

**Detección Automática de Actividades De La Vida Diaria En Primera Persona Por  
Medio De Visión por Computador**

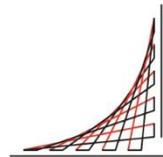
**Elkin Estiven González Cuellar  
Rolffer Alexander Ramírez Serrano**

**Trabajo Dirigido**

**Tutor  
Oscar Julián Perdomo Charry, PhD**



**Universidad del  
Rosario**



**ESCUELA  
COLOMBIANA  
DE INGENIERÍA  
JULIO GARAVITO**

**UNIVERSIDAD DEL ROSARIO  
ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO  
PROGRAMA DE INGENIERÍA BIOMÉDICA  
BOGOTÁ D.C  
2022**

## RESUMEN

El deterioro cognitivo se entiende como un amplio espectro de condiciones en el cual hay un decaimiento en las funciones cognitivas, generalmente se da de forma progresiva y es causado por la edad o alguna enfermedad grave, se caracteriza por alteraciones de la memoria, la razón y trastornos en la conducta.

En los últimos años el descenso de la mortalidad, el aumento de la esperanza media de vida y el descenso de la fecundidad, han generado que la población de personas con más de 65 años aumente año tras año. la OMS (Organización Mundial de la Salud) estima que para 2025 el 70% de la población mundial será mayor a 60 años. Esto se relaciona directamente con las personas que presenten deterioro cognitivo ya que la edad es una de las principales causas. [1]

La visión artificial también conocida como visión por computadora o visión técnica es aquello que incluye todos los métodos y técnicas utilizados para adquirir, procesar, analizar y comprender las imágenes del mundo real para producir información numérica o simbólica que pueda ser procesada por un ordenador. La visión artificial tiene múltiples aplicaciones tanto en medicina como en la industria, algunos ejemplos de estas aplicaciones se encuentran en la detección de tumores, la participación asistida de robots de alta precisión, seguridad industrial y control de calidad en la industria.

En años recientes el uso de esta tecnología se ha incrementado, en gran medida debido a que optimiza procesos que normalmente generan más gastos económicos, ambientales y sociales, un ejemplo de esto se encuentra en el cuidado dentro del hogar de personas con deterioro cognitivo, estas requieren de atención y monitoreo constante con el fin de dar asistencia ante cualquier eventualidad, Generando una gran carga para los profesionales encargados del cuidado o rehabilitación de múltiples pacientes. la visión artificial se puede convertir en una gran herramienta para los profesionales al cuidado de las personas con déficit cognitivo, ya que esta puede monitorear al paciente y dar alarma al profesional ante cualquier eventualidad en una actividad propuesta por ellos mismos.

Este documento se enfocará en brindar una solución a los posibles problemas que tienen en su vida cotidiana las personas con deterioro cognitivo, entrenando nuestro propio modelo de detección de objetos mediante PyTorch y complementándolo con un asistente de voz creado en DialogFlow.

## Contenido

1. INTRODUCCIÓN .....	6
2. MARCO TEÓRICO.....	7
2.1. Terapias para el deterioro cognitivo .....	7
2.1.1. Compensación.....	7
2.1.2. Capacitación para tareas específicas .....	7
2.1.3. Enfoques para rehabilitación.....	8
2.1.4. Tecnologías empleadas .....	8
2.2. Inteligencia Artificial .....	9
2.2.1. Aprendizaje de Máquina.....	9
2.2.2. Aprendizaje Profundo.....	10
2.3. Asistente de Voz .....	10
2.3.1. Experiencia Conversacional.....	10
2.3.2. Procesamiento del Lenguaje Natural.....	11
2.3.3. Plataformas de desarrollo .....	13
2.3.4. Plataformas de despliegue.....	16
2.3.5. Voice Kit.....	17
2.4. Visión por Computador .....	17
2.4.1. Aplicaciones de la visión computarizada .....	18
2.4.2. Redes Convolucionales.....	19
2.4.3. Frameworks .....	19
2.4.4. Detección de Objetos .....	20
2.4.5. Algoritmos de Detección .....	21
3. OBJETIVOS .....	24
3.1. General .....	24
3.2. Específicos.....	24
4. METODOLOGÍA.....	25
4.1. Nvidia Jetson Nano.....	25
4.1.1. Nvidia TensorRT.....	26
4.2. Repositorio GitHub.....	26
4.3. Base de datos .....	27
4.4. Transfer Learning with PyTorch .....	27
4.5. Entrenamiento de Nuestro Propio Modelo .....	28
4.5.1. Descargar los datos.....	28
4.5.2. Entrenamiento del modelo SSD-MobileNet.....	28

4.5.3.	Conversión del modelo a ONNX .....	29
4.5.4.	Poner en Marcha Nuestro Modelo de Detección de Objetos .....	29
4.6.	Asistente de Voz .....	29
4.6.1.	Diseño de la experiencia conversacional.....	30
4.6.2.	Construcción del Diagrama Principal .....	34
4.6.3.	Implementación en Dialogflow .....	35
4.6.4.	Voice kit .....	38
4.6.5.	Aplicación de prueba.....	39
5.	RESULTADOS .....	40
5.1.	Modelo de Reconocimiento de Objetos .....	40
5.2.	Asistente de Voz .....	41
5.2.1.	Diagrama final.....	43
6.	DISCUSIÓN.....	46
7.	RECOMENDACIONES Y TRABAJOS FUTUROS .....	48
8.	CONCLUSIONES.....	49
	Bibliografía .....	49
9.	ANEXOS.....	53
9.1.	GOOGLE COLAB .....	53
9.2.	GITHUB JETSON INFERENCE NVIDIA.....	54
9.3.	Open Images dataset v6.....	54
9.4.	VIDEOS DE PRUEBA .....	55
9.5.	Repositorio asistente de voz .....	56

## Índice de Tablas

Tabla 1. Lista de verificación para determinar el impacto de la experiencia conversacional. ....	31
Tabla 2 Entidades adicionales utilizadas en Dialogflow. ....	35
Tabla 3 Entidades que usa el agente en la implementación del sistema. ....	37
Tabla 4 . Realimentación de las pruebas aplicadas y observaciones. ....	43

## Índice de Figuras

Figura 1. Desglose de actividades de la vida diaria .....	8
Figura 2. Esquema de un agente con sus componentes Dialogflow. ....	14
Figura 3 .Funcionamiento de clasificación de intenciones. ....	14
Figura 4. Flujo básico para la coincidencia de intents y respuesta al usuario final usando contextos. ....	15
Figura 5. Red convolucional .....	19
Figura 6 . Arquitectura de la Detección de Objetos .....	20
Figura 7. R6-CNN .....	21
Figura 8. Mobilnet-ssd .....	22
Figura 9. Tutorial instalación Jetpack.....	25
Figura 10. TensorRT .....	26
Figura 11. Transfer Learning .....	27
Figura 12. Descarga de la base de datos carpeta 1,2 y 3.....	28
Figura 13. Argumentos10 de Entrenamiento .....	29
Figura 14. Comando para ejecutar el código .....	29
Figura 15. Comando cámara web .....	29
Figura 16. Secuencia de dialogo de prueba .....	33
Figura 17 . Diagrama principal inicial del flujo de conversacional planteado.....	34
Figura 18. Lógica inicial de los pasos indicados para completar una tarea.....	34
Figura 19. Ventana de Conexión usando Secure Shell.....	39
Figura 20. Prueba detección de objetos .....	40
Figura 21. Prueba detección de objetos .....	41
Figura 22. Diagrama de flujo que establecerá la lógica final que implementará el agente parte a. ....	44
Figura 23. Diagrama de flujo que establecerá la lógica final que implementará el agente parte b. ....	44
Figura 24. Prueba de la tarea “preparar café” diseñada en Dialogflow utilizando el Voice kit parte a.....	45
Figura 25. Prueba de la tarea “preparar café” diseñada en Dialogflow utilizando el Voice kit parte b.....	45
Figura 26. Prueba de la tarea “preparar café” diseñada en Dialogflow utilizando el Voice kit parte c.....	45
Figura 27. Prueba de distancia y orientación .....	46
Figura 28. Clasificación errónea .....	46

## 1. INTRODUCCIÓN

La cognición es la capacidad del ser humano para procesar la información a partir de la percepción producida por nuestros sentidos y la experiencia de conocimiento adquirido anteriormente. Los déficits cognitivos, como la dificultad para concentrarse, recordar, planificar, organizar o priorizar, limitan las actividades cotidianas en la vida de las personas. [2]

Los profesionales utilizan los principios de la rehabilitación cognitiva para lograr una cognición funcional que se refiere a las habilidades de pensamiento y procesamiento que utilizan las personas para realizar las actividades cotidianas, pero también se presenta como la nueva definición de enfoques adaptativos o funcionales, este enfoque enseña y capacita a las personas para que logren realizar las tareas y funciones que necesitan y desean realizar “La estrategia es adaptar el entorno a las capacidades de una persona”. [2] Los modelos de rehabilitación cognitiva abarcan desde tratamientos basados en neuroanatomía y enfoques restauradores o reparadores, hasta modelos compensatorios cognitivos, modelos funcionales o ambientales y modelos de habilidades o capacitación.

