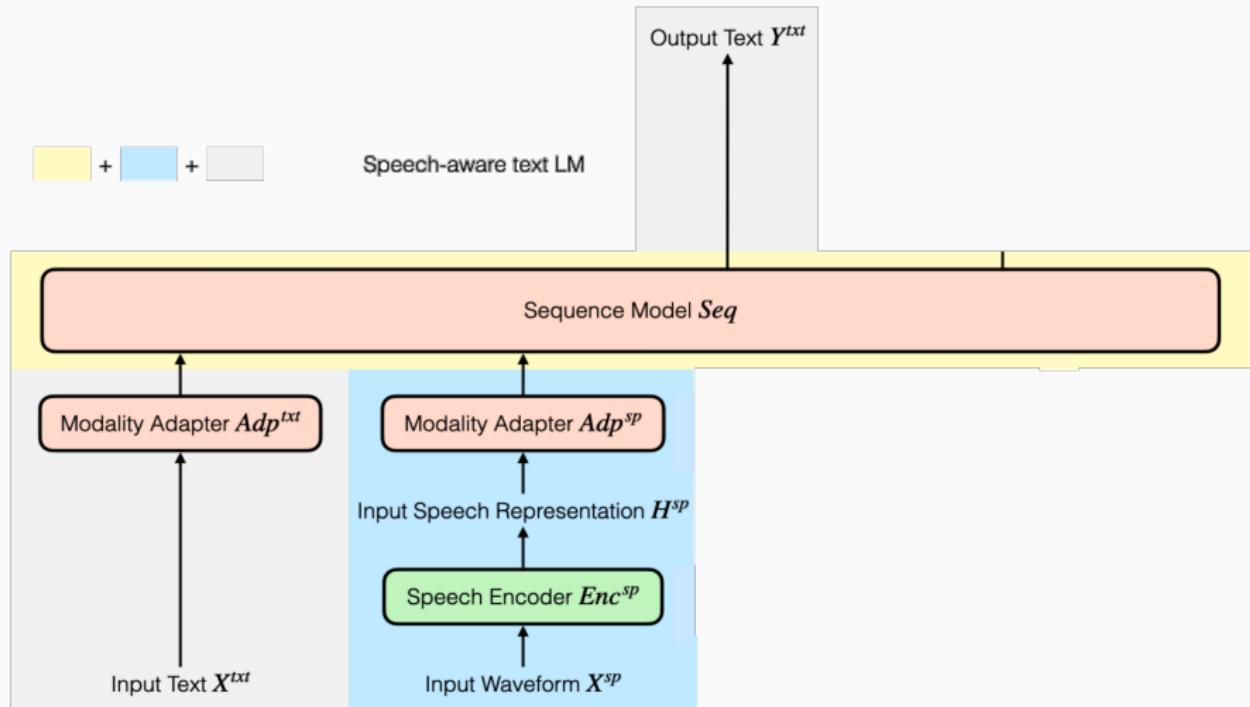
Conjuntos de datos de  
pre-entrenamiento

Aplicaciones

# LLMs de Voz [1]



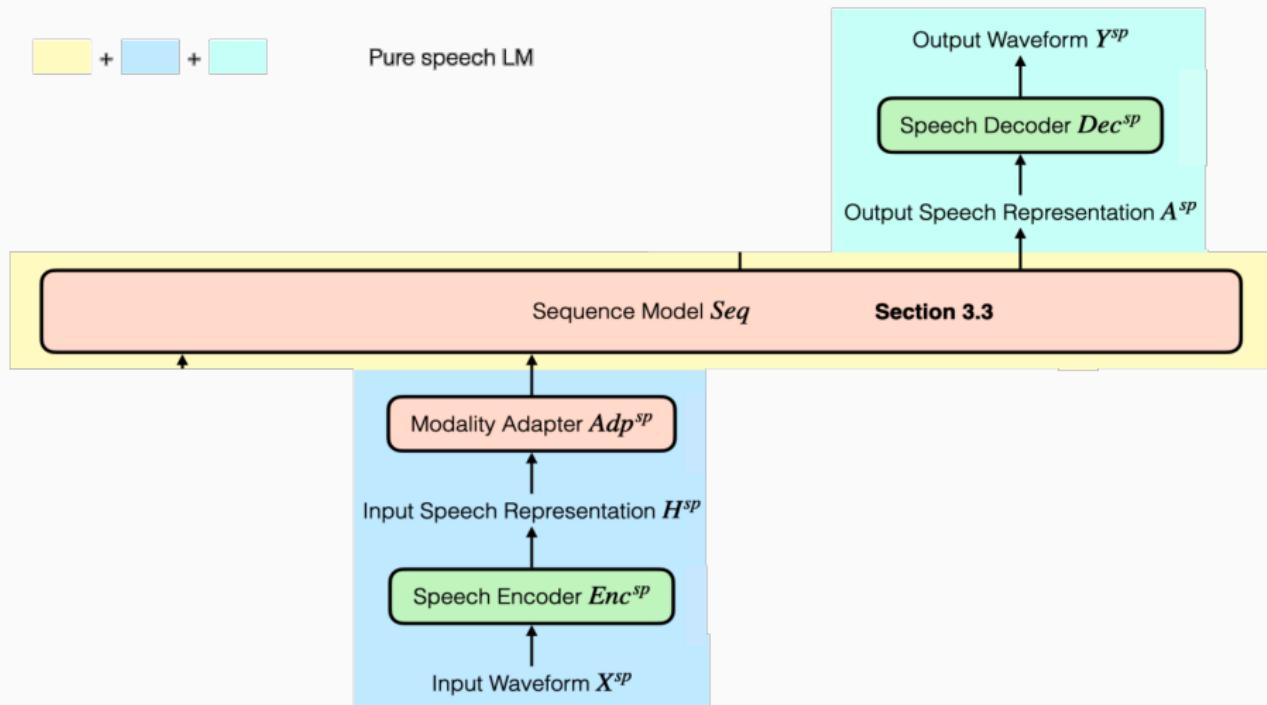
# LLMs de Voz [1]



