Desarrollo de un sistema de reconocimiento del habla en guaraní

Evaluación de variantes del modelo Whisper y técnicas de mejora de datos

Santiago Rubén Acevedo Zarza, Mateo Andrés Fidabel Gill,
Christian Daniel von Lücken Martínez, Diego Pedro Pinto Roa

Facultad Politécnica, Universidad Nacional de Asunción
{py.santi,mateofidabel}@fpuna.edu.py, {clucken,dpinto}@pol.una.py

Resumen El idioma guaraní es uno de los lenguajes autóctonos más hablados de América del Sur, y es utilizado por la mayoría de la población del Paraguay. Sin embargo, se encuentra poco representado en conjuntos de datos utilizados para el entrenamiento de modelos multilenguaje, por lo que existen pocas herramientas lingüísticas basadas en deep learning que sean compatibles con el guaraní. Este proyecto tiene como objetivo desarrollar un sistema de reconocimiento capaz de transcribir voz en guaraní y ponerlo a disposición del público.
El primer paso es llevar a cabo un análisis preliminar para identificar un criterio óptimo de selección de datos de entrenamiento y comparar el rendimiento de modelos de diferentes tamaños. Este estudio evalúa los siguientes criterios: el uso de una muestra pequeña de datos manualmente verificada, una muestra más grande con exactitud sin verificar, y una combinación de ambos enfoques mediante aprendizaje auto-supervisado. Todos los datos provienen del corpus de Mozilla Common Voice, y los modelos fueron entrenados a partir de diferentes versiones multilenguaje de Whisper. Encontramos que incluir una muestra más grande de datos sin verificar mejora drásticamente la precisión del modelo final, y que el aprendizaje auto-supervisado no mejora la precisión con respecto al modelo inicial.

Palabras claves: idioma guaraní, reconocimiento automático del habla, aprendizaje semi-super visado, low-rank adaptation

1. Introducción

El guaraní es un idioma originario de América del Sur, perteneciente a la familia tupí-guaraní, y hablado por alrededor de 6.5 millones de personas [1]. Es hablado principalmente en la República del Paraguay, donde tiene carácter oficial, y en algunas regiones de países vecinos. La mayoría de la población del Paraguay es bilingüe en español y guaraní o habla un dialecto llamado *jopara*, el cual incorpora elementos de ambos idiomas, y una fracción significativa de la población rural solamente habla guaraní [2].

A pesar de ser tan difundido alrededor de la región, el guaraní es considerado un lenguaje de escasos recursos ya que existen pocos datos disponibles con los cuales entrenar sistemas de aprendizaje automático [3]. Debido a que el rendimiento de este tipo de sistemas depende en gran medida de qué tan similares sean los datos de entrenamiento a los datos que deberá inferir, esta escasez significa que existen pocas herramientas de transcripción automática que sean compatibles con el guaraní. Por lo tanto, resulta de interés social el desarrollo de sistemas especializados para este lenguaje, ya que permitirá que las comunidades que lo hablan aprovechen los últimos avances tecnológicos en esta área.

Este trabajo representa la primera parte de un esfuerzo para crear un sistema de transcripción automática de voz especializado para su uso con el idioma guaraní. El objetivo de esta fase es obtener un conjunto de datos de entrenamiento y seleccionar una arquitectura apropiada. Decidimos utilizar el corpus en guaraní de *Common Voice* porque posee una licencia permisiva, junto con el modelo Whisper debido a que fue pre-entrenado con múltiples lenguajes y se encuentra disponible en varios tamaños. Realizamos una serie de experimentos para determinar si es más efectivo utilizar un subconjunto de los datos o aplicar una técnica de aumento de datos, y evaluar qué tamaño de modelo es más apropiado. El proceso de entrenamiento fue optimizado mediante el uso de parámetros LoRA [4] e incluye una fase de cuantización LLM.int8() [5].

De lo que resta de este trabajo el mismo se organiza de la siguiente manera. La sección II establece un marco teórico y describe los conceptos más importantes de este trabajo. La sección III habla de estudios previos en esta misma área, y la sección IV describe la metodología utilizada para este experimento. Finalmente, la sección V presenta los resultados obtenidos y explica las conclusiones más relevantes sobre este trabajo.