El enfoque de rehabilitación cognitiva elegido depende de varios factores, incluyendo la edad de la persona, el apoyo familiar, la autoconciencia, gravedad y cronicidad de los problemas de rendimiento cognitivo. La rehabilitación cognitiva incluye tanto el individuo e intervenciones grupales para apoyar mejor las necesidades y el progreso de cada cliente. La rehabilitación cognitiva permite individuos con varios niveles de severidad para obtener una mayor de los desafíos de vivir con problemas cognitivos. [3]

Actualmente existen múltiples herramientas tecnológicas que hacen uso de la inteligencia artificial, para favorecer la prevención, asistencia, detección y rehabilitación de pacientes con deterioro cognitivo. Es necesario que el uso de estas tecnologías sea llevado a cabo por personal especializado que supervise su uso para contribuir al cuidado del usuario de manera efectiva. En [4] se propone un sistema que envía notificaciones a pacientes y cuidadores para optimizar el tratamiento. Algunos prototipos que monitorean al paciente y que contienen funciones de cuidado a partir de ambientes inteligentes fueron implementados y son expuestos en [5], consisten en dispositivos que ofrecen opciones básicas que permiten obtener ayuda y generar recordatorios personalizados.

La detección de objetos es una rama de la visión artificial que se centra en la detección y clasificación de objetos semánticos en imágenes o videos, esta tecnología acompañada de un asistente de voz interactivo se podría convertir en una valiosa herramienta para ayudar a los profesionales en la ejecución de algunos modelos compensatorios ayudando a las personas que tengan problemas con su orientación, atención, memoria o problemas de organización y planificación.

## **2. MARCO TEÓRICO**

### **2.1. Terapias para el deterioro cognitivo**

Los profesionales de terapia ocupacional son capaces de guiar a personas con déficits cognitivos para que adquieran habilidades necesarias en su vida diaria, para ello se enfocan en la cognición funcional que incluye abordar la relación entre las personas, sus ocupaciones diarias y el contexto. [6] [7]

La rehabilitación cognitiva está dedicada a restaurar procesos mentales como la memoria, la atención, la comprensión, la resolución de problemas, el razonamiento, la autoconciencia y la percepción, utilizando técnicas diseñadas y enseñando estrategias compensatorias a través de intervenciones cognitivas que se enfocan en el funcionamiento cognitivo en lugar del funcionamiento conductual, La rehabilitación cognitiva nos permite remediar compensar o prevenir la pérdida del funcionamiento cognitivo y promover el bienestar de las personas. [2]

Las intervenciones cognitivas se administran en todo tipo de poblaciones; sin embargo, para que la intervención tenga éxito, se deben considerar las características grupales o individuales de las personas con la lesión o enfermedad. Las personas con alteraciones en funciones cognitivas deben ser monitoreadas constantemente y para esto el uso de la inteligencia artificial muestran buen rendimiento en aplicaciones de detección y seguimiento de personas en tiempo real. [7] [8]

#### **2.1.1. Compensación**

La compensación está orientada a métodos que suplan o solucionen dificultades que se presenten al realizar una tarea, función o proceso permitiendo completarlas con éxito y adaptarlas al paciente.

#### **2.1.2. Capacitación para tareas específicas**

El entrenamiento en tareas específicas como es el caso de las actividades de la vida diaria por sus AVDs. la rehabilitación cognitiva tiene como objetivo mejorar el desempeño de las tareas funcionales a través de la práctica dirigida a objetivos y la repetición. Al analizar las actividades de la vida diaria es fundamental considerar las tareas que las componen y los desafíos funcionales de cada tarea. En la siguiente figura se observa la descomposición de algunas de estas actividades.



Figura 1. Desglose de actividades de la vida diaria [7]

### 2.1.3. Enfoques para rehabilitación

Existen 3 enfoques principales utilizados actualmente en la rehabilitación cognitiva: Restaurativa, funcional e integrada. El enfoque restaurativo está basado en el entrenamiento jerárquico en el cual se realizan tareas con dificultad creciente y se centra en los procesos cognitivos deteriorados. [9]

El enfoque funcional busca reducir las limitaciones de la actividad y las restricciones de participación en lugar de remediar o restaurar las habilidades deterioradas, se puede subdividir en tres enfoques de intervención: Adaptación de la actividad o contexto [6]; entrenamiento en tareas específicas y compensación. El enfoque funcional aprovecha las fortalezas de una persona para mejorar la independencia en las AVDs. [8] [8]

Por último, Enfoque integrado es la combinación de los elementos del enfoque de restauración y funcional.

### 2.1.4. Tecnologías empleadas

En un estudio realizado por Boman, Tham, Granqvist, Bartfai y Hemmingsson (2007) en pacientes con lesión cerebral adquirida se utilizaron ayudas de memoria electrónica, intervenciones constantes de los terapeutas y repeticiones frecuentes. los pacientes recibieron tantas indicaciones como fuera necesario para poder utilizar las ayudas electrónicas de memoria dentro de un apartamento equipado con estas ayudas tecnológicas. Este estudio concluyó que una combinación de enfoques compensatorios y específicos de las tareas son eficaces para el aprendizaje de las AVDs. [10]

## **2.2. Inteligencia Artificial**

Existe un gran número de definiciones para la inteligencia artificial (IA), en términos simples esta descrita como la capacidad que tienen los sistemas o máquinas para realizar tareas propias de la inteligencia humana, como la capacidad de aprender, mejorar iterativamente a partir de la información que recopilan, y resolver problemas en base a ese conocimiento [11].

Dentro del campo de la inteligencia artificial se encuentran diferentes subcategorías que responden a diferentes comportamientos inteligentes, por ejemplo, el procesamiento del lenguaje natural está dedicado a la capacidad de entender el lenguaje de las personas. favoreciendo aplicaciones como la conversión de voz a texto o de texto a voz, también se encuentra el campo de la visión por computadora, que incluye métodos para adquirir, procesar, analizar y comprender las imágenes del mundo real. con el fin de producir información numérica o simbólica para que puedan ser tratados por un ordenador. En la práctica los sistemas que emplean IA están enfocados en una tarea específica y han sido aplicados en diferentes sectores como: salud, transporte, construcción, vigilancia y seguridad. [12]

Algunos ejemplos son:

- Chatbots/voicebots, utilizan IA para dar respuestas más eficientes.
- Motores de recomendación, proporcionan recomendaciones automatizadas para programas de TV según los hábitos de visualización de los usuarios.
- Automóviles autónomos.

Dentro de la inteligencia artificial encontramos diferentes ramas:

### **2.2.1. Aprendizaje de Máquina**

También llamado aprendizaje automático, es la rama de IA que estudia cómo dotar a las máquinas de capacidad de aprendizaje, entendido éste como la generalización del conocimiento a partir de un conjunto de experiencias, empleando un conjunto de técnicas mediante las que un algoritmo es capaz de modificar su comportamiento, basándose en datos como resultados de acciones pasadas o en acciones que se indiquen como correctas o incorrectas. [13]

En esta rama encontramos principalmente tres tipos de aprendizaje:

- Aprendizaje supervisado: se refiere al aprendizaje que se basa en descubrir la relación existente entre unas variables de entrada y unas variables de salida, es decir, enseñar al algoritmo cual es el resultado que se espera obtener para un determinado valor [13].
- Aprendizaje no supervisado: es el paradigma que consigue producir conocimiento, utilizando únicamente los datos de entrada sin mostrarle al sistema el resultado que se desea obtener, un ejemplo de un método para esto es la clusterización la cual identifica similitudes entre objetos, los cuales agrupa según características en común, por tanto, les diferencian de los otros grupos de objetos [13].
- De refuerzo: este tipo de aprendizaje está basado en recompensas, los algoritmos de este tipo no son condicionados a que acción deben ejecutar, sino que a partir de

las experiencias se conoce cuales acciones conllevan un resultado satisfactorio, comúnmente es utilizado en sistemas de navegación de robots, automóviles autónomos y drones [13].

