Procesamiento del Lenguaje Natural, Revista nº 76, marzo de 2026, pp. 239-250 recibido 05-12-2025 revisado 20-01-2026 aceptado 16-02-2026

Preservando la Identidad en el Habla: Transcripción Anonimizada para el Contexto Colombiano

Preserving Identity in Speech: Anonymized Transcription for the Colombian Context

Andrea Juliana Parra Ariza, Hoover Rueda-Chacón
Department of Computer Science, Universidad Industrial de Santander,
Bucaramanga, Colombia, 680002
andrea2269119@correo.uis.edu.co, hfarueda@uis.edu.co

Resumen: El habla ha motivado el desarrollo de modelos de Reconocimiento Automático del Habla (ASR, del inglés *Automatic Speech Recognition*) como *Whisper*, capaces de convertir el habla en texto escrito. Sin embargo, estos modelos requieren grandes volúmenes de datos (corpus), lo que limita su desempeño en idiomas o variantes con recursos limitados, como el español de Colombia, cuyos acentos y regionalismos están poco representados. Así mismo, el uso de grabaciones suele incluir información sensible, como nombres o identificaciones, que dificulta la recopilación e intercambio de estos corpus. Este trabajo propone desarrollar un modelo basado en la arquitectura de *Whisper* y el flujo de trabajo de *WhisperX* para la transcripción de voz anonimizada en el español colombiano, con anotación temporal y diarización de hablantes. Con modelos que alcanzan un 7,60 % de error de transcripción a nivel de palabra (WER), un F1-score de 60,81 % para reconocimiento de entidades y un F1-score de 76,10 % en anonimización, se aporta al cierre de la brecha entre los modelos existentes y los dialectos colombianos, garantizando un desempeño robusto incluso en entornos con datos escasos.
Palabras clave: Reconocimiento de Entidades Nombradas, Anonimización, Reconocimiento Automático del Habla, Transcripción.

Abstract: Speech has driven the development of Automatic Speech Recognition (ASR) models like *Whisper*, capable of converting spoken language into written text. However, these models require large amounts of data (corpora), which limits their performance in languages or variants with scarce resources, such as the Colombian Spanish, whose accents and regionalisms are underrepresented. Likewise, the use of recordings often includes sensitive information, such as names or IDs, which makes the collection and sharing of these corpora difficult. This work proposes the development of a model based on the *Whisper* arquitecture and *WhisperX*'s pipeline, for anonymized speech transcription in Colombian Spanish, with temporal annotation and speaker diarization. With models achieving a 7,60 % transcription word error rate (WER), an F1-score of 60,81 % for named entity recognition, and an F1-score of 76,10 % for anonymization, it contributes to closing the gap between existing models and Colombian dialects, ensuring robust performance even in low-resource settings.
Keywords: Named Entity Recognition, Anonymization, Automatic Speech Recognition, Transcription.

1 Introducción

El habla, una de nuestras habilidades más esenciales, ha motivado el desarrollo de modelos computacionales capaces de emular la comunicación verbal humana. Desde los primeros sistemas de Reconocimiento Automático del Habla (ASR) en la década de 1950 con Audrey (Moskvitch, 2017), hasta modelos actuales basados en Redes Neuronales Profundas (DNN), el campo ha experimentado avances significativos. Destaca especialmente *Whisper* (Radford et al., 2023), un modelo de transcripción multilingüe desarrollado por OpenAI, capaz de generalizar a múltiples *benchmarks* estándar y publicado

ISSN 1135-5948 DOI 10.26342/2026-76-16

como código abierto para servir como base en futuras investigaciones sobre el procesamiento robusto del habla. Defínase transcripción en el contexto de ASR, como el proceso de convertir una oración hablada en una secuencia escrita de letras y palabras (Ahlawat, Aggarwal, y Gupta, 2025). Más allá de la transcripción, se han comenzado a integrar tecnologías basadas en DNN en tareas de Procesamiento del Lenguaje Natural (NLP, del inglés *Natural Language Processing*), tal como el reconocimiento de entidades (NER, del inglés *Named Entity Recognition*), análisis de sentimientos, reconocimiento del lenguaje conversacional, entre otros (Qin et al., 2024).

Un desafío común en el reconocimiento de voz es la variabilidad de sílabas y fonemas de una misma palabra según la entonación y pronunciación del hablante, dificultando su identificación por parte de los modelos, problema que se intensifica en contextos como la variación de acentos en diferentes lenguas (Basak et al., 2022). Para llegar a un buen rendimiento en tales escenarios, estos modelos necesitan grandes cantidades de datos, pues un conjunto de datos incompleto, sesgado, o en formatos inconsistentes, afecta el desempeño del modelo al predecir o generar texto (Zeroual y Lakhouaja, 2018). Es por esto que la mayoría de estos avances se han limitado a los idiomas de mayor población hablante. En consecuencia, los idiomas con poca cantidad de habla transcrita o no transcrita, es decir, los idiomas con recursos limitados, se han quedado atrás (Yadav y Sitaram, 2022). En el caso de Colombia, aunque es un país hispanohablante, posee acentos y regionalismos que los modelos de lenguaje actuales no interpretan con precisión, debido a la escasez de datos transcritos de su variante lingüística. Desde el punto de vista de corpus lingüísticos colombianos se identifican 8 en el Instituto Caro y Cuervo (CLICC) (López, Cortés, y Guzmán, 2015), entre los que se encuentran: "Acervo de tradición oral afrocaucano 'Manuel Y Constanza USSA'", "Atlas Lingüístico-Etnográfico de Colombia", o "Literatura de la violencia bipartidista en Colombia". Estos conjuntos de datos, sin embargo, tienen 2 limitaciones importantes: (i) debido a las políticas de derechos de autor, no se permite acceder ni descargar obras completas, (ii) no están en el formato necesario para el entrenamiento de modelos de lenguaje modernos. También existe un corpus creado por la Comisión de la Verdad (CEV) (Comisión de la Verdad, 2024) que contiene la transcripción y anonimización de 2240 entrevistas relacionadas con el conflicto armado en Colombia; sin embargo, este corpus tampoco cuenta con etiquetas para tareas de comprensión de lenguaje natural. Asimismo, políticas de privacidad dificultan la creación de estos datos, ya que la difusión de esta información puede comprometer identidades sensibles mencionadas, lo que hace necesaria la anonimización de datos personales.