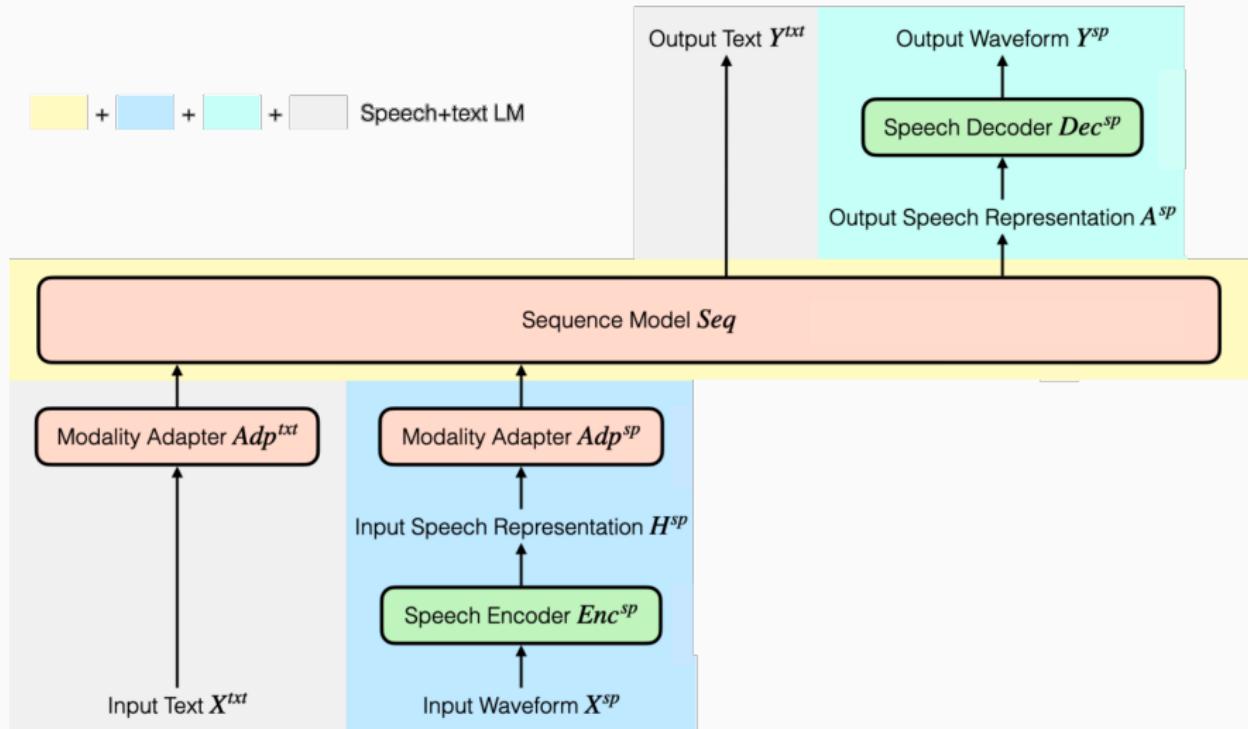
# LLMs de Voz [1]



