# SELECCIÓN DE UNA HERRAMIENTA DE RECONOCIMIENTO DE VOZ ANALIZANDO SUS CARACTERÍSTICAS

## *SELECTION OF A SPEECH RECOGNITION TOOL BY NALYZING ITS FEATURES*

***Salvador M. Malagón Soldara***
Tecnológico Nacional de México / IT de Celaya, México
*salvador.malagon@itcelaya.edu.mx*

***Daniela E. Campos Camacho***
Tecnológico Nacional de México / IT de Celaya, México
*19030528@itcelaya.edu.mx*

***Cesar A. Molina Guzmán***
Tecnológico Nacional de México / IT de Celaya, México
*19030791@itcelaya.edu.mx*

***Jorge L. Torres Ramírez***
Tecnológico Nacional de México / IT de Celaya, México
*18031678@itcelaya.edu.mx*

***José Luís Hurtado Chávez***
Tecnológico Nacional de México / IT de Celaya, México
*luis.hurtado@itcelaya.edu.mx*

**Recepción:** 3/octubre/2023 **Aceptación:** 30/noviembre/2023

## Resumen

En el presente trabajo se describe una investigación para lograr la selección de una librería con detección de voz. Donde, la aplicación de esta librería será el emitir comandos para un asistente inteligente de domótica. Por lo tanto, para evitar la programación de un algoritmo detector de palabras, se analizaron las siguientes cinco opciones. En primer lugar, se probó la librería Neural Intents en conjunto con la librería de reconocimiento de voz en Python, éstas en conjunto brindan la capacidad tanto de ofrecer respuestas a comandos de voz como de detectar estados de ánimo e intenciones del usuario. En segundo lugar, se utilizó la librería PyTorch, la cual se utilizó para entrenar el reconocimiento de voz de comandos

mediante redes neuronales. En tercer lugar, se utilizó la librería Text To Speech de Google, la cual es una librería que nos permite no sólo tener un método de reconocimiento de voz, sino que también logra que la asistente inteligente pueda hablar. En cuarto lugar, se analizó Google Assistant Flask, el cual es un chatbot basado en Dialogflow que trabaja sobre el lenguaje Python. En quinto y último lugar, se tiene la librería Python Speech Recognition la cual es una librería para reconocimiento de voz. De esta manera, la selección del asistente se realizó por medio de cuatro criterios: dificultad de uso, documentación online, soporte al usuario y un uso offline. Para finalizar, basados en estos criterios se eligió Speech Recognition y se expone un ejemplo.

**Palabras Clave:** reconocimiento de voz, inteligencia artificial, domótica.

## *Abstract*

*This paper describes an investigation to achieve the selection of a library with voice detection. Where, the application of this library will be to emit commands for an intelligent home automation assistant. Therefore, in order to avoid programming a word-detecting algorithm, the following five options were analyzed. First, the Neural Intents library was tested in conjunction with the Python speech recognition library, which together provide the ability to both provide responses to voice commands and detect user moods and intentions. Secondly, the PyTorch library was used, which was used to train the speech recognition of commands using neural networks. Thirdly, Google's Text To Speech library was used, which is a library that not only allows us to have a speech recognition method, but also enables the intelligent assistant to speak. In fourth place, Google Assistant Flask was analyzed, which is a chatbot based on Dialogflow that works on the Python language. In fifth and last place, we have the Python Speech Recognition library, which is a library for speech recognition. Thus, the selection of the assistant was based on four criteria: difficulty of use, online documentation, user support and offline use. Finally, based on these criteria, Speech Recognition was chosen, and an example is presented.*

***Keywords:*** *voice recognition, artificial intelligence, home automation.*

## 1. Introducción

El reconocimiento de voz implica que una máquina interprete y transforme el habla en un formato entendible para la tecnología. No obstante, aunque la interacción con asistentes digitales parece sencilla, el proceso es intrincado. Los humanos parecen tener una habilidad innata para escuchar y comprender, sin embargo, sigue en constante mejora durante toda la vida. Así mismo, la tecnología de reconocimiento de voz opera de forma análoga, se debe entrenar a las computadoras de manera similar implicando: creatividad, dedicación y estudio. El reconocimiento de voz robusto afronta desafíos entre el entrenamiento y las pruebas, necesitando mejoras para adaptarse a variaciones en altavoces, tipos de micrófonos, direcciones, posiciones, canales de transmisión y ambientes acústicos. Los micrófonos actúan como filtros en la señal de voz, y se debe tener en cuenta la pendiente espectral según el tipo de micrófono. Esta distorsión está relacionada con la señal del habla, además de la distancia entre el micrófono y el hablante [González, 2020].