### **2.2.2. Aprendizaje Profundo**

Es un subconjunto del Machine Learning basado en estructuras lógicas que se asemejan a la organización del sistema nervioso, teniendo capas de unidades de proceso (neuronas artificiales) conectadas entre sí, que se especializan en la detección de determinadas características ocultas en los datos, formando una red que está especializada en realizar una tarea específica [14]. Este tipo de algoritmo ha superado a los algoritmos tradicionales en tareas de clasificación de imágenes, texto y voz.

De las diferentes áreas de la inteligencia artificial se van a abordar las más relevantes para este proyecto, las cuales son el procesamiento del lenguaje natural PLN, siendo este el punto de partida para el entendimiento y posterior desarrollo del asistente de voz, las redes neuronales artificiales y Deep learning, enfocados en la clasificación de objetos para la detección de las AVDs por medio de visión por computadora.

## **2.3. Asistente de Voz**

### **2.3.1. Experiencia Conversacional**

Una experiencia conversacional es una estrategia diseñada para que los usuarios mediante un canal de comunicación como los chatbots y voicebots puedan tener una conversación de forma fluida, eficaz intuitiva y personalizada [15]. Los asistentes de voz deben ser particularmente buenos en entablar conversaciones que se sientan más profundas y naturales con sus usuarios, esto se logra debido a los avances significativos que se han llevado a cabo en la inteligencia artificial, particularmente en el área del PLN, proporcionando así herramientas para generar acciones que se sientan más personales con los usuarios [16]. El diseño conversacional involucra un gran número de disciplinas, entre las cuales destacan las interfaces de voz, el diseño interactivo, la escritura de experiencia de usuario, diseño visual y diseño de animación, la razón de esto es básicamente porque las experiencias conversacionales están disponibles en un gran número de dispositivos y algunos de esos dispositivos cuentan con pantallas. En el diseño se debe tener en cuenta las necesidades del usuario para establecer los requerimientos de la herramienta y definir los flujos necesarios para que el usuario tenga una experiencia satisfactoria [17].

### 2.3.2. Procesamiento del Lenguaje Natural

Es importante saber que se conoce como Procesamiento de Lenguaje Natural (PLN) ya que es parte fundamental de este proyecto, el asistente debe ser capaz de entender el contexto de las preguntas que se le están realizando y dar respuestas acordes a las necesidades del usuario. El PLN es un área que combina las ciencias de la computación, inteligencia artificial y lenguajes de programación, esto con el objetivo de entender la naturaleza de las interacciones entre humanos y máquinas, por medio del lenguaje que usan las personas para comunicarse [18]. El estudio formal del PLN empezó en el año 1950 cuando se publicó el artículo *Computing machinery and intelligence* de la autoría Alan Turing, en el cual se propone el “Test de Turing” para evaluar la capacidad de una máquina para exhibir un comportamiento inteligente similar al de un ser humano [19], no fue hasta el año 2014 que el bot conversacional Eugene Goostman logra ser la primera inteligencia artificial aprobada al someterse a dicho test [20]. El lenguaje humano es bastante ambiguo, razón por la cual diseñar un software preciso para determinar el significado de datos de texto y voz no es una tarea fácil [15].

El PLN comprende dos pasos principales, el preprocesamiento de datos y el desarrollo de algoritmos.

#### **Preprocesamiento.**

El objetivo es preparar y limpiar los datos de texto, para así destacar las características del texto con las que el algoritmo puede trabajar, existen varios métodos para realizar esta tarea como lo son:

- Tokenización: El cual separar en palabras toda la cadena de texto en tokens o unidades mínimas lingüísticas [19].
- Lematización: Convertir cada una de las palabras o tokens de una frase o una cadena de texto a su raíz fundamental, es decir, dada una forma flexionada (plural, en femenino, conjugada, etc.) debe convertirse a la forma que se acepta como representante de todas las formas flexionadas de una misma palabra [19].
- Etiquetado del discurso: Este clasifica las palabras en función de la parte del discurso como, sustantivos, verbos y adjetivos [19].

Realizado el preprocesamiento, el siguiente paso es el uso de algoritmos de PLN para procesarlos, son dos los comúnmente utilizados:

**Algoritmos basados en reglas:** Es el primer enfoque utilizado para el desarrollo de PLN, y está basado en un conjunto de reglas lingüísticas complejas cuidadosamente diseñadas, teniendo en cuenta las distintas combinaciones, matices, ambigüedades, etc. que tiene el lenguaje, a largo plazo es insostenible ya que además de ser una labor compleja requiere ser actualizada a medida que el lenguaje evoluciona [11].

**Sistema basado en Machine learning:** Utilizan métodos estadísticos y aprenden a realizar tareas, basándose en los datos de entrenamiento que reciben, perfeccionando sus propias reglas a través del procesamiento y el aprendizaje repetidos. [13]

Aplicaciones que hacen uso de PLN:

- Detección de spam.
- Traducción automática.
- Asistentes virtuales y chatbots
- Análisis del sentimiento en las redes sociales.
- Análisis y síntesis de voz.

Ahora que se han dado todos los conceptos base de inteligencia artificial y sus áreas, como el procesamiento del lenguaje natural, el cual es fundamental para el diseño y construcción de la parte del proyecto encargada de la asistencia por instrucciones de voz, se procede a indagar que es un voicebot y cuáles son las herramientas disponibles para desarrollarlo.

### **2.3.2.1. Asistentes Virtuales (Voicebots)**

Un voicebot es un software destinado a entablar una conversación con personas mediante voz, este utiliza algoritmos de IA dedicados al procesamiento del lenguaje natural dándole la capacidad de responder automáticamente y de forma autónoma utilizando el lenguaje humano [21]. La voz es el medio más rápido de comunicación humana y esto destaca a los voicebots por encima de los “chatbots” o asistentes digitales, que se comunican a través de instrucciones de texto y requieren que la persona sea capaz de manipular la tecnología, lo cual en caso de personas mayores o con déficits cognitivos puede ser poco eficiente [22]. Actualmente esta tecnología está en crecimiento debido a que cada día aparecen más campos de aplicación, gracias a las diferentes funcionalidades que se pueden ofrecer, son comúnmente utilizados en canales de atención al cliente en empresas de distintos sectores.

### **2.3.2.2. Voicebots en Salud**

En el sector salud existen diversos programas destinados a optimizar y agilizar procesos, como reservas de citas médicas, realizar consultas acerca de medicamentos e indicaciones de uso de estos, procesar datos clínicos, gestionar citas para control e incluso entretenimiento para largos periodos de tiempo en el hospital, en los últimos años se ha tratado de implementar para el diagnóstico de enfermedades [22]. La Asociación Médica Americana evalúa que al menos el 75% de las citas presenciales pueden sustituirse por atención médica a distancia lo cual ha tomado mayor relevancia por la reciente pandemia causada por el COVID-19, incluyendo a los voicebots y chatbots como un tipo de filtro para establecer el grado de urgencia o incluso aconsejando la especialidad más apropiada, optimizando tiempos de espera y mejorando la atención.

Algunos ejemplos de estos son:

- Alerta de Salud de la Organización Mundial de la Salud: Es un chatbot que utiliza la OMS para brindar información actualizada vía WhatsApp sobre el coronavirus.
- ACCU: La confederación de afectados por la enfermedad de Crohn y Colitis Ulcerosa en España, ha desarrollado un voicebot dirigido a pacientes con enfermedad inflamatoria intestinal.
- Elipse Salud: Es un bot con múltiples mecanismos de automatización en servicios sanitarios, el cual ayuda a las personas a prevenir posibles problemas de salud. Esta aplicación hace uso de voicebots, chatbots y callbots que brindan soluciones

de contabilidad, atención al cliente e incluso brindan servicio de TRIAGE a través de un WhatsApp Bot.

### **2.3.3. Plataformas de desarrollo**

En la actualidad herramientas como chatbots y voicebots están fuertemente demandadas por las grandes industrias, razón por la cual se han lanzado plataformas destinadas a facilitar el desarrollo de estas herramientas, en esta sección se abordarán algunas de las más relevantes en la actualidad y se profundizará en Dialogflow, en la cual se lleva a cabo el desarrollo de este proyecto.