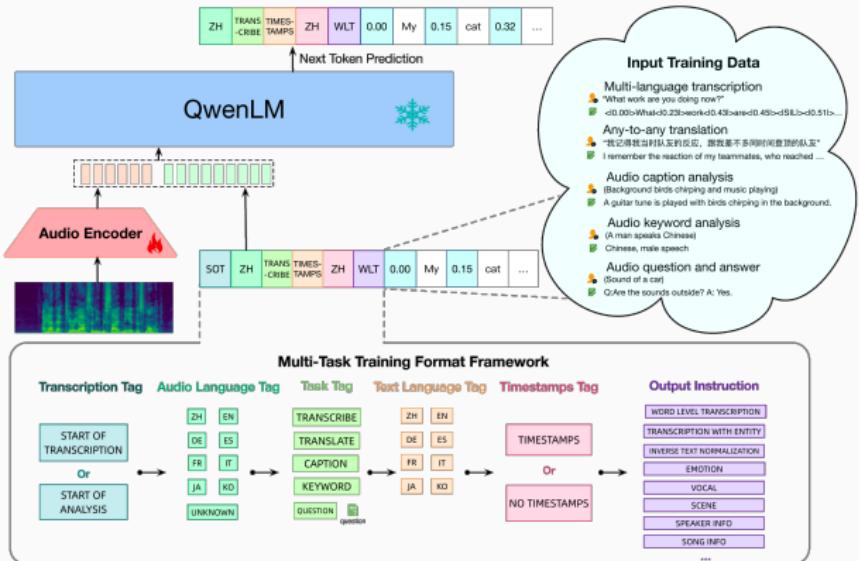
Pure speech LM



# LLMs de Voz [1]



# Qwen(2)-Audio [7, 6]



Types	Task	Description	Hours
Speech	ASR	Automatic speech recognition (multiple languages)	30k
	S2TT	Speech-to-text translation	3.7k
	OSR	Overlapped speech recognition	<1k
	Dialect ASR	Automatic dialect speech recognition	2k
	SRWT	English speech recognition with word-level timestamps	10k
	Mandarin speech recognition with word-level timestamps	Mandarin speech recognition with word-level timestamps	11k
	DID	Dialect identification	2k
	LID	Spoken language identification	11.7k
	SGC	Speaker gender recognition (biologically)	4.8k
Sound	ER	Emotion recognition	<1k
	SV	Speaker verification	1.2k
	SD	Speaker diarization	<1k
	SER	Speech entity recognition	<1k
	KS	Keyword spotting	<1k
	IC	Intent classification	<1k
	SF	Slot filling	<1k
	SAP	Speaker age prediction	4.8k
	VSC	Vocal sound classification	<1k
Music&Song	AAC	Automatic audio caption	8.4k
	SEC	Sound event classification	5.4k
	ASC	Acoustic scene classification	<1k
	SED	Sound event detection with timestamps	<1k
	AQA	Audio question answering	<1k
Music	SID	Singer identification	<1k
	SMER	Singer and music emotion recognition	<1k
	MC	Music caption	25k
	MIC	Music instruments classification	<1k
	MNA	Music note analysis such as pitch, velocity	<1k
	MGR	Music genre recognition	9.5k
	MR	Music recognition	<1k
	MQA	Music question answering	<1k

- Usa los embeddings de Whispev2/3-large [20]
- Pesos pre-entrenados del LLM de Qwen-7B [4]
- Congelar LLM y optimizar codificador de audio: Qwen-Audio, luego congelar el codificador de audio y entrenar el LLM: Qwen-Audio-Chat

# Audio Flamingo [17, 10]

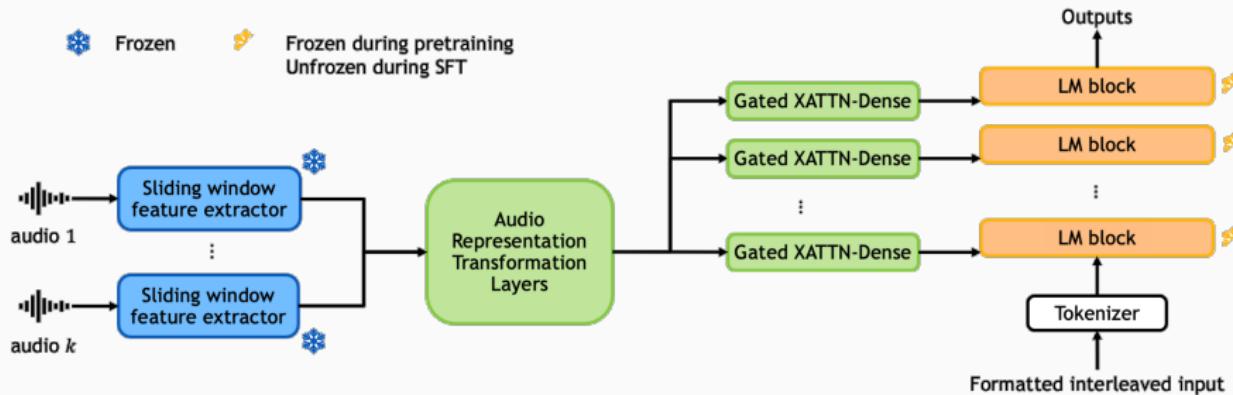


Figure 7: Audio y texto intercalados como entrada y salida de texto libre.