2. Preliminares

2.1. Reconocimiento automático del habla

El reconocimiento automático del habla (ASR, *automatic speech recognition*) es una rama de la inteligencia artificial enfocada en desarrollar sistemas computacionales capaces de procesar y transcribir la voz humana, generando un texto correspondiente [6]. Desarrollar estos sistemas requiere de un conjunto de datos de entrenamiento, el cual consiste en muestras de audio de voz junto con sus transcripciones correspondientes. El proceso de entrenamiento involucra extraer ciertas características de las muestras de audio y encontrar patrones que permitan relacionarlas con las transcripciones recibidas. El uso de un conjunto de datos representativo es de vital importancia, debido a que determina qué tipo de voces el sistema podrá reconocer y la precisión que este tendrá.

Las dos técnicas más utilizadas para implementar sistemas ASR son los modelos ocultos de Márkov y las redes neuronales; estas últimos pueden pueden ser de extremo a extremo (E2E, *end-to-end*) o híbridas. Los modelos E2E utilizan solamente redes neuronales para convertir el audio a texto, mientras que

los modelos híbridos se asisten de un modelo de lenguaje para asistir con la predicción, el cual generalmente es un modelo oculto de Márkov. El éxito y la rápida adopción de los modelos aprendizaje profundo convirtieron al E2E en el enfoque predominante [7]; sin embargo, estos tienen un rendimiento bastante pobre cuando los datos de entrenamiento son escasos.

2.2. Common Voice

El proyecto *Common Voice* fue fundado por la comunidad Mozilla con el fin de crear una base de datos lingüísticos de acceso gratuito para el desarrollo de sistemas ASR [8]. Las muestras de voz son grabadas y enviadas por voluntarios, quienes también verifican la validez de muestras enviadas por otros usuarios. Las grabaciones de voz junto con sus transcripciones se encuentran disponibles bajo una licencia de dominio público, garantizando que los desarrolladores puedan utilizarlas para desarrollar sus modelos sin costos o restricciones.

La versión más reciente del corpus de *Common Voice* contiene alrededor de 30.000 horas de audio con sus respectivas transcripciones y abarca 120 lenguajes, incluyendo el guaraní. Sin embargo, solamente un 63 % del corpus tuvo su validez verificada (cerca de 19.000 horas), siendo este porcentaje altamente variable entre distintos idiomas. En el caso del guaraní, se cuentan con 26 horas de audio de las cuales 13,5 % se encuentran validadas (3,5 horas). Algunas de las muestras no verificadas contienen transcripciones erróneas o pistas de audio dañadas, las cuales pueden impactar la precisión de un modelo entrenado con ellas.

2.3. Modelo Whisper

Whisper es un sistema ASR pre-entrenado a partir de 680 000 horas de audio con transcripciones, abarcando múltiples lenguajes [9] . La arquitectura de Whisper es E2E y está implementada como un transformador codificador-decodificador. El codificador recibe un espectrograma log-Mel con una longitud máxima de 30 segundos, mientras que el decodificador es entrenado para predecir la transcripción correspondiente.

Existen diferentes versiones de Whisper con diferentes tamaños, donde la más pequeña posee 9 millones de parámetros y la más grande posee 1.550 millones. Casi todos los tamaños de Whisper se encuentran disponibles en versiones plurilingües y entrenadas solamente en inglés.

2.4. Low Rank Adaptation

La adaptación de bajo rango (LoRA, *Low-Rank Adaptation*) es una técnica que permite tomar un modelo pre-entrenado para resolver un problema de manera general y adaptarlo para mejorar su rendimiento en una tarea específica [4]. Esto se hace mediante el uso de representaciones de menor dimensionalidad en vez de actualizar los pesos directamente, se generan pares de matrices entrenables para cada capa, también conocidos como parámetros LoRA. La suma entre

los pesos iniciales de una capa y el producto del par de matrices correspondiente produce los pesos actualizados para esa capa.

Debido a que cada par de matrices tiene en total significativamente menos elementos que su producto (el cual tiene las mismas dimensiones que la matriz de pesos que actualiza), los parámetros LoRA son considerados una representación de bajo rango. El uso de esta representación permite actualizar todos los pesos de un modelo entrenando un menor número de parámetros, lo cual reduce el costo computacional de adaptar modelos grandes.