Dentro del marco de los análisis de antecedentes en el reconocimiento de voz, se encontró que Cruz Montealegre [2021] creó y entrenó una red neuronal para un asistente. Dicha tecnología identificaba qué persona se comunica con ella, con base en las herramientas gratuitas de Spyder y Google Colab. Por otro lado, Vera [2021] se dio a la tarea de crear un algoritmo que pudiera evaluar de manera oral a los alumnos de cualquier materia que el docente desee, agilizando así el proceso de evaluación además de entregar una calificación inmediata al estudiante. Para lograrlo, utilizó el módulo PocketSphinx como su principal herramienta. Por otro lado, pensando en la inclusión, Villa [2017] desarrolló una interfaz que traducía el lenguaje de señas, con ayuda de la herramienta peech-Recognizer para la conversión de voz a texto. En adición usó Sklearn y NLTK para el procesamiento del mismo y MySQL entregaba como resultado las imágenes del lenguaje de señas guardadas en una base de datos. Para finalizar, Navarro [2020] utilizó la herramienta Speech Recognizer para crear un algoritmo capaz de preguntarle al usuario ¿qué canción desea escuchar? De esta manera, recibía respuesta y con base en ella reproducía esa melodía en cualquier plataforma de streaming musical.

A continuación, en el presente artículo se revisarán 5 tipos diferentes de tecnologías para emplearlas como reconocimiento de voz. Además, se buscará llevar a cabo un asistente de voz orientado a adultos mayores y su uso en la domótica. Por último, en los resultados se menciona que la tecnología más adecuada para utilizar es la librería Speech Recognition. De esta manera, dentro del artículo se mencionan los beneficios que esta herramienta presenta para llevar a cabo el asistente de voz, así como sus características generales y sus mayores ventajas para emplearlo.

## 2. Métodos

### Asistente Intelligent Voice en Python

Esta librería de reconocimiento de voz en Python [Amos, 2016] se usa para que el usuario entienda qué está diciendo el asistente de voz y, a su vez, él entienda que quiere decir el usuario. Además de dictarse diferentes comandos, los cuales, ayudan a dar respuestas determinadas o, podría decirse, ya programadas, como lo es un: “hola”. De esta forma, el asistente reconocerá que se le está saludando, por lo cual, él debe de contestar algo similar, en este caso algo ya predeterminado. Es importante aclarar que este tipo de asistentes ya cuentan con un cierto entrenamiento previo, el cual, se usa y manda llamar con el uso de diferentes librerías.

La diferencia del uso de la librería Intents con otros métodos similares es que esta librería permite brindarle una mayor inteligencia e independencia de la conexión a internet al asistente [Karumuri, 2022]. Así, como detectar las intenciones o estados de ánimo que tenga el usuario al hablar. Estas tres herramientas se usan para construir bots de chats y asistentes inteligentes.

### PyTorch

Se trata de una librería basada en Python, diseñada para realizar cálculos numéricos haciendo uso de la programación de tensores [Imambi, 2021]. Además, esta librería permite su ejecución en GPU para acelerar los cálculos. También, dispone de una interfaz muy sencilla para la creación de redes neuronales (RN). Pese a trabajar de forma directa con tensores, no tiene la necesidad de una librería

a un nivel superior, como puede ser Keras, Theano o Tensorflow [Torres, 2020]. Entre otras características, cuenta con un soporte para su ejecución en tarjetas gráficas (GPU), utiliza internamente CUDA y cuenta con una API desarrollada por NVIDIA que conecta la CPU con la GPU.

Otra de las ventajas que tiene PyTorch es su compatibilidad con Python, siendo éste el lenguaje que suele ser más utilizado para el desarrollo de Machine Learning e inteligencia artificial [Ketkar, 2021]. De igual manera, es un lenguaje fácil de aprender, permite depuración con muchas herramientas de Python, además de permitir cambiar el comportamiento de la RN en tiempo de ejecución, mejorando la optimización y los resultados. Adicionalmente, tiene una distribución sencilla en múltiples cores de CPU o GPU y, al tener una comunidad bastante activa, cuenta con abundante documentación.