- ClapCap como extractor de características de audio (clips de 7s) y fusionarlos con un pequeño transformer [9]
- **Pre-entrenar:** aprender las capas de transformación de representación de audio y las capas gated xattn-dense → obtener un buen conjunto de pesos de inicialización para estas capas
- **Ajuste Fino:** descongelar todo el LM, y entrenar todos los módulos (excepto ClapCap)

# Outline : Benchmarks

---

Datos de Audio

Representaciones

Codificadores de Voz

LLMs de Voz

Benchmarks

Conjuntos de datos de  
pre-entrenamiento

Aplicaciones

## Speech processing Universal PERformance Benchmark (SUPERB)

<https://superbbenchmark.org/>

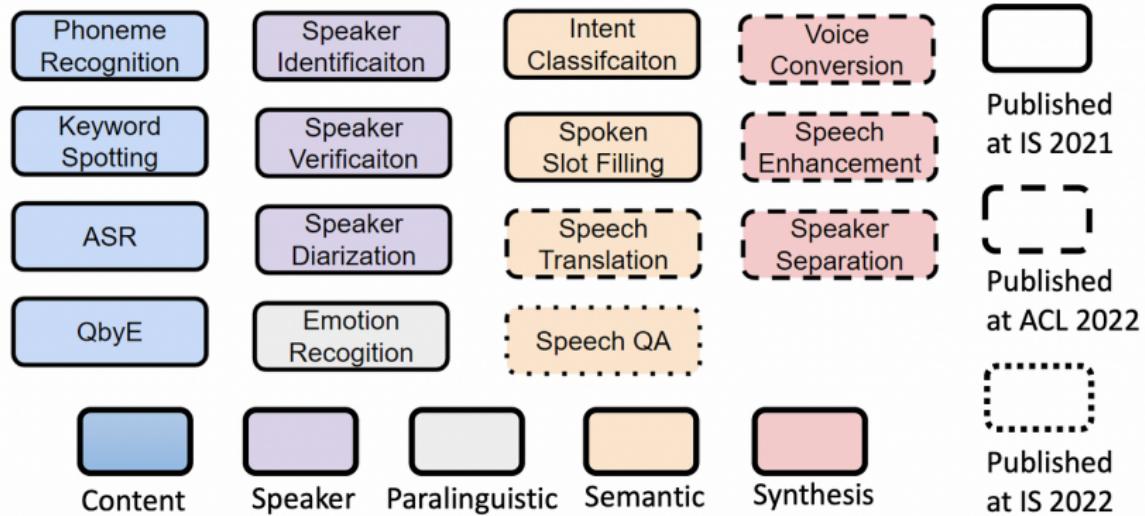
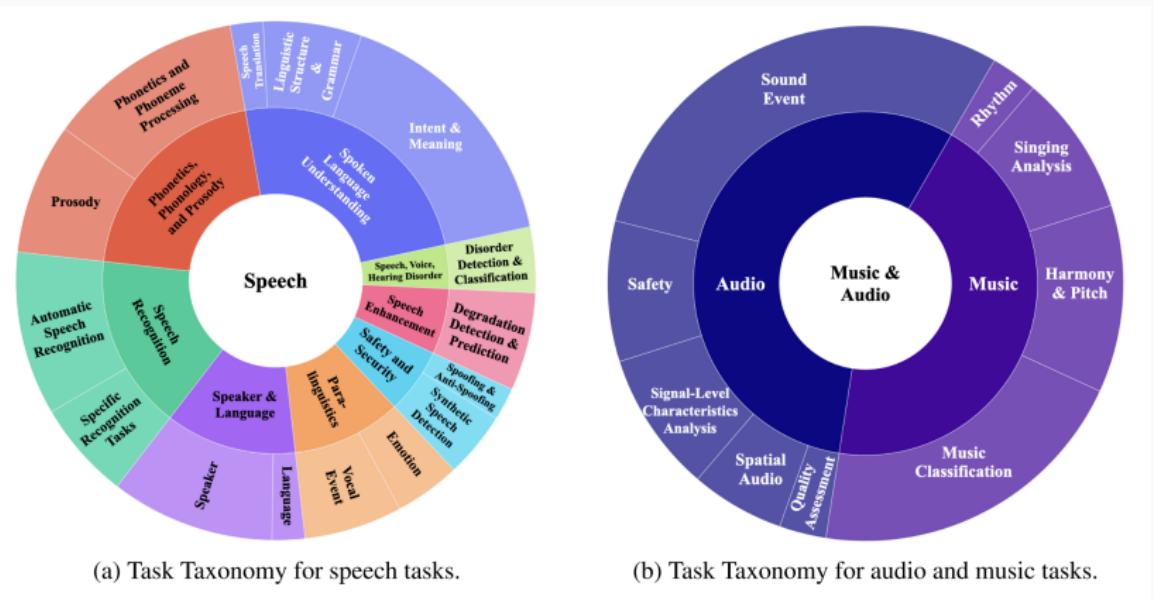


Figure 8: Los modelos se evalúan a través de una variedad de tareas posteriores. Como los modelos de NLP en GLUE [22]

# Dynamic-SUPERB [15, 14]



**Figure 9:** Dynamic-SUPERB es una colección en evolución de 180 tareas de "comprensión" de voz + audio (audio de entrada, texto de salida).