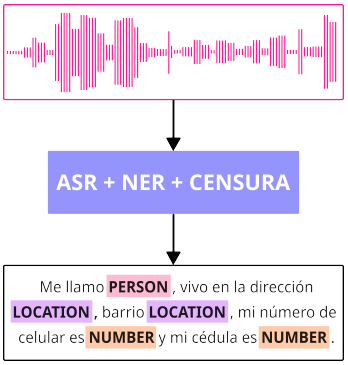

Figura 1: Representación del resultado de la transcripción anonimizada de audio.

En este trabajo se desarrolla un modelo de lenguaje natural para la transcripción anonimizada de grabaciones de audio del español de Colombia (véase la Figura 1), con el objetivo de ayudar a cerrar esta brecha entre los modelos de español actuales y el español de Colombia. Nos basamos en la arquitectura de *Whisper* y ajustamos con *LoRA* (Hu et al., 2022) el modelo *WhisperNER* (Ayache et al., 2024) a nuestro conjunto de datos, en el cual se alcanza un 7,60 % de error de transcripción a nivel de palabra (*Word Error Rate*), un F1-score de 60,81 % para NER exacto y un F1-score de 76,10 % en censura de entidades sensibles; resultados competitivos que demuestran el potencial de la estrategia utilizada y la necesidad de contar con conjuntos de datos etiquetados en múltiples variantes lingüísticas.

2 Trabajos relacionados

2.1 Reconocimiento de Entidades Nombradas (NER)

El Reconocimiento de Entidades Nombradas es un subcampo de la informática y el NLP que se centra en identificar y clasificar entidades presentes en textos no estructurados, en

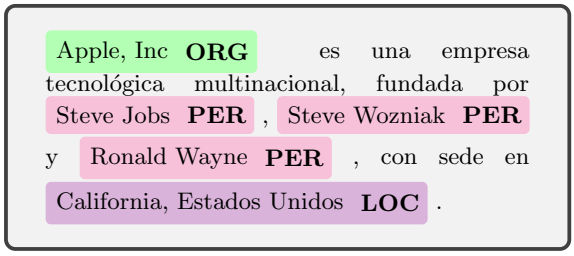

Figura 2: Ejemplo de reconocimiento de entidades nombradas.

categorías predefinidas como personas, ubicaciones geográficas y organizaciones (Tjong Kim Sang, 2002). Con el tiempo, estas categorías han evolucionado a incluir conceptos más complejos en dominios especializados como la biomedicina o el ámbito jurídigo-legal (Keraghel, Morbieu, y Nadif, 2024; Au, Lampos, y Cox, 2022). Por ejemplo, en la Figura 2 se observa como ciertas palabras y frases han sido resaltadas y etiquetadas con abreviaciones para indicar la entidad que representan: Apple, Inc es una organización (etiqueta ORG), California y Estados Unidos son localizaciones (etiqueta LOC), y Steve Jobs, Steve Wozniak y Ronald Wayne son personas (etiqueta PER).

NER tiene múltiples aplicaciones en varios dominios: En tareas de recuperación de información, es crucial para identificar entidades relevantes tanto en las consultas de búsqueda como en los resultados, mejorando así la precisión (Banerjee et al., 2019). En la generación de resúmenes automáticos ayuda a resaltar la información más importante, mientras que en el monitoreo de redes sociales se emplea para detectar y clasificar menciones de marcas o temas de interés (Roha et al., 2023; Keraghel, Morbieu, y Nadif, 2024). En la anonimización de documentos, la extracción de entidades o información sensible garantiza la privacidad de las personas y habilita el uso de datos delicados para la investigación sin comprometer la confidencialidad (Mancera y others, 2026).

2.2 Reconocimiento Automático del Habla (ASR)

El Reconocimiento Automático del Habla (ASR, por sus siglas en inglés *Automatic Speech Recognition*) es un subcampo interdisciplinario del NLP que permite el reconocimiento y la traducción del lenguaje hablado a texto (Rista y Kadriu, 2020). Actualmente, los modelos de ASR se basan en una arquitectura *Transformer*, la cual ha demostrado ser altamente efectiva y robusta, como se ve en Whisper (Radford et al., 2023). El audio de entrada se convierte primero en un *espectrograma Mel* normalizado, luego, el *encoder* o codificador procesa esta representación mediante dos capas convolucionales, añade *embeddings* posicionales sinusoidales y pasa la señal por varios bloques *Transformer* con mecanismos de atención, capas *feed-forward* y conexiones residuales de preactivación. Finalmente, el *decoder* o decodificador, que emplea *embeddings* posicionales aprendibles y atención cruzada al codificador, genera la transcripción. Este decodificador puede condicionarse mediante *prompts* o tokens especiales para adaptar la decodificación a tareas específicas o modos multitarea.

Una forma de condicionar estas transcripciones mediante *prompts* se ve en Whisper-NER, donde restringen el proceso de decodificación a una lista de etiquetas de entidad $\mathbf{p} = [p_1, p_2, \ldots, p_I]$ para el que cada p_i representa un tipo de entidad específica, como persona, localización, etc. Su salida es una secuencia de tokens $\mathbf{y} = [y_1, y_2, ..., y_N]$, que comprende tanto el texto transcrito como las etiquetas de entidad correspondientes. Esta alternativa es una solución a las arquitecturas modulares que suelen tomarse para tareas de post-procesamiento NLP que presentan acumulación de errores (Ayache et al., 2024).

Por otro lado, las marcas de tiempo predichas a nivel de enunciado en Whisper suelen presentar imprecisiones, y no se ofrecen de manera nativa a nivel de palabra. Debido a este problema surgen modelos como WhisperX, un sistema ASR con precisión temporal y marcas de tiempo a nivel de palabra, demostrando un rendimiento de vanguardia en la transcripción y anotación temporal de audios (Bain et al., 2023).