#### **2.3.3.1. Microsoft Bot Framework**

Es un kit de desarrollo de software (SDK) que brinda herramientas para construir, probar y desplegar bots inteligentes, este incluye plantillas y servicios de IA destinados al entendimiento del lenguaje natural, además cuenta con un repositorio con ejemplos de desarrollos previos [23].

Para el desarrollo es posible utilizar lenguajes de programación como C#, Java Script, Python y Java, además brinda herramientas como Bot Framework Composer, un entorno de desarrollo integrado (IDE) de código abierto, en el cual es posible crear diálogos y modelos de comprensión del lenguaje, facilitando la construcción en casos complejos [23].

#### **2.3.3.2. DialogFlow**

Es una plataforma destinada a la comprensión del lenguaje natural que permite diseñar interfaces conversacionales y brinda múltiples posibilidades para integrarlas en aplicaciones como Messenger, Google Assistant, Voximplant, Twilio, Telegram, etc. Esta herramienta permite procesar entradas y dar respuestas en voz y texto, lo cual permite convertir fácilmente un chatbot en voicebot.

Esta plataforma pertenece a Google el cual proporciona múltiples herramientas, entre las cuales se encuentran Google Cloud y Firebase, que usándolas en conjunto con DialogFlow permiten el desarrollo de proyectos de más alto nivel con una complejidad aceptable, además proporciona técnicas de Machine Learning desarrolladas por Google en las que encontramos Google's text-to-speech y speech-to-text las cuales permiten la conversión de voz a texto y viceversa, todo esto proporciona una gran cantidad de datos que esta plataforma utiliza para su constante entrenamiento.

Dialogflow hace uso de algunos conceptos básicos generalmente utilizados para el desarrollo de estas aplicaciones, a continuación, se explican los más relevantes.

**Agente:** Los agentes son módulos de comprensión del lenguaje natural que se encargan de manejar las conversaciones. Dialogflow traduce el texto o el audio del usuario final durante una conversación a datos estructurados que puedan ser utilizado por los servicios usados en la aplicación. [24]

# Agentes

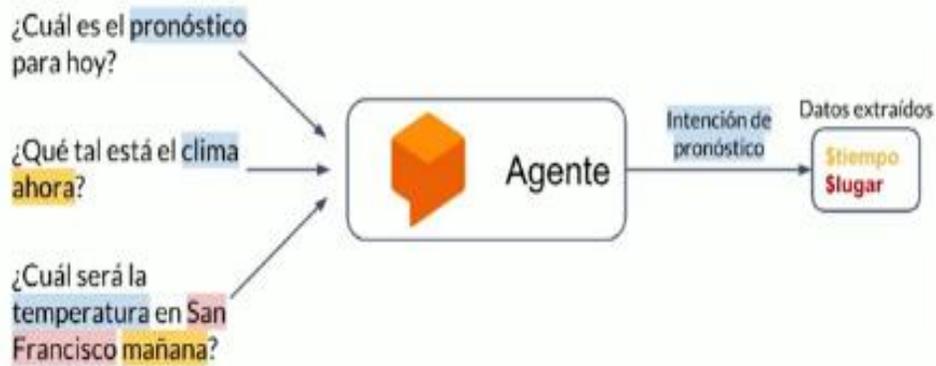


Figura 2. Esquema de un agente con sus componentes Dialogflow. [25]

**Intents:** Son los encargados de identificar la intención del usuario en el transcurso de la conversación, múltiples agentes son los encargados de manejar el flujo conversacional que se defina. Dialogflow hace coincidir la expresión del usuario con el Intent más apropiado. [24]

## ¿Clasificación de Intenciones?



Figura 3. Funcionamiento de clasificación de intenciones. [22]

Un Intent está formado por los siguientes elementos:

- **Frases de entrenamiento:** Son una serie de posibles respuestas dadas por el usuario que son utilizadas para coincidir con los intents disponibles, no es necesario proporcionar muchas respuestas ya que Dialogflow hace uso de aprendizaje automático para incrementar esta lista detrás de escena. [24]
- **Parámetros:** Estas son las palabras extraídas de las frases, cada parámetro este asociado a una entidad que es encargada de clasificar el tipo al que pertenece la palabra, lo cual nos ayuda a clasificar los datos extraídos para poder establecer la lógica en la conversación que generara las respuestas. [24]
- **Respuestas:** Es necesario definir los tipos de respuesta que daremos al usuario puede utilizar texto, voz e incluso imágenes [24].

**Entidades:** Cada Intent tiene distintos parámetros encargados de asociar las palabras, estas son las entidades las cuales están conformada por un grupo de palabras que pertenecen a un mismo grupo. El sistema tiene entidades por defecto que el sistema usa para encontrar fechas, horas, colores, tamaños, etc. Es posible crear entidades propias para clasificar las frases como parámetros de un mismo tipo. [24]

**Contextos:** Los contextos de Dialogflow son similares al contexto del lenguaje natural. Si una persona dice "es de color naranja", necesita contexto para saber qué es de ese color [24]. Los contextos nos ayudan a dirigir el flujo de la conversación, estos son incluidos en los Intents como contextos de entrada o salida, los que nos ayuda a conectar el flujo de la conversación. [24]

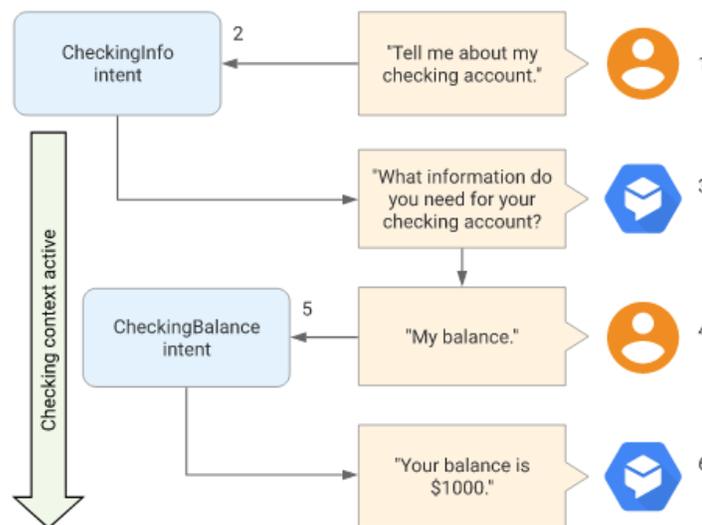


Figura 4. Flujo básico para la coincidencia de intents y respuesta al usuario final usando contextos. [24]

### 2.3.3.3. Amazon Lex

Es un servicio de Amazon Web Services destinado a la creación de interfaces de conversación para dispositivos y aplicaciones que usan voz y texto, este destaca por utilizar

el mismo motor que emplea el famoso asistente de Amazon Alexa y que ahora está disponible para cualquier desarrollador. Esta herramienta no requiere de mucha experiencia o conocimientos técnicos ya que basta con establecer el flujo de conversación básico y Amazon Lex se encarga de ajustar dinámicamente las respuestas en la conversación, su servicio puede ser integrado en dispositivos móviles y aplicaciones web, al igual que Google proporciona otros servicios como Amazon Cognito, Amazon CloudWatch y Amazon DynamoDB que son fácilmente integrables para la construcción de la aplicación. [26]

#### **2.3.4. Plataformas de despliegue**

La computación en la nube es la posibilidad de almacenar datos de forma remota, específicamente en la nube, esto facilita el intercambio de información entre distintos sectores [27]. En la actualidad existen una gran cantidad de plataformas que ofrecen servicios de computación en la nube, por lo cual no es necesario disponer de servidores propios para el despliegue de las aplicaciones, esto ahorra tiempo de trabajo y proporciona una mejora en el procesamiento, a continuación, se mencionarán algunas de las plataformas más relevantes que ofrecen este servicio.