Sin embargo, tiene algunas desventajas en ciertas áreas como es el área de producción. Sus interfaces de visualización y supervisión están limitadas pues es necesario utilizar herramientas de visualización de datos de Python o conectarse externamente a TensorBoard. Por último, no es una herramienta de desarrollo de extremo a extremo, de manera que sólo se usa para algunos sectores del código.

**Google Text to Speech**

Un sistema Text to Speech es una aplicación que convierte un texto escrito a audio, permitiendo escuchar cualquier texto o entradas de lenguaje de marcación de síntesis de voz (SSML) convirtiéndolos en datos de audio como MP3 o LINEAR16 [Rithika, 2016]. Esta librería también permite crear voces humanas sintéticas con un sonido natural con audio reproducible. Además, se pueden usar los archivos de datos de audios creados con Text to Speech para potenciar las aplicaciones o mejorar los medios como videos o grabaciones de audio. Por lo tanto, Text to Speech es ideal para cualquier aplicación que reproduzca audio de voz humana a los usuarios. Permite convertir strings, palabras y oraciones arbitrarias en la voz de una persona.

Además de ser utilizado para síntesis (convertir texto en audio) y generar voces humanas artificiales, se pueden configurar otros aspectos de los resultados de datos

de audio que se crean con la síntesis de voz, como la configuración de la velocidad de la voz, el tono, el volumen y la tasa de muestreo en hercios.

Dentro de las ventajas de estas aplicaciones es que ayuda al usuario a ahorrar tiempo aumentando así la productividad en el trabajo y mayor movilidad. La desventaja que presenta es que se trata de una aplicación ya desarrollada, por lo que no permite muchas modificaciones al programa, sino que únicamente te rentan el poder utilizarla.

**Google Assistant Flask**

Flask es un micro-Framework [Salvi, 2019] escrito en Python y concebido para facilitar el desarrollo de aplicaciones web bajo el patrón MVC (modelo-vista-controlador). Se denomina “micro” porque al instalar Flask se tienen las herramientas necesarias para crear una aplicación web funcional, pero si se necesita en algún momento una nueva funcionalidad, hay un conjunto muy grande de extensiones (plugins) que se pueden instalar para dotarlo de nuevas funcionalidades. Además, al tratarse de un Framework es una herramienta que facilita y abstrae la construcción de páginas web dinámicas [Ríos, 2016]. Por otro lado, el patrón MVC es una manera o forma de trabajar que permite diferenciar y separar lo que es el modelo de datos (los datos que va a tener la app normalmente están guardados en una base de datos), la vista (página HTML) y el controlador (donde se gestionan las peticiones de la app web) [Sabharwal, 2020].

Utilizar la herramienta Flask trae consigo diferentes ventajas y desventajas, entre las ventajas está el alcance, pues difícilmente existe un Framework más ágil, además de que es rápido de instalar y utilizar, sin embargo, esta misma puede presentarse como desventaja, puesto que, dependiendo de la utilidad que se le quiera dar, será necesario instalar herramientas por separado, mientras que otros Frameworks ofrecen muchas más funciones preinstaladas, esto mismo trae como desventaja la dependencia con proveedores de terceros.

Otra ventaja de Flask es que, ofrece una flexibilidad extraordinaria. Puede resolver problemas e implementar las bibliotecas que se necesiten, abordando cada proyecto de forma individual. Así mismo, es fácil de aprender a utilizar con un

tutorial. El Framework es deliberadamente sencillo, pero puede utilizarse igualmente para proyectos exigentes, siendo una gran opción para principiantes y expertos. Como última desventaja a mencionar se tiene el mantenimiento, pues mientras que otros Frameworks se mantienen automáticamente, Flask traslada esa responsabilidad al usuario. Esto supone un mayor control, pero también genera más trabajo [Relan, 2019].

**Python Speech Recognition Library**

La librería Speech Recognition es una librería que se utiliza para realizar reconocimiento de voz en tiempo real o en archivos de audio pregrabados. Ésta tiene soporte de varias APIs [Buse 2012], en línea y fuera de línea. Dentro de éstas se encuentran: Google Speech Recognition, Wit.ai, Microsoft Azure Speech, Microsoft Bing Voice Recognition, IBM Speech to Text, Tensorflow, OpenAI whisper, Whisper API, entre otras. Esta librería está programada en el lenguaje Python y admite varios formatos de archivo de audio, como WAV, AIFF, y FLAC.