3 Propuesta

3.1 Creación del conjunto de datos

Frente a la ausencia de conjuntos de datos públicos anotados para Colombia, se procedió a crear uno por medio de un *pipeline* automatizado que combina extracción automática de datos web (*web scraping*) y descarga de audios. La recolección de datos se realizó a partir de videos publicados en la red social *TikTok*, aprovechando su sistema de *hashtags*; esta elección se fundamenta en que los usuarios tienden a etiquetar la ciudad en sus pu-

Ciudad	Región	Dialecto	# Audios	Duración [min]
Arauca	Arauca	Llanero	184	62.93
Barranquilla	Atlántico	Costeño – Atlántico	521	186.29
Bogotá	Cundinamarca	Andino – Oriental	415	149.36
Bucaramanga	Santander	Andino – Oriental	466	163.10
Cali	Valle del Cauca	Andino – Occidental	379	133.15
Cúcuta	Norte de Santander	Andino – Oriental	422	150.64
Medellín	Antioquia	Andino – Occidental	467	152.51
Neiva	Huila	Andino – Oriental	279	96.58
Pasto	Nariño	Andino – Occidental	330	114.09
Quibdó	Chocó	Costeño – Pacífico	219	73.96
San Andrés	Arch. San Andrés	San Andrés	219	78.98
Tunja	Boyacá	Andino – Oriental	342	118.05
Yopal	Casanare	Llanero	279	95.93
Total	-	-	**4522**	**1575.57**

Tabla 1: Audios recopilados, comparados por ciudad, región, dialecto, cantidad y duración.

blicaciones, lo que facilita la identificación de videos asociados a cada región y favorece la representatividad del corpus. En primer lugar, se define una función para obtener el audio de un video a partir de su URL utilizando *yt-dlp*(yt-dlp Project, 2025), seguido de un postprocesamiento con *FFmpeg*(FFmpeg Developers, 2025) para extraer el audio y convertirlo a formato `wav` con calidad de 192 kbps. Para recolectar los enlaces de los videos se usa *Selenium* (SeleniumHQ, 2025) mediante su *WebDriver*, el cual permite automatizar la interacción con el navegador, accediendo a la página de TikTok mediante una palabra clave o *hashtag* específico (por ejemplo, #Bucaramanga). Una vez cargado el contenido principal, se realiza un scroll iterativo hasta que no se detecta contenido nuevo y se extraen todos los enlaces que contienen videos, filtrando aquellos que no corresponden a material audiovisual. Finalmente, cada enlace de video se procesa con la función de descarga de audio, registrando el progreso y los posibles errores que se presenten.

Para asegurar diversidad de acentos y expresiones, se escogieron 13 regiones colombianas que representan distintos dialectos y acentos descritos en el Atlas Lingüístico-Etnográfico de Colombia (ALEC) y otros estudios (Bernal Chávez, 2017). Aunque no cubren toda la riqueza dialectal del país, incluyen subdialectos principales (como el acento *paisa*) y garantizan una muestra diversa de los 5 dialectos reconocidos: costeño atlántico, costeño pacífico, andino occidental, andino oriental y llanero. Puesto a que San Andrés no presenta clasificación dialectal explícita en el ALEC, se considera una zona dialectal propia. En la Tabla 1 se muestra la ciudad, región, dialecto, número de audios recopilados y duración total del conjunto de datos

Tras recopilar los audios de cada ciudad, se generaron transcripciones automáticas iniciales utilizando *Whisper large-v2* (Radford et al., 2023). Posteriormente, se verificó cada audio y su transcripción manualmente, descartando aquellos inutilizables, por ejemplo en otros idiomas, con acentos de regiones distintas a la esperada o con sólo música. Debido a este descarte, la base de datos final presenta una distribución no uniforme de audios entre ciudades y, en consecuencia, una representación desigual en dialectos. Adicionalmente, la propia plataforma limitó el número de videos accesibles por cada etiqueta. En algunos casos el sistema permitía descargar alrededor de 400 videos, en otros hasta 700, pero siempre existía un punto en el que no retornaba nuevos resultados. Esto generó diferencias notorias entre regiones: ciudades con mayor volumen de publicaciones, como Bogotá, aportaron muchos más audios que regiones con menor actividad o visibilidad, como Pasto.

Finalmente, para automatizar la obtención de las etiquetas de entidades de cada transcripción se utilizó el modelo *NER-spanish-large* de la librería Flair (Schweter y Akbik, 2020a), basado en la arquitectura FLERT para reconocimiento de entidades nombradas, por sus resultados competitivos en el conjunto de datos CoNLL-03 (Spanish) donde reporta un F1-score del 90,54 %. El modelo permite identificar únicamente cuatro tipos de entidades: *"PER"* (persona), *"LOC"* (localización), *"ORG"* (organización) y *"MISC"* (misceláneas). Las etiquetas generadas automáticamente se emplearon como *ground truth* para el entrenamiento y la

evaluación del modelo propuesto. Debido a la complejidad y al tiempo requerido, no fue posible realizar una verificación manual exhaustiva de todas las etiquetas NER generadas. Una vez completado el proceso de etiquetado, el conjunto de datos fue dividido en entrenamiento, validación y prueba.

3.2 Flujo de trabajo para la transcripción anonimizada

Para nuestro método, las entradas consisten un archivo de audio y una lista de *prompts* textuales $\mathbf{p} = [p_1, p_2, \ldots, p_I]$, para el cual cada p_i representa un tipo de entidad en específico. El modelo WhisperNER se adapta a nuevas entidades pero para este proyecto por defecto se trabaja con 4 tipos de entidades: *"person"*, *"location"*, *"organization"* y *"miscellaneous"*.

Antes de procesar el audio con el modelo de ASR se usa un modelo de detección de voz (VAD, por sus siglas en inglés *Voice Activity Detection*) –bien sea *Silero* (Team, 2024), un modelo ligero en PyTorch optimizado para detección rápida, o *pyannote* (Bredin et al., 2020), modelo más complejo con mayor robustez en escenarios de diarización– para pre-segmentarlo en regiones que contienen habla activa. Para esta tarea, la forma de onda del audio de entrada se representa como una secuencia de vectores de características acústicas extraídos por paso de tiempo $\mathcal{A} = \{a_1, a_2, ..., a_T\}$ y la salida es una secuencia de etiquetas binarias $z = [z_1, z_2, ..., z_T]$, para la cual $z_t = 1$ significa que hay habla en el paso de tiempo t, y $z_t = 0$ viceversa.

Estas predicciones son luego representadas como una secuencia de segmentos de habla activos $s = \{s_1, s_2, ..., s_M\}$, con índices de inicio y fin $s_m = (l_0^m, l_1^m)$. Luego, se hace una operación *Cut & Merge* (cortar y fusionar en español): cada segmento s_m se corta en el punto con puntuación mínima de activación de voz; este corte se restringe entre $\frac{1}{2}|\mathcal{A}_{train}|$ y $|\mathcal{A}_{train}|$, siendo $|\mathcal{A}_{train}|$ la duración máxima que acepta el modelo ASR durante entrenamiento, en nuestro caso 30 segundos. En caso que hayan fragmentos s_m cortos, se hace la operación inversa; teniendo 2 segmentos adyacentes s_m y s_{m+1}, si la duración total del intervalo combinado $d_{i,i+1} = l_1^{m+1} - l_0^m$ es menor a un umbral de duración máxima τ, para el que $\tau \leq |\mathcal{A}_{train}|$, entonces se fusionan.