- Instrucción + entrada de audio → salida de texto
- Evaluado con LLM como juez

# Outline : Conjuntos de datos de pre-entrenamiento

---

Datos de Audio

Representaciones

Codificadores de Voz

LLMs de Voz

Benchmarks

Conjuntos de datos de  
pre-entrenamiento

Aplicaciones

# Conjuntos de Datos de Voz

---

- **LibriSpeech:** 1,000 horas de voz de audiolibros.
- **VoxCeleb:** Conjunto de datos de reconocimiento de hablante con voces diversas: 7k+ horas en condiciones reales.
- **AudioSet:** Más de 2 millones de clips de audio etiquetados de 600+ clases

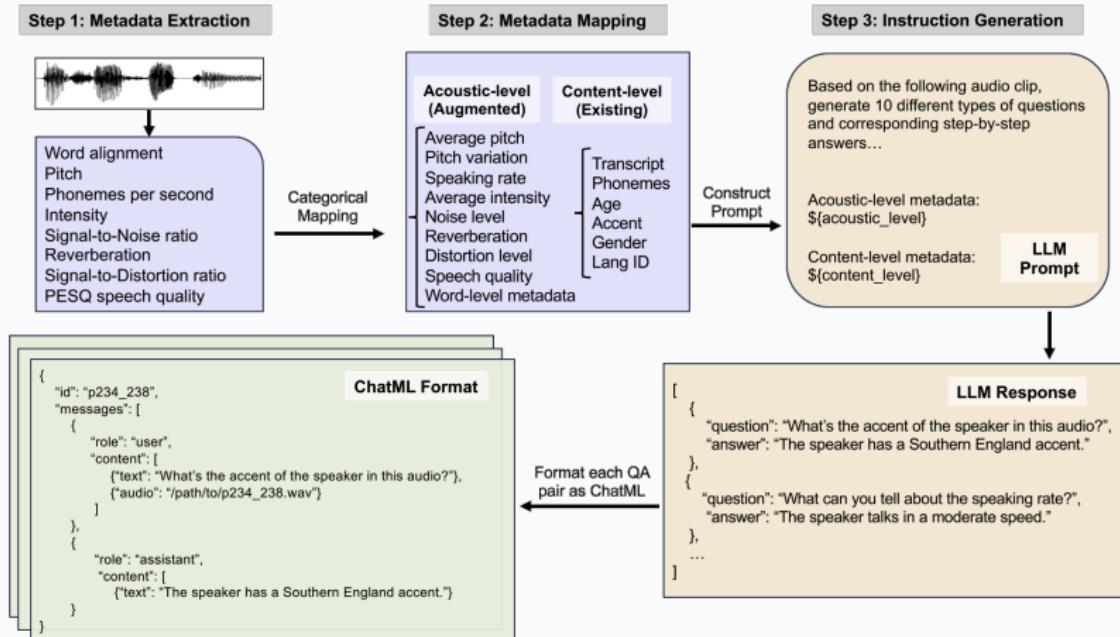
# Conjuntos de Datos de Pre-entrenamiento de Voz y Audio

Conjunto	Tamaño	Tipo	Características Clave
LibriSpeech	1,000h	Voz limpia	Audiolibros en inglés, alta calidad, ampliamente usado para ASR
LibriLight	60,000h	Voz sin etiquetar	LibriSpeech extendido para aprendizaje auto-supervisado
VoxPopuli	400,000h	Voz multilingüe	23 idiomas de grabaciones del Parlamento Europeo
VoxCeleb 1/2	2,000h	ID de hablante	7,000+ hablantes, condiciones reales, acentos diversos
Common Voice	20,000h+	Crowdsourced	100+ idiomas, hablantes diversos, impulsado por comunidad
AudioSet	2M clips	Audio general	632 clases, sonidos ambientales, música, voz
FSD50K	51,000 clips	Eventos sonoros	Conjunto de Freesound, sonidos cotidianos diversos
MusicCaps	5,500 clips	Música	Música con subtítulos de texto para generación musical

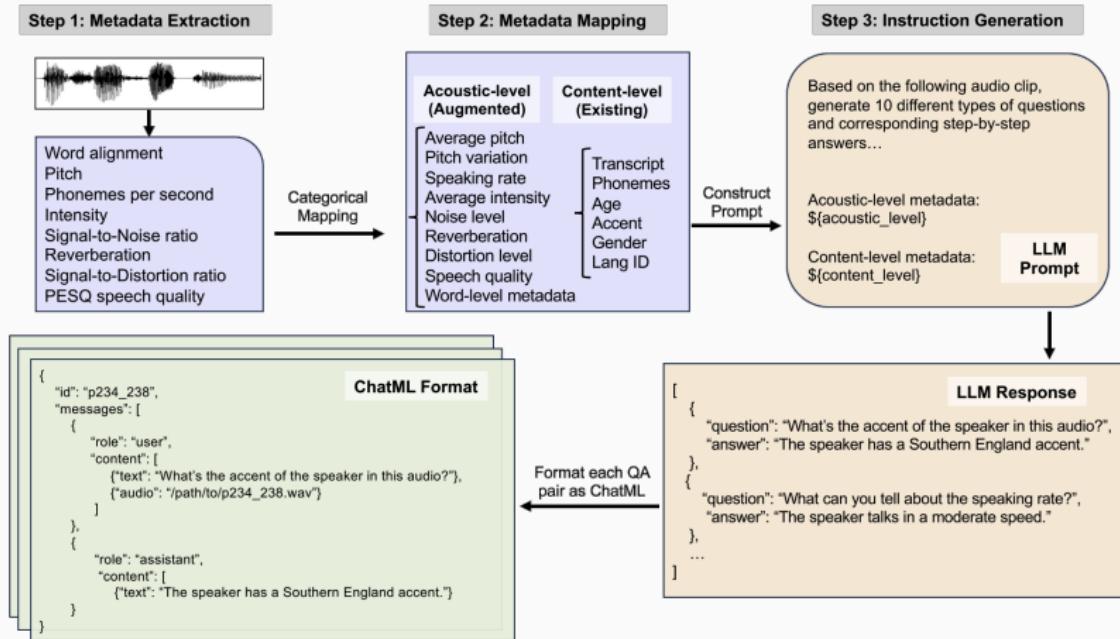
## Especialización de Conjuntos de Datos