##### **2.3.4.1. Google Cloud**

Google Cloud Platform (GCP) es un proveedor de recursos de computación en la nube, que cuenta con una infraestructura completa con diversas características y herramientas que facilitan el desarrollo, implementación y operación de aplicaciones web. GCP proporciona diversos servicios, entre los cuales encontramos a la infraestructura como servicio (IaaS) que es básicamente la contratación o alquiler de hardware como, la capacidad de proceso (procesadores), memoria RAM y el espacio de almacenamiento, de igual manera ofrece Plataforma como servicio (PaaS) que permite lanzar aplicaciones como bases de datos, middleware, herramientas de desarrollo y por último se encuentra el Software como servicio (SaaS) encargado de almacenar todo el software y sus datos en la nube permitiendo que puedan ser accedidos desde cualquier lugar sin necesidad de instalarlo en un entorno local, evitándonos problemas que puedan surgir por recursos del hardware, como puede ser, sistemas operativos, recursos de almacenamiento e incluso aplicaciones. [28]

##### **2.3.4.2. Amazon Web Services (AWS)**

Esta plataforma de computación en la nube al igual que GCP nos brinda capacidad de almacenamiento, recursos específicos del hardware, bases de datos entre otras muchas funcionalidades propias de estas plataformas. AWS está en funcionamiento desde el año 2006, evitando a empresas emergentes la necesidad de realizar una inversión en infraestructura de ordenadores potentes y costosos. [29]

Uno de sus servicios más relevantes es Elastic Computing Cloud (EC2) el cual está orientado a IaaS, este permite el alquiler de computadores virtuales en los cuales pueden ejecutar sus propias aplicaciones.

##### **2.3.4.3. Microsoft Azure**

Es el servicio de computación en la nube lanzado por Microsoft en el año 2010, proporciona servicios IaaS, SaaS y PaaS además es compatible con distintos lenguajes como PHP,

C++, Ruby, Java y cuenta con un ecosistema para desarrolladoras basado en el framework .NET. Esta plataforma hace uso de un sistema operativo que utiliza un clúster como sistema de procesamiento paralelo, localizado en los servidores de Microsoft y se encarga de la gestión de los recursos de las aplicaciones que utilizan Microsoft Azure. [30]

Una característica relevante es que ofrece copias de seguridad automatizadas en un servicio de almacenamiento, además el usuario no debe preocuparse por las tareas de configuración de estos servicios, gracias a su enfoque en la gestión y recuperación de datos.

### **2.3.5. Voice Kit**

Este kit ofrecido por Google permite la construcción de un prototipo propio para procesamiento de lenguaje natural y facilita su conexión con el Asistente de Google o al servicio Cloud Speech to Text. El voice kit cuenta con todos los materiales en los cuales se encuentra una Raspberry Pi Zero, que hace de ordenador para el procesamiento de datos de entrada, que junto con el Voice Bonnet la cual es una tarjeta que facilita la captura y reproducción de audio se conectan para formar un procesador de lenguaje natural, todo en una pequeña caja que será el asistente de voz, el cual responderá las preguntas y ejecutará las ordenes incluidas en el modelo. [31]

El voice kit cuenta con ejemplos de código que permiten utilizar el servicio de Google Cloud Speech to Text, el cual se encarga de convertir la voz en texto, para utilizar dicho servicio es necesario conectar el kit a internet por medio de wi-fi y dispone de dos opciones para esto, la primera es por medio de un celular Android, descargando la app AIY Projects disponible en Google Play Store y la segunda es por medio de un computador Windows, Mac o Linux, un monitor adicional, mouse y teclado, todos estos se conectan al kit y para ello es necesario adquirir algunos adaptadores adicionales para los puertos de la Raspberry pi. [31]

Google proporciona toda la documentación acerca de las APIs desarrolladas para facilitar la construcción de proyectos llevados a cabo con el voice kit, todas las APIs están integradas en un paquete de Python llamado aiy, este viene preinstalado en el archivo .img.xz que se encuentra en el repositorio de GitHub google/aiyprojects-raspbian, el cual es cargado en la SD que se coloca en la Raspberry pi.

Las siguientes APIs son de especial interés para el desarrollo de este proyecto:

Aiy. assistant: Facilita la integración con la API de Google Assistant.

Aiy. cloudspeech: Encargada de la interacción con el servicio Google Cloud Speech-to-Text.

Aiy.board: API encargada de la interacción del Push button que va conectado al Voice Bonnet. Una de sus ventajas es que al ser desarrollado por Google se puede conectar fácilmente con la plataforma Dialogflow.

## **2.4. Visión por Computador**

Para entender qué es la visión computarizada primero necesitamos entender el concepto de imagen. Para las personas una imagen es una representación visual de un objeto o de un espacio. Pero para las computadoras, una imagen es una matriz que tiene un valor numérico en cada uno de sus espacios llamados pixeles, que representan la imagen en una

computadora, un video es un determinado número de imagen en un determinado tiempo frames per second. [32]

¿Qué tener en cuenta en una imagen?

- Resolución en pixeles.
- Modo del color (RGB, BGR, HSV, Escala de grises)
- Tamaño de la imagen
- FPS (Vídeo)
- Compresión

En la inteligencia artificial siempre se ha intentado replicar el comportamiento o funciones humanas, en el caso de la visión artificial replicar la función que cumplen nuestros ojos por medio de cámaras y procesar estos datos para determinadas aplicaciones como la clasificación de imágenes. por ejemplo, en el campo médico cuando una computadora detecta si en una imagen de radiografía existe presencia de cáncer o no. en la industria cuando una cámara detecta si una fruta está en estado de cosecha o no, y así con cualquier tipo de aplicación que nos podamos imaginar. Otra gran aplicación de la visión computarizada es la detección de objetos, cuando podemos detectar múltiples objetos dentro de una misma imagen con su coordenada y la clase a la que pertenece, también tenemos aplicaciones en el arte, como transferencia de estilos y Red generativa antagónica GANs, y por último tenemos el reconocimiento facial, una de las aplicaciones más usadas de la visión computarizada implementado por gobiernos como el chino. [32]

#### **2.4.1. Aplicaciones de la visión computarizada**

En la actualidad la visión por computadora es usada y mejorada constantemente por grandes empresas en diferentes sectores de la industria a continuación daremos algunos ejemplos de donde se utiliza esta tecnología y la importancia que representa.

##### **2.4.1.1. Vehículos autónomos**

Tesla, BMW, Volvo y Audi utilizan múltiples cámaras, lidar, radar y sensores ultrasónicos para que sus autos autónomos puedan obtener imágenes que al ser procesada por la IA de los vehículos la convierte en información valiosa para la navegación por el entorno detectando autos, peatones, señales de tránsito y de más variables importantes en la seguridad de la conducción.

##### **2.4.1.2. Aplicación Google Translate**

Esta aplicación es una poderosa herramienta que nos permite traducir una oración por ejemplo escrita en un libro solo con nuestro celular y la cámara incorporada a él. todo esto mediante software de reconocimiento de símbolos o caracteres de cientos de distintas lenguas.

### 2.4.1.3. Reconocimiento facial

China definitivamente está a la vanguardia del uso de la tecnología de reconocimiento facial, y la usan para el trabajo policial, también grandes empresas como Apple con sus dispositivos que se pueden desbloquear por medio del reconocimiento facial y deben garantizar la seguridad y privacidad del sistema.

### 2.4.1.4. Cuidado de la salud

Dado que el 90 por ciento de todos los datos médicos se basan en imágenes, hay una gran cantidad de usos para la visión por computadora en medicina. Ayudando a los diagnósticos de enfermedades analizando radiografías, mamografías y otras exploraciones. hasta monitorear pacientes para identificar problemas antes y ayudar con la cirugía. [33]

## 2.4.2. Redes Convolucionales

La red neuronal convolucional (CNN) es una red neuronal especializada en el reconocimiento de imágenes, intenta simular el comportamiento de la corteza visual de nuestro cerebro, analiza los bordes, texturas, formas, colores y demás características tal y como lo hace nuestro cerebro.