Los segmentos de voz resultantes, con una duración temporal (en segundos) aproximadamente igual a la longitud promedio de los ejemplos de entrenamiento del modelo ASR, $|s_m| \approx |\mathcal{A}_{\text{train}}|$, son preprocesados para convertirlos en una representación adecuada para el modelo. Este preprocesamiento sigue el pipeline estándar de Whisper, transformando la señal en un espectrograma Log-Mel de 80 canales, calculado con ventanas de 25 ms y un paso (*stride*) de 10 ms ($\mathbf{X} \in \mathbb{R}^{F \times 80}$, siendo F el número de tramas (*frames*) de tiempo generados). Nuestro $\mathbf{X}$ resultante representa las características de cada audio de entrada que entran al codificador, el cual produce una secuencia de estados ocultos(*hidden states*), $\mathbf{h} = Encoder(\mathbf{X})$, que se usan para condicionar el decodificador junto a nuestro conjunto de etiquetas de entidad $\mathbf{p}$. El decoder genera una secuencia de tokens $\mathbf{y} = [y_1, y_2, ..., y_N]$, que comprende tanto el texto transcrito como las etiquetas de entidad correspondientes, por ejemplo: *Hola, soy <person> Miguel <person>*. Para ASR se implementaron tres adaptadores LoRA (Hu et al., 2022) –Focal NER, Cross NER y Censura (ver Figura 4)–, cada uno competente en una tarea específica; entiéndase LoRA como un método de *fine-tuning* para adaptar un modelo preentrenado mediante la inyección de matrices de descomposición de bajo rango entrenables y los adaptadores como los módulos que incorporan dichas matrices dentro del modelo. Focal NER destaca en la transcripción, Cross NER en el reconocimiento de entidades, y Censor o Censura incorpora la anonimización directamente en el proceso de inferencia.

Finalmente, para cada segmento de audio s_m empleamos un modelo de reconocimiento de fonemas (bloque *Phoneme model* en la Figura 3) junto con *Dynamic Time Warping* (bloque *Forced alignment* en la Figura 3) con el fin de obtener marcas de inicio y fin a nivel de palabra con alta precisión. Además, utilizamos la *pipeline* de diarización de pyannote.audio (Bredin et al., 2020) para identificar los hablantes en cada segmento. Estos módulos siguen la arquitectura propuesta por WhisperX (Bain et al., 2023).

3.3 Estrategia de entrenamiento

Para mitigar el desbalance entre ciudades y dialectos, se optó por una partición de entrenamiento, validación y test que (1) distribuyera de la manera más equilibrada posible las ciudades en cada subconjunto y (2) garantizara que, en los casos donde un audio

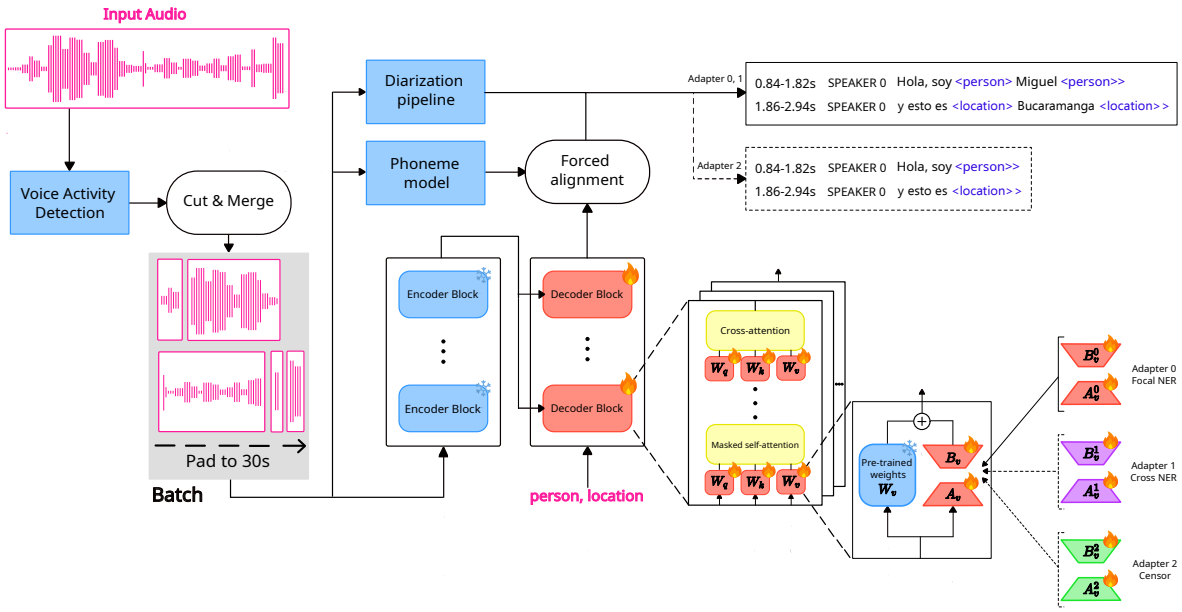

Figura 3: Flujo de trabajo del método propuesto para la transcripción anonimizada de grabaciones de audio del español de Colombia, con anotación temporal y diarización de hablantes. El modelo integra el sistema propuesto por WhisperX para obtener marcas temporales precisas, complementado con el método de diarización de *pyannote*. Para la transcripción, se emplea un *fine-tuning* de WhisperNER utilizando LoRA sobre los pesos de atención en el *decoder*. Durante la inferencia, es posible seleccionar distintos adaptadores LoRA en función del modelo deseado (FL + NER, CE + NER o censura).

Entidad	Total	Train	Val	Test
PER	1447	903	217	327
ORG	830	481	123	226
LOC	4610	2693	932	985
MISC	2126	1276	425	422
Total	**9013**	**5353**	**1697**	**1961**

Tabla 2: Distribución de entidades por partición.

Dialecto	Train	Val	Test
Llanero	289	96	78
Costeño – Atlántico	310	111	100
Costeño – Pacífico	111	51	57
Andino – Oriental	1190	373	361
Andino – Occidental	675	241	260
San Andrés	138	41	40
Total	**2713**	**913**	**896**

Tabla 3: Distribución de audios por dialecto y partición.

se dividía en varios segmentos según su duración, todos los segmentos pertenecieran a un mismo subconjunto, evitando así fugas de información (*data leakage*). No se hizo un filtrado por usuario, ya que los datos se manejaron de manera anonimizada, por lo que no fue posible agrupar por identidad del hablante (nombre de usuario). La Tabla 3 muestra la distribución de los registros del conjunto de datos según el dialecto y partición de los datos (entrenamiento, validación y test), donde puede observarse el claro desbalance. Por otra parte, el total de entidades recopiladas se muestra en la Tabla 2 junto a su partición de entrenamiento, validación y test.