Otra ventaja de esta técnica es que si se desea adaptar un mismo modelo para dos o más tareas diferentes, es posible almacenar solamente el modelo original y los parámetros LoRA. Cuando se desee utilizar un modelo adaptado, sólo será necesario calcular las matrices de pesos actualizadas y el modelo se podrá ejecutar sin ningún costo adicional en la fase de inferencia.

2.5. LLM.int8()

LLM.int8() es un procedimiento de cuantización que permite reducir a la mitad la memoria utilizada en el proceso de inferencia sin degradar la precisión del modelo [5]. Esta técnica fue diseñada para optimizar modelos de lenguaje grandes con varios billones de parámetros, y depende de ciertas propiedades emergentes de los transformadores.

El primer paso consiste en una cuantización vectorial de las capas que involucren multiplicaciones entre matrices, utilizando una constante de normalización propia para cada producto interno entre una fila y una columna. El segundo involucra una descomposición de precisión mixta en las dimensiones que contengan características atípicas. En las capas ocultas de un transformador suelen existir ciertos parámetros con una magnitud mucho mayor al resto, y cualquier error de redondeo en estos parámetros suele tener un impacto muy drástico en la precisión del modelo. Al utilizar multiplicación de matrices de 16 bits en las dimensiones que contengan estos valores, es posible cuantizar las dimensiones restantes sin degradar el rendimiento del modelo.

3. Reseña de la literatura

3.1. Reconocimiento automático del habla en Guaraní

El único otro sistema ASR para Guaraní que encontramos en la literatura fue Eñe'e, desarrollado por Maldonado et al. [10]. Este fue implementado utilizando la herramienta CMU Sphinx y está basado en modelos ocultos de Márkov. Es un modelo híbrido debido a que no sólo analiza una entrada acústica para obtener los fonemas correspondientes, sino que también utiliza un modelo de lenguaje para predecir qué palabra tiene mayor probabilidad de ocurrir teniendo en cuenta la secuencia de palabras anteriores.

Los autores de este trabajo recopilaron y verificaron manualmente un conjunto de datos de entrenamiento de 1.000 oraciones leídas por 114 hablantes

distintos, alcanzando un total de 54 minutos de entrenamiento. La tasa de error de palabras (WER) obtenida fue del 9,29 %, y el sistema contó con un vocabulario 678 palabras.

3.2. Aumento de datos de entrenamiento

El estudio de Laptev et al. (2020) evaluó la efectividad del aumento de datos mediante TTS y aprendizaje semi-supervisado [11]. Este concluye que partiendo de un subconjunto de datos etiquetados de 100 horas de LibriSpeech, al generar 360 horas de datos sintéticos, estas técnicas aumentan considerablemente la precisión del modelo de reconocimiento automático del habla. El modelo inicial sin ninguna técnica logró un WER de 10.3 % en el conjunto test-clean de LibriSpeech, mientras que el aumento de datos mediante TTS logró un WER de 6.3 % y el aprendizaje semi-supervisado logró uno de 6.2 %.

El estudio de Bartelds et al. (2023) evaluó la efectividad del aprendizaje semi-supervisado para cuatro lenguajes topológicamente diferentes, cada uno contando con solo 192 minutos de datos de entrenamiento [12]. Se aplicó la técnica a subconjuntos etiquetados de 24, 28, y 96 minutos, sintetizando transcripciones para aquellos datos que no fueron incluidos en cada subconjunto. En casi todos los casos, el aprendizaje semi-supervisado mejoró la precisión del modelo resultante con respecto al modelo inicial, pero no tanto como aumentar manualmente el número de datos etiquetados manualmente.

4. Metodología

4.1. Selección de los datos

El objetivo de este experimento es comparar el desempeño de modelos entrenados tras utilizar tres criterios de selección de datos. El primer criterio consiste en utilizar solamente datos verificados. El segundo criterio involucra utilizar estos datos para entrenar un modelo intermedio, y a partir de este inferir las transcripciones para las muestras de audio sin verificar; estos datos luego serán adjuntos a los datos verificados. Finalmente, el tercer criterio consistirá en usar tanto datos verificados como sin verificar.