El proceso es primero a la imagen, se le aplica una convolución, que es un filtro el cual recorre cada píxel de la imagen y reconoce todas las características o patrones mencionadas anteriormente. [32]

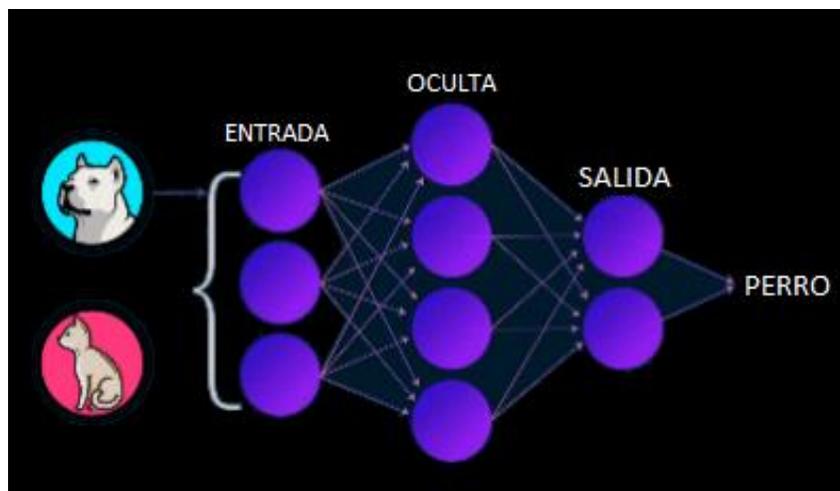


Figura 5. Red convolucional [29]

### 2.4.3. Frameworks

Existen distintos frameworks para el manejo de la inteligencia artificial como, tensorflow, theano, keras, pytorch, estos se utilizan para diseñar y ejecutar modelos de inteligencia artificial, es un esquema o marco de trabajo que ofrece una estructura base para la

organización y desarrollo de un software. en nuestro proyecto utilizaremos Pytorch la cual es una biblioteca de aprendizaje automático (machine learning), PyTorch es de código abierto y se utiliza para aplicaciones que implementan visión artificial y procesamiento de lenguajes naturales, es desarrollado en el laboratorio de inteligencia artificial de Facebook, siendo un software libre que utiliza una gran cantidad de piezas de software de Aprendizaje Profundo como Tesla Autopilot.

#### 2.4.4. Detección de Objetos

La detección de objetos es una tecnología de ordenador relacionada con la visión artificial, esta tecnología trata de detectar objetos en vídeos e imágenes digitales.

La detección de objetos tiene 2 características principales, la primera es la localización del objeto, que se trata de un recuadro llamado bounding box, este me dice en que ubicación de mi imagen se encuentra el objeto con coordenadas X y Y. La segunda característica es nuestro clasificador, definiendo la clase del objeto en la detección.

##### 2.4.4.1. Arquitectura de la Detección de Objetos

Lo primero es nuestra imagen de entrada, seguida por un filtro convolucional que nos permite extraer características principales por medio de la multiplicación de unos filtros o matrices convolucionales, después de esto pasamos a un filtro pool, que nos permite reducir dimensionalidad, posterior a esto pasamos al flatten, la capa que nos convierte la información de matriz a array, y por último a las capas que agregan el bounding box y la clase del objeto. [32] [34]



Figura 6 . Arquitectura5 de la Detección de Objetos [29]

##### 2.4.4.2. Métricas de Desempeño

Además de las dos métricas comunes accuracy y loss se tiene otra métrica la cual es Intersección entre uniones (IoU) que se utiliza para medir el solapamiento de dos o más bounding boxes.

Existe un problema cuando el objeto es más grande que estas ventanas, pues será categorizado múltiples veces en varios bounding boxes, que se resuelve con selección selectiva, o búsqueda selectiva con Non maximum suppression (NMS), la cual Es una variable que define un umbral de IoU, el cual define cuándo se deben considerar dos o más bounding boxes de un mismo objeto. Si una bounding box no supera este umbral, es descartada. De lo contrario es almacenada, Al final, la bounding box elegida es aquella que tiene mayor IoU. [32]

### 2.4.5. Algoritmos de Detección

Los algoritmos de detección de objetos más utilizados son los siguientes explicaremos brevemente y profundizaremos en el empleado para este proyecto SSD

#### 2.4.5.1. R-CNN

Region Based Convolutional Neural Networks, Ross Girshicketal, propuso un método en el que usamos la búsqueda selectiva para extraer solo 2000 regiones de las imágenes, estas regiones se generan utilizando el algoritmo de búsqueda selectiva, las regiones candidatas se deforman en un cuadrado y se alimentan a una red neuronal convolucional, que produce un vector de características de 4096 dimensiones de salida. Las características extraídas se alimentan a una máquina de soporte vectorial que clasifica la presencia del objeto dentro de las regiones candidatas. este algoritmo posee varios inconvenientes, uno de ellos es el no poder implementar en tiempo real, ya que toma alrededor de 47 segundos para cada imagen de prueba.

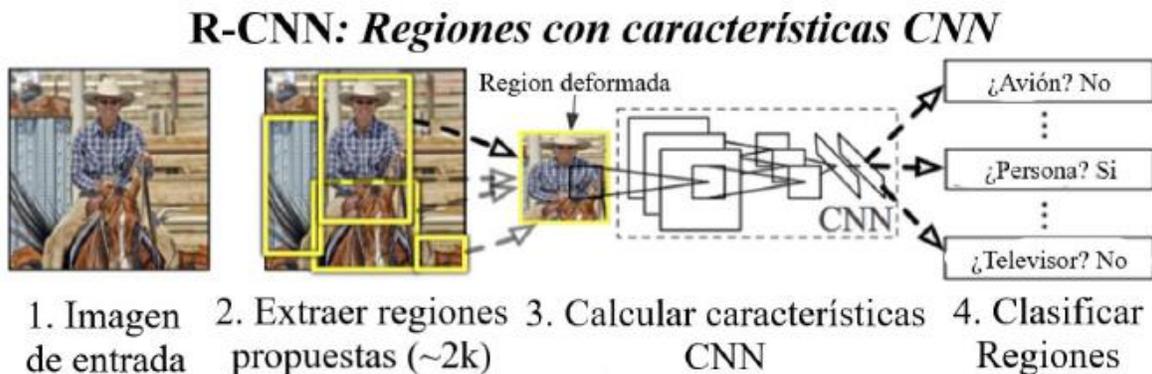


Figura 7. R6-CNN [32]

#### 2.4.5.2. Faster R-CNN

Faster Region Based Convolutional Neural Networks, este método elimina la búsqueda selectiva y en su lugar emplea una red neuronal (RPN o Región Proposal Network) para determinar las regiones de interés, lo que lo convierte en un algoritmo mucho más rápido que sus anteriores versiones, permitiendo que se pueda realizar detección de objetos en tiempo real y su precisión es bastante buena a la hora de detectar objetos [35]

### 2.4.5.3. YOLO

Este algoritmo fue creado en 2005 por Joseph Redmon junto con el grupo de investigación de inteligencia artificial de Facebook.

Yolo nos arroja como resultado un cuadro delimitador y la predicción para el objeto en dicho cuadro. además de esto es un modelo extremadamente rápido llegando en sus mejores versiones hasta los 155 fotogramas por segundo, está basado en el modelo de Darknet, en parte mantiene su estructura y pesos, solamente agregando algunas capas adicionales y optimizaciones específicas, esto se conoce como Transfer learning, es decir, un modelo pre entrenado en una estructura más grande con un enfoque diferente. Utiliza toda la imagen como entrada y no necesita crear regiones de interés y analizar varias veces una misma región, eliminar duplicados de un objeto, entre otros procesos que toman bastante tiempo, de aquí su nombre “You Only Look Once” (YOLO), en español “solo mira una vez”.

Este algoritmo tiene limitaciones en cuanto al número de objetos que se pueden detectar, lo cual hace que se vea limitado en la detección de objetos presentados en grupos. Yolo V4 fue diseñado en 2020 por Alexey Bochkovskiy, Chien-Yao Wang y Hong-yuan Mark Liao, y mostraron una notable mejora en el desempeño, velocidad y precisión. [36] [35]

### 2.4.5.4. SSD

Single Shot Detector es el algoritmo que se usa en el curso.