El entrenamiento del modelo ASR consistió en un *fine-tuning* sobre el decodificador de WhisperNER con LoRA (Hu et al., 2022) durante 1000 pasos, siendo *paso* una actualización de parámetros del modelo, con evaluación en el conjunto de validación cada 40 pasos, empleando la función de pérdida como métrica principal de selección de *checkpoints*. Los adaptadores LoRA se entrenaron congelando los parámetros del modelo base y actualizando únicamente las proyecciones de *query*, *key* y *value* de atención cruzada y autoatención del decodificador, y su configuración consistió en un rango de 16, un factor *alpha* de 16 y un dropout del 10 %; véase en la Figura 3 los adaptadores entrenados.

Todos los experimentos se llevaron a cabo en una GPU Tesla P100, utilizando un tamaño de lote *(batch size)* de 4 con acumulación de gradientes de 8 pasos para simular un tamaño de lote mayor sin superar

la memoria de la GPU disponible. El entrenamiento se realizó utilizando el optimizador *prodigy* (Mishchenko y Defazio, 2024), el cual ajusta dinámicamente el tamaño de paso sin necesidad de un ajuste manual extenso de sus hiperparámetros, convergiendo mucho más rápido. Así mismo, se usó un programador de tasa de aprendizaje de tipo coseno *(cosine scheduler)* para un ajuste dinámico de la tasa de aprendizaje.

El modelo está entrenado para minimizar la función de pérdida *standard cross-entropy* (CE) entre la secuencia de salida predicha $\mathbf{y}$ y la secuencia original $\mathbf{y}^*$, que incluye tanto la transcripción correcta como las etiquetas de entidad correctas. Se puede describir como la Ecuación 1, donde $p_n = P(y_n = y_n^* \mid y_{1:n-1}, \mathbf{h}, \mathbf{p})$. No obstante, se hicieron experimentos usando también la *focal loss* (FL), vista en la Ecuación 2, donde γ en $(1 - p_n)^\gamma$ actúa como un factor modulador que reduce la contribución de ejemplos fáciles y concentra la optimización en los difíciles, y α_n como un factor de ponderación que compensa el desbalance de clases (Huang et al., 2024).

$$\mathcal{L}(y, y^*) = -\sum_{n=1}^{N} \log(p_n), \tag{1}$$

$$\mathcal{L}_{\text{FL}}(p_n) = -\alpha_n (1 - p_n)^\gamma \log(p_n). \tag{2}$$

Para este trabajo se usó $\gamma = 2$ y $\alpha_n = 0{,}25$ para todas las clases. Finalmente, también se entrenó el modelo sobre datos ya censurados, de modo que la predicción se limite a la categoría de la entidad sin incluir su texto correspondiente.

4 Resultados

4.1 Métricas

Para evaluar el rendimiento del método propuesto se adoptaron las métricas estándar utilizadas en el estado del arte para las tareas de ASR y NER, estas son el *Word Error Rate* (WER), y *Character Error Rate* (CER), y *precision*, *recall*, y *F1-Score*, respectivamente. Sin embargo, para NER una evaluación basada únicamente en coincidencias exactas puede ser demasiado estricta, pues no distingue entre diferentes tipos de error -por ejemplo, cuando el modelo detecta correctamente los límites de la entidad pero falla en asignar la categoría adecuada- Debido a esto, siguiendo el esquema de evaluación introducido en SemEval 2013 (Manandhar y Yuret, 2013), los resultados se reportan bajo cuatro modos distintos: Modo *Strict*(S), que busca una coincidencia exacta en los límites de la palabra y el tipo de la entidad, modo *Exact* (E), que busca una coincidencia exacta en los límites de la palabra pero sin importar el tipo de entidad, modo *Partial* (P), que busca una coincidencia parcial de límites, sin importar el tipo, y el modo *Type* (T), que busca cierta superposición entre la entidad etiquetada por el sistema y la entidad original.

Para el caso de la anonimización también se utilizaron *precision*, *recall* y *F1* pero adaptadas a la tarea de censura. En este contexto, un Verdadero Positivo (TP) se considera cuando una palabra sensible (presente como entidad originalmente) ha sido correctamente censurada por el modelo, es decir, no aparece en el texto resultante; un Falso Negativo (FN) ocurre cuando una entidad sensible no fue censurada y sigue visible; un Falso Positivo (FP) se define como cualquier censura adicional que no corresponde a una entidad anotada, independientemente de su tipo o contenido. Esta evaluación no considera si la entidad fue correctamente clasificada ni si su ubicación coincide exactamente con la original, por lo que es más laxa que las métricas tradicionales en tareas de reconocimiento de entidades; el objetivo es verificar si la información sensible fue eliminada del texto, más allá de la forma exacta en que se anotó.

4.2 Evaluación de ASR

En la Tabla 4 se observan los resultados de test para la tarea de ASR en nuestro conjunto de datos para los cuatro estudios de ablación: usando CE (*Cross-Entropy*) y FL (*Focal Loss*) como funciones de pérdida para el etiquetado y anonimización de entidades.

Analizando los resultados, se observa que el uso de FL mejora la transcripción automática, reduciendo WER y CER en casi 1 % respecto a CE, lo que indica una mayor precisión general en el ASR. Sin embargo, la aplicación de censura degrada significativamente el rendimiento, aumentando los errores en la transcripción.

Para complementar el análisis, se compararon los resultados por dialecto en la Tabla 5. El mejor desempeño se observó en San Andrés (WER) y en el dialecto Llanero (CER), posiblemente debido al menor número de audios evaluados. En contraste, los dialectos Andino-Oriental y Andino-Occidental presentaron los mayores errores. Tras revi-

Modelo	WER	CER
CE + NER	8,30	4,55
FL + NER	**7,60**	**3,88**
CE + censura	8,55	6,38
FL + censura	11,63	8,57

Tabla 4: Resultados del estudio de ablación para ASR. Se comparan cuatro configuraciones: CE con NER, FL con NER, CE con censura incluida y FL con censura incluida. Se muestran el WER y el CER para cada configuración. El mejor puntaje se resalta en **negrilla**, mientras que el segundo mejor se indica con subrayado.