**Solo voz:** LibriSpeech, LibriLight, VoxPopuli, Common Voice — **Reconocimiento de hablante:** VoxCeleb — **Audio general:** AudioSet, FSD50K — **Música:** MusicCaps

# Ajuste Fino de Instrucciones: SIFT-50m [19]



# Ajuste Fino de Instrucciones: SIFT-50m [19]



Model	Closed-Ended		Open-Ended		Dynamic-Superb Tasks					
	DS-1	EvalSIFT	AB-Chat	EvalSIFT	Audio	PL	Semt.	Degr. Degr.	Content	Speaker
SALMONN-7B	34.7	21.9	6.4	6.0	31.7	<u>30.5</u>	<u>47.5</u>	30.0	45.2	31.9
Qwen2-Audio-Inst.	<u>48.0</u>	<u>25.1</u>	<u>7.2</u>	<u>7.3</u>	<b>53.5</b>	28.9	40.3	43.9	70.6	<u>43.6</u>
O-ASQA-LLM	45.9	22.9	6.6	4.7	28.5	30.0	38.6	<u>45.9</u>	<u>72.3</u>	40.7
SIFT-LLM (ours)	<b>57.4</b>	<b>46.1</b>	<b>7.3</b>	<b>7.8</b>	<u>37.5</u>	<b>42.8</b>	<b>51.3</b>	<b>63.6</b>	<b>75.6</b>	<b>47.7</b>

# Outline : Aplicaciones

---

Datos de Audio

Representaciones

Codificadores de Voz

LLMs de Voz

Benchmarks

Conjuntos de datos de  
pre-entrenamiento

## Aplicaciones

Síntesis de voz

Generación de música y audio

Reconocimiento automático de voz

# WaveNet [18]

## Ventajas de las convoluciones causales dilatadas apiladas

- **Convoluciones dilatadas** → campo receptivo exponencialmente creciente
- **Paralelizable** sobre tiempo → entrenamiento rápido
- **Causal** → sin fuga de futuro

# Jukebox: Generación Neural de Música

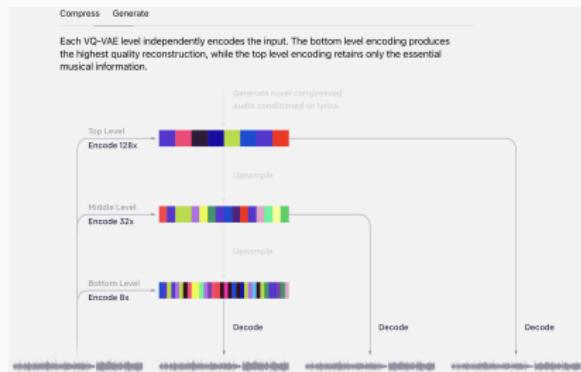
## Descripción general, y ejemplos

Genera música como **audio crudo** con estilos de artista, géneros y canto. Primera generación neural de música a gran escala.