Es un algoritmo diseñado para la detección de objetos en tiempo real. SSD acelera el proceso que realiza Faster R-CNN al eliminar la necesidad de la red de propuesta de región. Y para recuperar la caída en la precisión que esto genera, SSD aplica algunas mejoras de escala múltiple y cuadros predeterminados. [35]

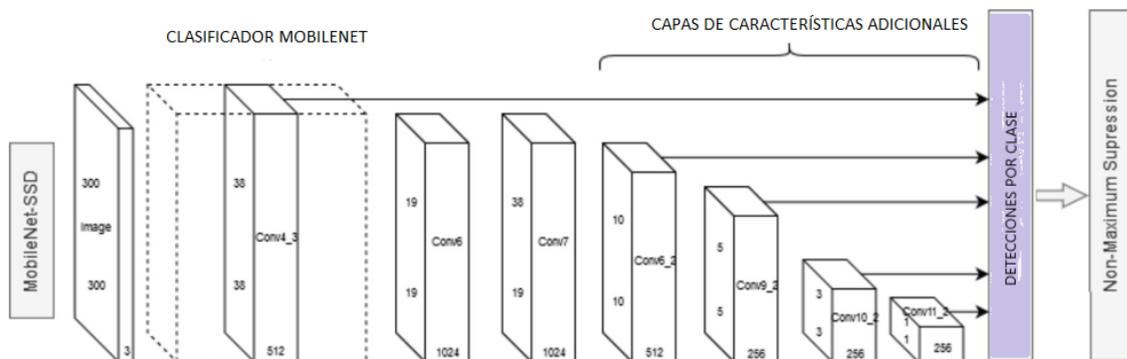


Figura 8. Mobilnet-ssd [31]

El primer paso es la entrada una imagen fija de 300 x 300 píxeles, para pasar a un clasificador, en nuestro caso Mobilenet, que es un modelo de visión capaz de distinguir entre 1000 clases de objetos diferentes.

## MobileNet V1

- Es el clasificador, es perteneciente a Google y es de bajo coste.
- Aplica la convolución a cada canal por separado, por ejemplo, en una imagen RGB hay un canal R, un canal G y uno B, en lugar de a toda la matriz, esto ahorra recursos.
- DeepWise: Separa las dimensiones de cada color de la matriz en la parte inicial de la red convolucional.
- PointWise: Da sentido a los resultados individuales de cada dimensión de color, juntándolos.
- Otra ventaja es que aplican conjuntos de filtros unidos en lugar de los filtros tradicionales.
- Originalmente clasificaba sólo 1000 objetos.

## MobileNet V2

- Mejora la V1 agregando un proceso llamado expansión, si la imagen es pequeña, la expande añadiendo dimensiones hasta 144, haciendo a la red más robusta.
- Projection Layer: Es una capa que devuelve la imagen a sus dimensiones originales.

Después de esto aplica capas y filtros convolucionales que nos permiten extraer las características de la imagen y hacer una reducción de dimensionalidad, por último, se aplica Non maximum suppression (NMS) que nos permite tomar el mejor bounding box. [32]

### **3. OBJETIVOS**

#### **3.1. General**

Desarrollar una herramienta que identifique automáticamente Objetos requeridos para el desempeño de actividades diarias, para proporcionar asistencia de voz a personas con déficits cognitivos.

#### **3.2. Específicos**

Objetivos específicos del proyecto.

- a) Detectar automáticamente objetos de cocina
- b) Asistir en la ejecución de una actividad cotidiana de preparación de alimentos, a través de indicaciones verbales mediadas por un asistente de voz.
- c) Evaluar la eficacia del sistema durante la ejecución de una actividad de preparación de alimentos en un contexto simulado.

## 4. METODOLOGÍA

El primer paso en nuestra metodología fue buscar una tarjeta embebida para la ejecución de algoritmos de aprendizaje profundo en tiempo real, para esto teníamos diferentes opciones de distintas marcas como, la Google coral y la raspberry, pero por términos de potencia y facilidad en la ejecución y entrenamiento de los modelos, optamos por la jetson nano.

### 4.1. Nvidia Jetson Nano

La Jetson Nano es una tarjeta de computación integrada de Nvidia, que se diseñó para potenciar los dispositivos y aplicaciones básicas de IA en el Edge. es la herramienta perfecta para comenzar a aprender la IA y la robótica en entornos del mundo real, con proyectos listos para usarse como grabadoras de video en red (NVR) básicas, robots domésticos y puertas de enlace inteligentes con capacidad de análisis completas.

Jetson Nano es compatible con NVIDIA JetPack™, con el mismo conjunto de software CUDA-X™ que se utiliza para productos innovadores basados en IA en todas las industrias. JetPack es la solución más completa para crear aplicaciones de IA. Para acelerar su aplicación de IA de extremo a extremo, incluimos TensorRT y cuDNN para acelerar la inferencia de IA. [34]

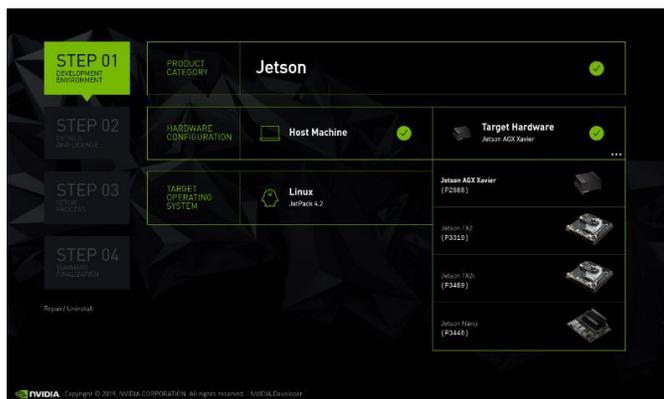


Figura 9. Tutorial instalación Jetpack.

Para instalar el jetpack solo necesitamos una memoria SD, a la cual le instalaremos la imagen del sistema JetPack que simplifica la instalación del sistema operativo, los controladores y contiene los siguientes componentes: [34]

- Núcleo L4T/BSP
- Kit de herramientas CUDA
- cuDNN
- TensorRT
- OpenCV
- VisionWorks
- API multimedia

### 4.1.1. Nvidia TensorRT

TensorRT es un Kit de desarrollo de Software Diseñado para aprendizaje profundo de alto rendimiento, incluye un optimizador de aprendizaje profundo y un tiempo de ejecución con baja latencia, funcionando hasta 36 veces más rápido que las plataformas solo de CPU, ya que este se enfoca específicamente en ejecutar una red ya entrenada de manera rápida y eficiente en el hardware de NVIDIA. [37]

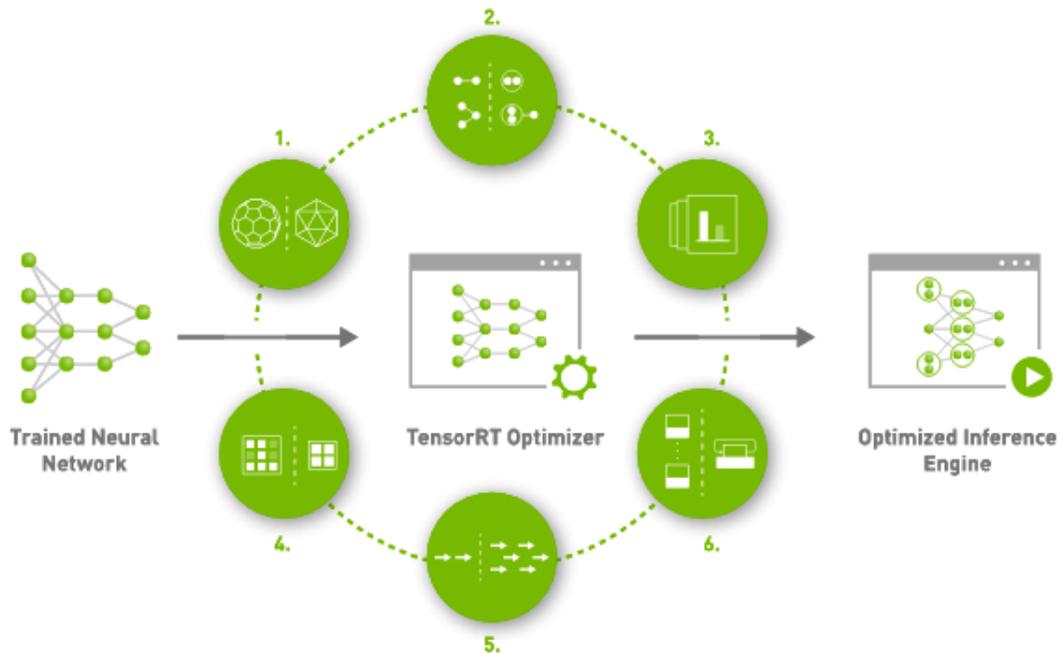


Figura 10. TensorRT [34]

### 4.2. Repositorio GitHub

Una vez instalado el jetpack procedemos a clonar el repositorio creado por desarrolladores de nvidia, una guía básica para implementar redes de aprendizaje profundo y primitivas de visión profunda con TensorRT y NVIDIA Jetson.