El Cuadro 1 indican qué splits se utilizan en cada criterio y el número de pares de audio y transcripciones. El split *transcribed* utiliza las mismas muestras de audio que *other* pero con transcripciones inferidas por un modelo entrenado utilizando el primer criterio. Los otros splits (*train*, *validation*, y *other*) forman parte del corpus de Common Voice. Adicionalmente, se utiliza el split *test* para evaluar la precisión de los modelos.

4.2. Pre-procesamiento

Los datos de audio de Common Voice tienen una frecuencia de muestreo de 42 kHz, mientras que Whisper recibe audio con una frecuencia de 16kHz, por lo

Cuadro 1. Criterios de selección de datos

Criterio	Splits	# Muestras
datos validados	*train*	1571
	validation	360
auto-supervisado	*train*	1571
	validation	360
	transcribed	18779
sin validar	*train*	1571
	validation	360
	other	18779

que es necesario un proceso de remuestreo antes de trabajar con los datos. Cada fragmento de audio es rellenado hasta alcanzar una duración de 30 segundos y luego convertido a un espectrograma Mel de magnitud logarítmica.

4.3. Proceso de entrenamiento

Una vez seleccionado el conjunto de datos, el modelo a entrenar se prepara para el proceso de cuantización. Se utiliza la técnica LLM.int8() para reducir el costo computacional sin impactar la precisión, y también se aplica la técnica LoRA para reducir significativamente el número de parámetros entrenables. Finalmente, el modelo se entrena durante 5 épocas (*epochs*) con un tamaño de lote (*batch size*) de 16.

Los hiper-parámetros utilizados para LoRA son $r = 32$, $\alpha = 64$, y una probabilidad de dilución (*dropout*) del 5 %. El Cuadro 2 indica el número de parámetros totales y entrenables para cada variante de Whisper.

Cuadro 2. Número de parámetros

Variante	# Parámetros	# Entrenable	% Entrenable
whisper-tiny	39M	590K	1.54 %
whisper-small	244M	3M	1.44 %
whisper-medium	769M	9M	1.22 %
whisper-large-v2	1550M	15M	1.00 %
whisper-large-v3	1550M	15M	1.00 %

4.4. Evaluación de los modelos

Este proceso de entrenamiento fue realizado para cada combinación de criterio de selección de datos y variante de modelo. Los resultados luego fueron evaluados midiendo la tasa de error de palabras (WER, *word error rate*). Esta

métrica mide la fracción de palabras erróneas en la transcripción obtenida, y el estándar utilizado para evaluar sistemas ASR [13]. La fórmula utilizada para medir el WER es la siguiente:

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

Donde I es el número de palabras insertadas, D es el número de palabras borradas, S es el número de palabras substituidas, y N es el número de palabras en la transcripción de referencia.

4.5. Entorno de ejecución

Los experimentos fueron ejecutados en una máquina virtual proveída por Google Compute Engine y equipada con una NVIDIA L4. Las librerías utilizadas para implementar las técnicas fueron PyTorch, PEFT y Transformers de la compañia Hugging Face.

5. Resultados y conclusiones

El Cuadro 3 muestra la tasa de errores obtenida al final de cada prueba. Para todas las versiones de Whisper evaluadas, el modelo auto-supervisado obtuvo la misma precisión o empeoró con respecto al modelo intermedio. Este efecto ocurrió independientemente del tamaño y de la precisión utilizado. Para todos los tamaños, observamos que la inclusión de los datos sin validar aumenta significativamente la precisión del modelo final, incluso cuando las transcripciones de estos datos no son completamente exactas.

La variante de modelo que obtuvo el mejor desempeño fue *whisper-large-v3*. Este posee la misma arquitectura que *whisper-large-v2*, pero fue pre-entrenado con un conjunto de datos más grande y variado. Esto sugiere que no solo es importante elegir una arquitectura adecuada, sino también partir de un modelo que haya sido generado a partir de un conjunto de datos diverso.

Concluimos que para este problema, la técnica de entrenamiento auto-supervisado no es un método efectivo para mejorar la precisión de un modelo, y que la transcripción y verificación manual de las muestras permiten un aprovechamiento mejor de los datos.