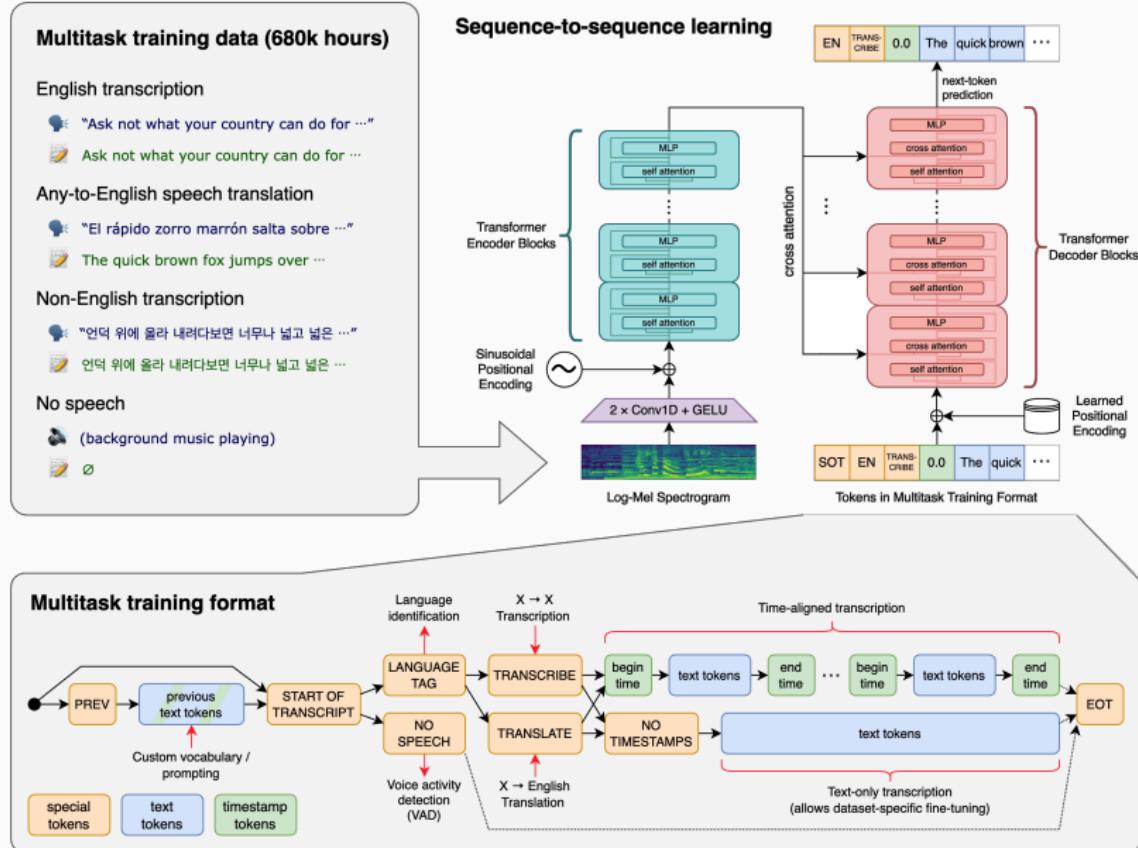
**Desafío Clave:** Canción de 4 min a 44kHz = **10M+ pasos temporales**  
⇒ Debe aprender dependencias de largo alcance

**Conjunto de datos:** 1.2M canciones, letras, metadatos (Artista, género, año, etiquetas de ánimo)

**Condicionamiento:** Artista, género y letras vía atención codificador-decodificador



# Whisper [20]



**Questions?**

## References i

-  S. Arora, K.-w. Chang, C.-m. Chien, Y. Peng, H. Wu, Y. Adi, E. Dupoux, H.-Y. Lee, K. Livescu, and S. Watanabe.  
**On The Landscape of Spoken Language Models: A Comprehensive Survey.**  
TMLR, pages 1–40, 2025.
-  A. Baevski, S. Schneider, and M. Auli.  
**Vq-Wav2Vec: Self-Supervised Learning of Discrete Speech Representations.**  
8th International Conference on Learning Representations, ICLR 2020, pages 1–12, 2020.
-  A. Baevski, H. Zhou, A. Mohamed, and M. Auli.  
**wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations.**  
arXiv, (Figure 1):1–19, 2020.

## References ii

-  J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, F. Huang, B. Hui, L. Ji, M. Li, J. Lin, R. Lin, D. Liu, G. Liu, C. Lu, K. Lu, J. Ma, R. Men, X. Ren, X. Ren, C. Tan, S. Tan, J. Tu, P. Wang, S. Wang, W. Wang, S. Wu, and B. Xu.  
**Qwen Technical Report.**  
pages 1–59, 2023.
-  S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao, J. Wu, L. Zhou, S. Ren, Y. Qian, Y. Qian, J. Wu, M. Zeng, X. Yu, and F. Wei.  
**WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing.**  
IEEE Journal on Selected Topics in Signal Processing,  
16(6):1505–1518, 2022.

-  Y. Chu, J. Xu, Q. Yang, H. Wei, X. Wei, Z. Guo, Y. Leng, Y. Lv, J. He, J. Lin, C. Zhou, and J. Zhou.  
**Qwen2-Audio Technical Report.**  
pages 1–16, 2024.
-  Y. Chu, J. Xu, X. Zhou, Q. Yang, S. Zhang, Z. Yan, C. Zhou, and J. Zhou.  
**Qwen-Audio: Advancing Universal Audio Understanding via Unified Large-Scale Audio-Language Models.**  
pages 1–18, 2023.
-  A. Défossez, J. Copet, G. Synnaeve, and Y. Adi.  
**High Fidelity Neural Audio Compression.**  
Transactions on Machine Learning Research, 2023:1–19, 2023.

-  A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby.  
**An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.**  
In ICLR, pages 1–21, 2021.
-  B. Elizalde, S. Deshmukh, and H. Wang.  
**NATURAL LANGUAGE SUPERVISION FOR GENERAL-PURPOSE AUDIO REPRESENTATIONS.**  
In ICASSP, 2024.

## References v

-  A. Goel, S. Ghosh, J. Kim, S. Kumar, Z. Kong, S.-g. Lee, C.-H. H. Yang, R. Duraiswami, D. Manocha, R. Valle, and B. Catanzaro.  
**Audio Flamingo 3: Advancing Audio Intelligence with Fully Open Large Audio Language Models.**  
pages 1–61, 2025.
-  Y. Gong, Y. A. Chung, and J. Glass.  
**Ast: Audio spectrogram transformer.**  
In Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, volume 1, pages 56–60, 2021.