Dialecto	WER	CER	# Audios
Llanero	6,64	**2,76**	78
Andino-Oriental	8,28	4,36	361
Andino-Occidental	7,43	3,63	207
Costeño-Atlántico	6,67	2,97	100
Costeño-Pacífico	6,66	3,76	57
San Andrés	**6,22**	3,33	93

Tabla 5: Rendimiento del modelo *FL + NER* para ASR por dialecto, mostrando el número de segmentos evaluados y duración total.

sar la partición de test, estos resultados se asocian principalmente a la presencia de audios de baja calidad y a inconsistencias en las transcripciones de referencia. Esto evidencia la sensibilidad de métricas como WER a la calidad del corpus y resalta la importancia de implementar mecanismos de control y validación en las anotaciones como *inter-rater agreement*.

Finalmente, comparamos el rendimiento de nuestro modelo con otros del estado del arte: WhisperNER estándar y Whisper *large-v2* (Tabla 7). El modelo FL + NER obtiene un WER de 7,60 % y un CER de 3,88 %, superando a ambos enfoques en ASR. El menor rendimiento de WhisperNER puede atribuirse a su ajuste fino exclusivo en inglés, lo que reduce parte de la capacidad multilingüe del modelo base, fenómeno asociado al denominado colapso representativo (Aghajanyan et al., 2021).

Adicionalmente, evaluamos el modelo en un conjunto externo en otro dominio del español (España), FLEURS (Conneau et al., 2022), y comparamos los resultados con sistemas reportados en Open ASR Leaderboard (Srivastav et al., 2023) (Tabla 6). Aunque nuestro modelo (WER = 8,86 %) no supera a sistemas de gran escala como Whisper (*large-v3*), Voxtral o ElevenLabs, quienes fueron

Modelo	WER
Whisper *(large-v3)*	**2,62**
Voxtral Mini *(3VV-2507)*	3,34
ElevenLabs Scribe *(v1)*	7,65
Nuestro *(focal NER)*	8,86

Tabla 6: Comparación del rendimiento ASR sobre el dataset FLEURS (test-es).

Modelo	WER	CER
Whisper *(large-v2)*	8,57	5,75
WhisperNER	18,37	13,76
Nuestro	**7,60**	**3,88**

Tabla 7: Comparación del rendimiento del modelo FL + NER contra WhisperNER y Whisper *large-v2*.

preentrenados con cientos de miles de horas de audio multilingüe, incluyendo la partición de entrenamiento de FLEURS, mantiene un desempeño competitivo considerando la limitada cantidad de datos utilizados en su ajuste y la ausencia de preentrenamiento específico en dicho corpus.

4.3 Evaluación de NER

Con respecto a NER, se observa que aunque la FL ofrece beneficios para ASR, existe un compromiso en la tarea de NER: CE supera consistentemente a FL en las cuatro métricas de evaluación, con resultados ligeramente superiores, visto en la Tabla 8. Para subrayar, el modo que tuvo un mejor rendimiento en ambos modelos fue el *partial*; esto indica que el modelo logra identificar entidades parcialmente, aunque no siempre coincida exactamente con la anotación esperada. Esta diferencia entre modos ejemplifica la sensibilidad de las métricas con errores de segmentación o etiquetado incompleto, comunes en NER. En consecuencia, el uso de los múltiples modos de evaluación permite obtener una visión más variada del comportamiento del modelo, diferenciando entre errores críticos y errores más tolerables.

En la Tabla 9 se muestran los resultados para NER comparándonos con el estado del arte. Nuestro modelo demuestra un rendimiento competitivo, a pesar de no superar a modelos preentrenados en grandes corpus de español como *flair-spanish-large*(Schweter y Akbik, 2020b) —ajustado sobre el subconjunto de datos en español de CoNLL-03 (Tjong Kim Sang, 2002), con 11.758 archivos de texto—, logra superar al estado del

Modelo	Modo	Prec.	Rec.	F1
CE	S	**55,58**	**55,17**	**54,70**
CE	E	**61,86**	**61,51**	**60,81**
CE	P	**63,40**	**63,26**	**62,38**
CE	T	**57,69**	**57,56**	**56,87**
FL	S	53,16	54,28	53,15
FL	E	58,55	60,22	58,67
FL	P	60,20	62,10	60,35
FL	T	55,36	56,74	55,37

Tabla 8: Resultados del estudio de ablación para NER. Se comparan dos configuraciones: Cross entropy y Focal loss. Se muestran *precision*, *recall* y F1 para los 4 modos de evaluación planteados: *Strict*, *Exact*, *Partial* y *Type*.

arte GLINeR multilenguaje 2.1 (Zaratiana et al., 2023) en los cuatro modos de evaluación (*strict*, *exact*, *partial* y *type*). Cabe destacar que Flair y GLINeR son modelos exclusivamente de NER en texto; en nuestro caso, primero obtuvimos las transcripciones utilizando *Whisper-large-v2* y luego aplicamos estos modelos como post-procesamiento sobre el texto generado. El enfoque "todo en uno" de nuestro modelo, juntando ASR y NER mantiene resultados sólidos y consistentes, habiendo sido entrenados $8,028,160$ parámetros, equivalentes al 0,5 % de un total de $1,551,169,280$, logrando un rendimiento competente aún con recursos limitados. Además, esta diferencia de rendimiento frente al modelo de *Flair* puede atribuirse en gran medida a (1) la escasez de datos con entidades utilizados para ajustar nuestro modelo, en contraste con el entrenamiento masivo sobre texto que recibió *Flair-spanish-large*, y (2) la arquitectura de FLERT, basada en modelos tipo codificador como BERT. *Whisper* utiliza un decodificador autoregresivo que solo puede condicionarse en el texto previamente generado, más modelos como *BERT* son bidireccionales, entonces pueden aprovechar simultáneamente el contexto a la izquierda y a la derecha de cada token, y esto les permite un mejor entendimiento del lenguaje.

4.4 Evaluación de anonimización

La Tabla 10 muestra los resultados del estudio de ablación sobre los modelos de censura. Bajo nuestra métrica, ambos modelos obtienen un desempeño muy sobresaliente, especialmente en *recall*, donde logran eliminar la mayoría de las entidades sensibles. No obstante, se observa que ambos sacrifican *precision* en favor de un mayor *recall*, lo que sugiere que tienden a censurar más de la cuenta, posiblemente por el tipo de entrenamiento: al entrenar con transcripciones ya censuradas (por ejemplo <person>> en lugar de <person>Miguel<person>>), el modelo pierde parte del contexto léxico. Esto puede llevarlo a 'suponer' qué entidades deben censurarse, en lugar de apoyarse en la información real, aumentando así la probabilidad de falsos positivos.