La siguiente fase de este proyecto estará enfocada en optimizar el proceso de entrenamiento para mejorar el rendimiento obtenido con *whisper-large-v3*, y en construir una aplicación que se pueda disponibilizar al público.

Disclosure of Interests. Los autores no tienen conflictos de interés a declarar que sean relevantes al contenido de este artículo.

Cuadro 3. Comparación de tasas de errores de palabras

Variante del modelo	Criterio de selección	WER
whisper-tiny	datos validados	82.09 %
	auto-supervisado	84.45 %
	sin validar	**52.36 %**
whisper-small	datos validados	53.29 %
	auto-supervisado	63.80 %
	sin validar	**26.65 %**
whisper-medium	datos validados	65.27 %
	auto-supervisado	65.47 %
	sin validar	**18.53 %**
whisper-large-v2	datos validados	58.25 %
	auto-supervisado	58.08 %
	sin validar	**21.24 %**
whisper-large-v3	datos validados	54.72 %
	auto-supervisado	57.94 %
	sin validar	**13.77 %**

Referencias

1. D. M. Eberhard, G. F. Simons, and C. D. Fennig, *Ethnologue: Languages of the World*, 23rd ed. SIL International, 2020. [Online]. Available: https://www.ethnologue.com/24/language/gug
2. B. Estigarribia, "Guaraní-spanish jopara mixing in a paraguayan novel: Does it reflect a third language, a language variety, or true codeswitching?" *Journal of Language Contact*, vol. 8, no. 2, pp. 183 – 222, 2015. [Online]. Available: https://brill.com/view/journals/jlc/8/2/article-p183_2.xml
3. L. Chiruzzo, M. Agüero-Torales, A. Alvarez, and Y. Rodríguez, "Initial experiments for building a Guarani WordNet," in *Proceedings of the 12th Global Wordnet Conference*, G. Rigau, F. Bond, and A. Rademaker, Eds. University of the Basque Country, Donostia - San Sebastian, Basque Country: Global Wordnet Association, Jan. 2023, pp. 197–204. [Online]. Available: https://aclanthology.org/2023.gwc-1.24
4. E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," 2021. [Online]. Available: https://arxiv.org/abs/2106.09685
5. T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer, "Llm.int8(): 8-bit matrix multiplication for transformers at scale," 2022. [Online]. Available: https://arxiv.org/abs/2208.07339
6. J. Levis and R. Suvorov, *Automatic Speech Recognition*. John Wiley & Sons, Ltd, 2020, pp. 1–8. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781405198431.wbeal0066.pub2
7. R. Prabhavalkar, T. Hori, T. N. Sainath, R. Schlüter, and S. Watanabe, "End-to-end speech recognition: A survey," 2023. [Online]. Available: https://arxiv.org/abs/2303.03329
8. R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," 2020. [Online]. Available: https://arxiv.org/abs/1912.06670

9. A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, "Robust speech recognition via large-scale weak supervision," 2022. [Online]. Available: https://arxiv.org/abs/2212.04356
10. D. M. Maldonado, R. Villalba Barrientos, and D. P. Pinto-Roa, "Eñe' e: Sistema de reconocimiento automático del habla en guaraní," in *Simposio Argentino de Inteligencia Artificial (ASAI 2016)-JAIIO 45 (Tres de Febrero, 2016).*, 2016. [Online]. Available: https://sedici.unlp.edu.ar/handle/10915/56979
11. A. Laptev, R. Korostik, A. Svischev, A. Andrusenko, I. Medennikov, and S. Rybin, "You do not need more data: Improving end-to-end speech recognition by text-to-speech data augmentation," in *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. IEEE, Oct. 2020. [Online]. Available: http://dx.doi.org/10.1109/CISP-BMEI51763.2020.9263564
12. M. Bartelds, N. San, B. McDonnell, D. Jurafsky, and M. Wieling, "Making more of little data: Improving low-resource automatic speech recognition using data augmentation," 2023. [Online]. Available: https://arxiv.org/abs/2305.10951
13. A. Ali and S. Renals, "Word error rate estimation for speech recognition: e-WER," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, I. Gurevych and Y. Miyao, Eds. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 20–24. [Online]. Available: https://aclanthology.org/P18-2004