-  J. B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko.  
**Bootstrap your own latent a new approach to self-supervised learning.**  
In Advances in Neural Information Processing Systems, volume 2020-Decem, 2020.
-  W. N. Hsu, B. Bolte, Y. H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed.  
**HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units.**  
IEEE/ACM Transactions on Audio Speech and Language Processing, 29(Cv):3451–3460, 2021.

- 
- C.-y. Huang, W.-C. Chen, S.-w. Yang, A. T. Liu, C.-a. Li, Y.-X. Lin, W.-c. Tseng, A. Diwan, Y.-j. Shih, J. Shi, W. Chen, C.-k. Yang, W. Ren, X. Chen, C.-Y. Hsiao, P. Peng, S.-H. Wang, C.-Y. Kuan, K.-H. Lu, K.-W. Chang, F. Ritter-Gutierrez, K.-P. Huang, S. Arora, Y.-K. Lin, M. T. Chuang, E. Yeo, K. Chang, C.-m. Chien, K. Choi, J.-y. Wang, C.-h. Hsieh, Y.-C. Lin, C.-E. Yu, I.-H. Chiu, H. R. Guimarães, J. Han, T.-Q. Lin, T.-Y. Lin, H. Chang, T.-W. Chang, C. W. Chen, S.-J. Chen, Y.-h. Chen, H.-c. Cheng, K. Dhawan, J.-L. Fang, S.-x. Fang, K.-y. F. Chiang, C. A. Fu, H.-f. Hsiao, C. Y. Hsu, S.-S. Huang, L. C. Wei, H.-C. Lin, H.-H. Lin, H.-T. Lin, J.-r. Lin, T.-c. Liu, L.-c. Lu, T.-m. Pai, A. Pasad, S.-Y. S. Kuan, S. Shon, Y. Tang, Y.-S. Tsai, J.-C. Wei, T.-C. Wei, C. Wu, D.-R. Wu, C.-H. H. Yang, C.-C. Yang, J. Q. Yip, S.-X. Yuan, V. Noroozi, Z. Chen, H. Wu, K. Livescu, D. Harwath, S. Watanabe, and H.-y. Lee.

**Dynamic-SUPERB Phase-2: A Collaboratively Expanding Benchmark for Measuring the Capabilities of Spoken Language Models with 180 Tasks.**

In ICLR, pages 1–67, 2025.



C. Y. Huang, K. H. Lu, S. H. Wang, C. Y. Hsiao, C. Y. Kuan, H. Wu, S. Arora, K. W. Chang, J. Shi, Y. Peng, R. Sharma, S. Watanabe, B. Ramakrishnan, S. Shehata, and H. Y. Lee.

**DYNAMIC-SUPERB: TOWARDS A DYNAMIC, COLLABORATIVE, AND COMPREHENSIVE INSTRUCTION-TUNING BENCHMARK FOR SPEECH.**

In ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, pages 12136–12140, 2024.

-  P. Y. Huang, H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, and C. Feichtenhofer.  
**Masked Autoencoders that Listen.**  
Advances in Neural Information Processing Systems,  
35(NeurIPS):1–13, 2022.
-  Z. Kong, A. Goel, R. Badlani, W. Ping, R. Valle, and B. Catanzaro.  
**Audio Flamingo: A Novel Audio Language Model with Few-Shot Learning and Dialogue Abilities.**  
Proceedings of Machine Learning Research, 235:25125–25148, 2024.
-  A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu.  
**WaveNet: A Generative Model for Raw Audio.**  
pages 1–15, 2016.

## References x

-  P. Pandey, R. V. Swaminathan, K. V. V. Girish, A. Sen, J. Xie, G. Strimel, and A. Schwarz.  
**SIFT-50M: A Large-Scale Multilingual Dataset for Speech Instruction Fine-Tuning.**  
In ACL, volume 1, pages 13921–13942, 2025.
-  A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever.  
**Robust Speech Recognition via Large-Scale Weak Supervision.**  
Proceedings of Machine Learning Research, 202:28492–28518, 2023.
-  A. Van Den Oord, O. Vinyals, and K. Kavukcuoglu.  
**Neural discrete representation learning.**  
Advances in Neural Information Processing Systems,  
2017-Decem(Nips):6307–6316, 2017.

-  A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman.  
**GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding.**  
In EMNLP 2018 - 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Proceedings of the 1st Workshop, pages 353–355, 2018.
-  Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov.  
**Large-Scale Contrastive Language-Audio Pretraining with Feature Fusion and Keyword-to-Caption Augmentation.**  
ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2023-June, 2023.

-  S. W. Yang, P. H. Chi, Y. S. Chuang, C.-i. I. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G. T. Lin, T. H. Huang, W. C. Tseng, K. T. Lee, D. R. Liu, Z. Huang, S. Dong, S. W. Li, S. Watanabe, A. Mohamed, and H. Y. Lee.

**SUPERB: Speech processing Universal PERformance Benchmark.**

In Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, volume 4, pages 3161–3165, 2021.