Modelo	Modo	Prec.	Rec.	F1
FLERT	S	**74,79**	**73,18**	**73,36**
FLERT	E	**76,22**	**74,61**	**74,77**
FLERT	P	**77,02**	**75,42**	**75,55**
FLERT	T	**76,11**	**74,53**	**74,66**
GLiNER	S	50,67	42,70	44,75
GLiNER	E	53,73	44,99	47,21
GLiNER	P	55,17	46,08	48,34
GLiNER	T	52,57	44,19	46,26
WhisperNER	S	27,93	27,35	25,96
WhisperNER	E	30,26	29,76	27,95
WhisperNER	P	31,38	31,18	29,08
WhisperNER	T	29,45	29,28	27,49
Nuestro	S	55,58	55,17	54,70
Nuestro	E	61,86	61,51	60,81
Nuestro	P	63,40	63,26	62,38
Nuestro	T	57,69	57,56	56,87

Tabla 9: Comparación del rendimiento del modelo CE + NER contra WhisperNER, y modelos NER post-procesamiento FLERT *flair-spanish-large* (Schweter y Akbik, 2020b) y *GLiNER multi-v2.1* (Zaratiana et al., 2023). Se muestran *precision*, *recall* y F1 para los 4 modos de evaluación planteados: *Strict*, *Exact*, *Partial* y *Type*.

Modelo	Precision	Recall	F1
Focal loss	62,02	**85,32**	71,83
Cross entropy	**71,24**	81,68	**76,10**

Tabla 10: Resultados del estudio de ablación para censura. Se comparan CE y FL. Se muestran *precision*, *recall* y F1. No se evalúa el acierto ni la ubicación de las entidades sino la eliminación de información sensible.

Aunque la métrica utilizada es laxa –ya que sólo evalúa si las entidades censuradas aparecen en el texto final y no su clasificación o posición exacta–, los resultados sugieren que los modelos cumplen razonablemente bien su objetivo principal: eliminar información sensible del texto. Destaca especialmente el modelo CE con un F1 de 76.10 %.

4.5 Resultados cualitativos

Para complementar el análisis cuantitativo, se presenta una evaluación cualitativa de los resultados obtenidos por nuestro método Focal NER con censura en las Figuras 4 y 5.

La notación usada es la siguiente: Para la tarea de transcripción se indican en rojo los caracteres o palabras añadidos incorrectamente por el modelo, en azul aquellos que fueron omitidos, y en negro todo lo correcto. En la identificación de entidades, cada tipo se marcó con un color: **“person”** se denota con el color magenta, **“location”** con violeta, **“organization”** con verde y **“miscellaneous”** con naranja. Además, se añadió un indicador *miss* en azul que indica la ausencia de una etiqueta de entidad o una omisión en la censura de una palabra sensible.

Es el impresionante restaurante mexicano en Bucaramanga **LOC** que debes conocer ahora mismo porque realmente te transporta a Ciudad de México **LOC** . Allí manejan los mejores tacos de birria de todo el área metropolitana **LOC** , sin contar su nuevo y exquisito producto, la increíble pizza de birria. Perfecta recomendación para invitar a tu pareja, amigos, familia o a la abuelita cleotilde **MISS** . Ellos son la taquería chula **MISS** , papá. Ubicados específicamente en un parqueadero **LOC** , en la carrera 28 #4968 **LOC** , Barrio Sotomayor **LOC** , no hay pérdida, hay mucho espacio para moto.

Figura 4: Resultados con el modelo FL + NER para ASR y NER.

En audios claros y fluidos como en la Figura 4, el modelo realiza una transcripción casi perfecta, con errores mínimos y pocas omisiones, con un WER $= 0,256$, y un CER $= 0,080$. A pesar de ello, las métricas de NER muestran resultados moderados F1-Score $= 0,33$, ello debido a entidades sin etiquetar y entidades etiquetadas erróneamente, por ejemplo se etiquetó “carrera 28” como *LOCATION*, sin embargo originalmente es una etiqueta *MISCELLANEOUS*. En audios fuera del conjunto de datos y con otras entidades, véase la Figura 5, el modelo demuestra una notable capacidad para generalizar a múltiples entidades gracias a los *prompts* suministrados, siendo *NUMBER* una entidad fuera del conjunto de datos de entrenamiento que se etiquetó luego de especificarla en inferencia.

Mi nombre es **PERSON** , mi número de celular es **NUMBER** . Estoy estudiando biología en la **LOCATION** . Estoy en mi último semestre. Vivo en un conjunto de apartamentos que se llama **LOCATION** , Edificio **LOCATION** , apartamento [NÚMERO APARTAMENTO].

Figura 5: Resultados experimentales con el modelo de censura para ASR y NER.

5 *Conclusiones y trabajo futuro*

En este trabajo, la construcción de un conjunto de datos propio de audio en español colombiano permitió desarrollar un modelo ASR adaptado a su realidad lingüística y dialectal. El modelo alcanzó un 7,60 % en términos de WER, un F1-score de 60,81 % en términos de NER y un F1-score de 76,10 % en anonimización, lo que evidencia que es posible integrar transcripción, NER y censura en un único modelo con resultados competitivos, aún con un conjunto de datos de tamaño limitado. Es importante señalar que, ante la ausencia de recursos públicos disponibles para el dominio del español colombiano, el corpus fue construido mediante técnicas de extracción web, lo cual, si bien permitió realizar el estudio, no sería la alternativa ideal desde un punto de vista ético y resalta la necesidad de crear recursos abiertos. Los experimentos muestran que la elección de diferentes funciones de pérdida influye directamente en el desempeño; como se observó, mejoras en ASR no siempre se traducen en mejoras en NER, y viceversa, un comportamiento previamente reportado (Ayache et al., 2024). Asimismo, la incorporación de censura puede afectar la detección de entidades, lo que abre la puerta a explorar estrategias de entrenamiento conjuntas o funciones de pérdida híbridas. Finalmente, para consolidar un futuro *benchmark* colombiano de ASR + NER, será fundamental ampliar y depurar el conjunto de datos, abordar el desbalance de clases y fortalecer el proceso de anotación mediante múltiples anotadores y la medición de *inter-rater agreement*, garantizando mayor consistencia y confiabilidad en las etiquetas.

Bibliografía

Aghajanyan, A., A. Shrivastava, A. Gupta, N. Goyal, L. Zettlemoyer, y S. Gupta. 2021. Better fine-tuning by reducing re-

presentational collapse. En *International Conference on Learning Representations (ICLR) 2021, Poster*.

Ahlawat, H., N. Aggarwal, y D. Gupta. 2025. Automatic speech recognition: A survey of deep learning techniques and approaches. *International Journal of Cognitive Computing in Engineering*, 6:201–237.