Otras características útiles de la librería Speech Recognition en Python son, el reconocimiento de palabras clave, el reconocimiento de múltiples idiomas y la integración con servicios de transcripción en línea. Esta librería es muy útil para aquellos que desean agregar capacidades de reconocimiento de voz en sus proyectos en Python. Así como también para aquellos que trabajan con datos de voz y necesitan transcribirlos a texto para su análisis. Actualmente, esta librería es utilizada por una amplia gama de personas, desde desarrolladores y científicos de datos, hasta investigadores, y entusiastas de la tecnología. Además, esta librería es especialmente popular por su facilidad de uso y la gran cantidad de documentación y ejemplos disponibles en línea. Además, muchas empresas y organizaciones también utilizan la librería Speech Recognition en sus productos y servicios, especialmente en el campo de la tecnología de la voz y el habla. Algunos ejemplos incluyen Amazon, Google, Microsoft, IBM, y Nuance Communications.

Dentro de sus ventajas se encuentra su facilidad de uso, el gran soporte que tiene y su documentación, así como la gran cantidad de ejemplos que existen en línea. Parte de esta librería se puede utilizar sin tener conexión a internet. Por otro lado,

dentro de sus desventajas se puede encontrar que, para una mejor integración y uso total del potencial de la librería en un proyecto o sistema, el dispositivo que se usa para obtener la voz debe tener conexión a internet. De no tener conexión a internet, los usos son más limitados.

## 3. Resultados

En este reporte técnico se realizó una revisión de 5 tipos de tecnologías disponibles: Neural Intents, PyTorch, Google Cloud Speech-to-Text, Flask, y Speech Recognition. En el análisis comparativo de la Tabla 1 se observan cuatro rubros en los cuales fueron evaluadas las tecnologías: dificultad de uso, documentación online, soporte al usuario disponible y, uso offline.

Tabla 1 Comparativa de las herramientas para reconocimiento de voz.

| | **Dificultad de uso** | **Documentación online** | **Soporte al usuario disponible** | **Uso offline** |
|---|---|---|---|---|
| Neural Intents | Avanzado | Si | No | Si |
| PyTorch | Avanzado | Si | No | Si |
| Google Cloud Speech-to-text | Medio | Si | No | No |
| Flask | Avanzado | Si | No | Si |
| Speech Recognition | Bajo | Si | Si | Si |

*Fuente: elaboración propia.*

En cuanto a dificultad de uso, se considera “bajo” cuando los conocimientos básicos de programación son suficientes para implementar la tecnología en un proyecto. “Medio” se considera cuando aparte de los conocimientos básicos de programación, se debe tener entendimiento de algoritmos de programación. “Avanzado” se considera cuando los algoritmos utilizados son más complejos y/o se requiere de “entrenamiento” de la Inteligencia Artificial. En lo que a documentación online se refiere es a si existen ejemplos de aplicaciones, y documentos a los que se pueda acceder para utilizar la tecnología de manera gratuita. El Soporte al usuario disponible se refiere a si existe algún “soporte técnico” de manera oficial gratuita al que se pueda acudir y, por último, se consideró si se pueden utilizar las herramientas sin conexión a internet.

Después de una revisión minuciosa, se encontró que la librería Speech Recognition fue la mejor opción para convertir la voz a texto utilizando Python. Dicha librería demostró una alta precisión de reconocimiento de voz en varios idiomas y en diferentes entornos acústicos. Además de ser fácil de usar y tener una gran cantidad de recursos disponibles en línea para su aprendizaje y desarrollo.

## 4. Discusión

Actualmente existe un auge de la inteligencia artificial, tecnologías como el reconocimiento de voz y la toma de decisiones son temas ya muy avanzados. Donde, uno de los mayores avances en esta área es, que los desarrolladores no deben comenzar desde cero. Es decir, ya existe un repositorio público en el que empresas grandes han colocado sus tecnologías para simplemente ser llamadas y utilizadas. Es aquí donde comienza este trabajo y, por lo tanto, es importante el comparar a los diferentes proveedores de este servicio. Por otro lado, para la comparación, dentro de este trabajo se decidió utilizar: la dificultad de uso, la documentación online, el soporte al usuario disponible y el uso offline. En este sentido, se encontró que la librería que ofrece mejores prestaciones es Speech Recognition. Dicha librería demostró una alta precisión de reconocimiento de voz en varios idiomas y en diferentes entornos acústicos. Además de ser fácil de usar y tener una gran cantidad de recursos disponibles en línea para su aprendizaje y desarrollo. No obstante, dentro de este trabajo se reconoce que éstas son tecnologías emergentes. Por lo tanto, tal vez en algunos años, el resultado de esta comparación pueda ser diferente.