Au, T. W. T., V. Lampos, y I. Cox. 2022. E-NER — an annotated named entity recognition corpus of legal text. En N. Aletras I. Chalkidis L. Barrett C. Goanță, y D. Preoțiuc-Pietro, editores, *Proceedings of the Natural Legal Language Processing Workshop 2022*, páginas 246–255, Abu Dhabi, United Arab Emirates (Hybrid), Diciembre. Association for Computational Linguistics.

Ayache, G., M. Pirchi, A. Navon, A. Shamsian, G. Hetz, y J. Keshet. 2024. Whisperner: Unified open named entity and speech recognition. *arXiv preprint arXiv:2409.08107*.

Bain, M., J. Huh, T. Han, y A. Zisserman. 2023. Whisperx: Time-accurate speech transcription of long-form audio. *INTERSPEECH 2023*.

Banerjee, P. S., B. Chakraborty, D. Tripathi, y others. 2019. A information retrieval based on question and answering and ner for unstructured information without using sql. *Wireless Personal Communications*, 108:1909–1931.

Basak, S., H. Agrawal, S. Jena, S. Gite, M. Bachute, B. Pradhan, y M. Assiri. 2022. Challenges and limitations in speech recognition technology: A critical review of speech signal processing algorithms, tools and systems. *Computer Modeling in Engineering Sciences*, 135:1–37, 10.

Bernal Chávez, Julio Alexander, D. R. C. E. 2017. Caracterización panorámica del español hablado en colombia : fonología y gramática.

Bredin, H., R. Yin, J. M. Coria, G. Gelly, P. Korshunov, M. Lavechin, D. Fustes, H. Titeux, W. Bouaziz, y M.-P. Gill. 2020. pyannote.audio: neural building blocks for speaker diarization. En *ICASSP 2020, IEEE International Conference on Acoustics, Speech, and Signal Processing*, Barcelona, Spain, May.

Comisión de la Verdad. 2024. Transcripciones de entrevistas (anonimizadas). Informe Final - Comisión de la Verdad [Internet]. [cited 1 Aug 2024].

Conneau, A., M. Ma, S. Khanuja, Y. Zhang, V. Axelrod, S. Dalmia, J. Riesa, C. Rivera, y A. Bapna. 2022. Fleurs: Few-shot learning evaluation of universal representations of speech. *arXiv preprint arXiv:2205.12446*.

FFmpeg Developers. 2025. Ffmpeg: multimedia framework. https://ffmpeg.org/.

Hu, E. J., Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, y W. Chen. 2022. LoRA: Low-rank adaptation of large language models. En *International Conference on Learning Representations*.

Huang, Z., L. He, Y. Yang, y others. 2024. Application of machine reading comprehension techniques for named entity recognition in materials science. *Journal of Cheminformatics*, 16(76).

Keraghel, I., S. Morbieu, y M. Nadif. 2024. Recent advances in named entity recognition: A comprehensive survey and comparative study.

López, R. R., A. S. L. Cortés, y N. S. Guzmán. 2015. Corpus lingüísticos del instituto caro y cuervo (clicc). *Linguamática*, 15:89–96.

Manandhar, S. y D. Yuret, editores. 2013. *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Atlanta, Georgia, USA, Junio. Association for Computational Linguistics.

Mancera, G. y others. 2026. Pba-llm: Privacy- and bias-aware nlp using named-entity recognition (ner). En L. Jin R. Zanibbi, y V. Eglin, editores, *Document Analysis and Recognition – ICDAR 2025 Workshops*, volumen 16225 de *Lecture Notes in Computer Science*. Springer, Cham.

Mishchenko, K. y A. Defazio. 2024. Prodigy: An expeditiously adaptive parameter-free learner. En R. Salakhutdinov Z. Kolter

K. Heller A. Weller N. Oliver J. Scarlett, y F. Berkenkamp, editores, *Proceedings of the 41st International Conference on Machine Learning*, volumen 235 de *Proceedings of Machine Learning Research*, páginas 35779–35804. PMLR, 21–27 Jul.

Moskvitch, K. 2017. The machines that learned to listen. https://www.bbc.com/future/article/20170214-the-machines-that-learned-to-listen. Último acceso: 4 de marzo de 2026.

Qin, L., Q. Chen, X. Feng, Y. Wu, Y. Zhang, Y. Li, M. Li, W. Che, y P. S. Yu. 2024. Large language models meet nlp: A survey.

Radford, A., J. W. Kim, T. Xu, G. Brockman, C. Mcleavey, y I. Sutskever. 2023. Robust speech recognition via large-scale weak supervision. En *Proceedings of the 40th International Conference on Machine Learning*, volumen 202 de *Proceedings of Machine Learning Research*, páginas 28492–28518. PMLR, 23–29 Jul.

Rista, A. y A. Kadriu. 2020. Automatic speech recognition: A comprehensive survey. *SEEU Review*, 15:86–112, 12.

Roha, V. S., N. Saini, S. Saha, y J. G. Moreno. 2023. Moo-cmds+ner: Named entity recognition-based extractive comment-oriented multi-document summarization. En J. Kamps y others, editores, *Advances in Information Retrieval*, volumen 13981 de *Lecture Notes in Computer Science*. Springer, Cham.

Schweter, S. y A. Akbik. 2020a. Flert: Document-level features for named entity recognition.

Schweter, S. y A. Akbik. 2020b. Flert: Document-level features for named entity recognition.

SeleniumHQ. 2025. Selenium webdriver. https://www.selenium.dev/.

Srivastav, V., S. Majumdar, N. Koluguri, A. Moumen, S. Gandhi, y others. 2023. Open automatic speech recognition leaderboard. https://huggingface.co/spaces/hf-audio/open_asr_leaderboard.

Team, S. 2024. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad.

Tjong Kim Sang, E. F. 2002. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. En *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Yadav, H. y S. Sitaram. 2022. A survey of multilingual models for automatic speech recognition. En *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, páginas 5071–5079, Marseille, France, Junio. European Language Resources Association.

yt-dlp Project. 2025. yt-dlp: A feature-rich command-line audio/video downloader. https://github.com/yt-dlp/yt-dlp.

Zaratiana, U., N. Tomeh, P. Holat, y T. Charnois. 2023. Gliner: Generalist model for named entity recognition using bidirectional transformer.

Zeroual, I. y A. Lakhouaja. 2018. Data science in light of natural language processing: An overview. *Procedia Computer Science*, 127:82–91. Proceedings of the first international conference on Intelligent Computing in Data Sciences, ICDS2017.