## 5. Conclusiones

Dentro de este trabajo se analizaron cinco herramientas para reconocimiento de voz, donde la intención es utilizar una herramienta para reconocimiento de voz. Después de una revisión minuciosa, se encontró que la librería Speech Recognition fue la mejor opción para convertir la voz a texto utilizando Python. Dicha librería demostró una alta precisión de reconocimiento de voz en varios idiomas y en diferentes entornos acústicos. Además, de ser fácil de usar y tener una gran

cantidad de recursos disponibles en línea para su aprendizaje y desarrollo. Dicha librería demostró una alta precisión de reconocimiento de voz en varios idiomas y en diferentes entornos acústicos. Además de ser fácil de usar y tener una gran cantidad de recursos disponibles en línea para su aprendizaje y desarrollo. No obstante, dentro de este trabajo se reconoce que éstas son tecnologías emergentes. Por lo tanto, tal vez en algunos años, el resultado de esta comparación pueda ser diferente.

## 6. Bibliografía y Referencias

[1] Amos, D. (2016). The ultimate guide to speech recognition with python. Real Python.

[2] Buse, R. P., & Weimer, W. (2012, June). Synthesizing API usage examples. In 2012 34th International Conference on Software Engineering (ICSE) (pp. 782-792). IEEE.

[3] Cruz Montealegre, M. C. (2021). Identificación de idioma y respuesta en tiempo real usando técnicas de Deep Learning con espectrogramas y reconocimiento de voz.

[4] González, A. (2020). Reconocimiento de voz: que es, cómo funciona y programas que existen. Ayuda Ley Protección Datos. https://ayudaleyprotecciondatos.es/2020/05/19/reconocimiento-voz/

[5] Imambi, S., Prakash, K. B., & Kanagachidambaresan, G. R. (2021). PyTorch. Programming with TensorFlow: Solution for Edge Computing Applications, 87-104.

[6] Karumuri, H., Kimche, L., Toker, O., & Doryab, A. (2022, April). Context-Aware Recommendation Via Interactive Conversational Agents: A Case in Business Analytics. In 2022 Systems and Information Engineering Design Symposium (SIEDS) (pp. 375-380). IEEE.

[7] Ketkar, N., Moolayil, J., Ketkar, N., & Moolayil, J. (2021). Introduction to pytorch. Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch, 27-91.

[8] Navarro Sánchez, F. (2020). Reproductor de música con reconocimiento de voz (Doctoral dissertation, Universitat Politècnica de València).

[9] Relan, K., & Relan, K. (2019). Beginning with flask. Building REST APIs with Flask: Create Python Web Services with MySQL, 1-26.

[10] Ríos, J. R. M., Mora, N. M. L., Ordóñez, M. P. Z., & Sojos, E. L. L. (2016). Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python. Archivo de la revista Latinoamericana de Ingeniería de Software, 4(4), 201-207.

[11] Rithika, H., & Santhoshi, B. N. (2016, December). Image text to speech conversion in the desired language by translating with Raspberry Pi. In 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (pp. 1-4). IEEE.

[12] Sabharwal, N., Agrawal, A., Sabharwal, N., & Agrawal, A. (2020). Introduction to Google dialogflow. Cognitive virtual assistants using google dialogflow: develop complex cognitive bots using the google dialogflow platform, 13-54.

[13] Salvi, S., Geetha, V., & Kamath, S. S. (2019, October). Jamura: a conversational smart home assistant built on Telegram and Google Dialogflow. In TENCON 2019-2019 IEEE Region 10 Conference (TENCON) (pp. 1564-1571). IEEE.

[14] Torres, J. (2020). Python deep learning: Introducción práctica con Keras y TensorFlow 2. Alpha Editorial.

[15] Vera Popoca, R., Osnaya Baltierra, S., & Mendoza Frías, R. (2021). Prototipo para la aplicación de exámenes a través del reconocimiento de voz. Revista RedCA, 3(9), 3-17.

[16] Villa Román, J., & Villa Román, V. F. (2017). Interfaz voz-texto para la consulta de base de datos para la traducción al lenguaje de señas.