Con el repositorio se proporciona una biblioteca de redes de aprendizaje profundo aceleradas por TensorRT para reconocimiento de imágenes, detección de objetos con localización (es decir, cuadros delimitadores) y segmentación semántica. Esta biblioteca está específicamente diseñada para ejecutarse en Jetson y se descargan automáticamente varios modelos pre entrenados para una rápida ejecución. [34]

### 4.3. Base de datos

Open Images dataset v6 es un conjunto de datos de 9 millones de imágenes anotadas con etiquetas de nivel de imagen, cuadros delimitadores de objetos, máscaras de segmentación de objetos, relaciones visuales y narraciones localizadas.

- Contiene un total de 16 millones de cuadros delimitadores para 600 clases de objetos en 1,9 millones de imágenes.
- El conjunto de datos se divide en un conjunto de entrenamiento, un conjunto de validación y un conjunto de pruebas. [38]

### 4.4. Transfer Learning with PyTorch

El aprendizaje por transferencia es una técnica muy eficaz utilizada para volver a entrenar un modelo de red neuronal profunda con un nuevo conjunto de datos, esto lleva mucho menos tiempo que entrenar una red neuronal de cero, cargándole los pesos de un modelo previamente entrenado para clasificar un conjunto de datos personalizado, en este trabajo usaremos la red SSD-MobileNet.

El entrenamiento se realizará en la nube con GPU discretas, específicamente en colab debido a los grandes conjuntos de datos que se utilizan y PyTorch es el marco de aprendizaje automático que usaremos.

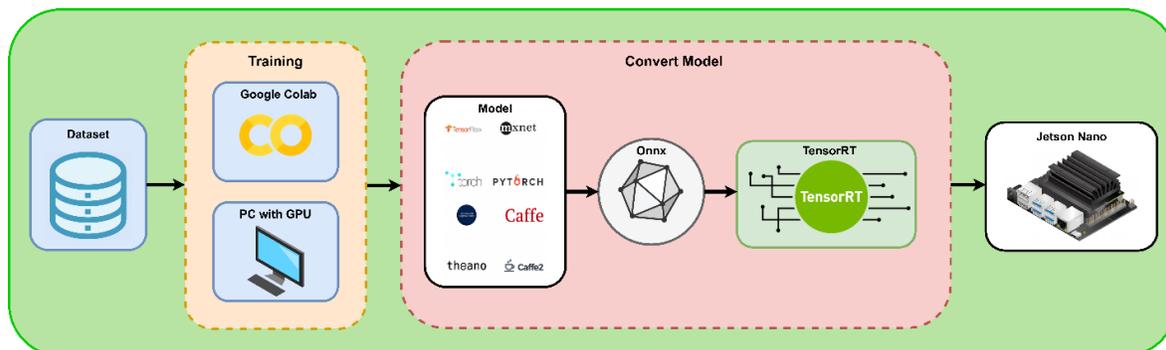


Figura 11. Transfer Learning [36]

## 4.5. Entrenamiento de Nuestro Propio Modelo

### 4.5.1. Descargar los datos

Descargamos los datos de la base de datos Open Images. Los datos ya han sido etiquetados por la base de datos y se descargan listos para el entrenamiento del modelo. las clases que usaremos son licuadora, plato hondo, gabinete, estufa de gas, mesa de cocina, utensilios de cocina, pocillo, plato, refrigerador y sartén. en total 10 clases para un total de 12275 imágenes con más de 27574 etiquetas utilizadas en el entrenamiento.

```
-----
'validation' set statistics
-----
Image count: 323
Bounding box count: 499
Bounding box distribution:
  Cabinetry: 188/499 = 0.38
  Plate: 66/499 = 0.13
  Kitchen utensil: 56/499 = 0.11
  Bowl: 51/499 = 0.10
  Kitchen & dining room table: 44/499 = 0.09
  Wok: 24/499 = 0.05
  Mug: 24/499 = 0.05
  Refrigerator: 19/499 = 0.04
  Gas stove: 17/499 = 0.03
  Blender: 10/499 = 0.02

(1)
```

```
-----
'test' set statistics
-----
Image count: 910
Bounding box count: 1415
Bounding box distribution:
  Cabinetry: 450/1415 = 0.32
  Plate: 212/1415 = 0.15
  Kitchen & dining room table: 157/1415 = 0.11
  Bowl: 152/1415 = 0.11
  Kitchen utensil: 141/1415 = 0.10
  Mug: 106/1415 = 0.07
  Refrigerator: 82/1415 = 0.06
  Gas stove: 49/1415 = 0.03
  Wok: 36/1415 = 0.03
  Blender: 30/1415 = 0.02

(2)
```

```
-----
'train' set statistics
-----
Image count: 11042
Bounding box count: 25660
Bounding box distribution:
  Cabinetry: 9087/25660 = 0.35
  Plate: 5221/25660 = 0.20
  Bowl: 4482/25660 = 0.17
  Mug: 2159/25660 = 0.08
  Kitchen & dining room table: 2124/25660 = 0.08
  Wok: 728/25660 = 0.03
  Refrigerator: 591/25660 = 0.02
  Kitchen utensil: 546/25660 = 0.02
  Gas stove: 509/25660 = 0.02
  Blender: 213/25660 = 0.01

(3)
```

Figura 12. Descarga de la base de datos carpeta 1,2 y 3.

### 4.5.2. Entrenamiento del modelo SSD-MobileNet

Una vez tenemos los datos descargados, corremos train\_ssd.py para empezar con el entrenamiento del modelo, algunas opciones comunes con las que puede ejecutar el script de entrenamiento son:

Argumento	Defecto	Descripción
--data	data/	la ubicación del conjunto de datos
--model-dir	models/	directorio para generar los puntos de control del modelo entrenado
--resume	Ninguna	ruta a un punto de control existente para reanudar el entrenamiento desde
--batch-size	4	intente aumentar dependiendo de la memoria disponible
--epochs	30	hasta 100 es deseable, pero aumentará el tiempo de entrenamiento
--workers	2	número de subprocesos del cargador de datos (0 = deshabilitar subprocesos múltiples)

Figura 13. Argumentos de Entrenamiento [31]

Nuestro modelo se entrena con alrededor de 40 épocas, las cuales nos ofrecen un modelo lo suficientemente confiable, sin que se gaste demasiado tiempo en el entrenamiento que realiza Google colab en su versión gratuita.

### 4.5.3. Conversión del modelo a ONNX

A continuación, se convierte el modelo entrenado de PyTorch a ONNX, para que pueda ser ejecutado en TensorRT. con el fin de que este mejore el rendimiento del modelo, específicamente en el hardware de Nvidia. [34]

### 4.5.4. Poner en Marcha Nuestro Modelo de Detección de Objetos

Para ejecutar nuestro modelo necesitamos una web cam conectada a nuestra Jetson nano y procedemos a ejecutar nuestro modelo con un comando.

```
detectnet --model=models/fruit/ssd-mobilenet.onnx --labels=models/fruit/labels.txt \
--input-blob=input_0 --output-cvg=scores --output-bbox=boxes \
"$IMAGES/fruit_*.jpg" $IMAGES/test/fruit_%i.jpg
```

Figura 14. Comando para ejecutar el código [31]

En este comando solo estamos señalando donde se encuentra nuestro modelo y donde se encuentra la información de entrada para que pueda ser procesada, para indicar que nuestra información de entrada es una cámara web debemos poner lo siguiente como parámetros de entrada.

MIPI CSI camera	csi://	csi://0	CSI camera 0 (substitute other camera numbers for 0)
-----------------	--------	---------	--

Figura 15. Comando cámara web [34]

Una vez definido, si todo se realizó correctamente la Jetson Nano comenzará a ejecutar nuestro modelo de detección de objetos en tiempo real.

### 4.6. Asistente de Voz

Para la construcción del asistente de voz se decidió utilizar la plataforma de Dialogflow, esto debido a que trabajaremos en conjunto con otras tecnologías de Google como lo son el

Voice kit, Google Cloud y Firebase, además de estos nos da la opción de incorporarlo fácilmente con el Google Assistant facilitando el testeo de los flujos que se definen en la experiencia conversacional.

#### 4.6.1. Diseño de la experiencia conversacional

Primero se debe definir de manera consistente si la aplicación de la experiencia conversacional que se quiere implementar es relevante, para ello se elaboró una tabla de verificación con los puntos que se consideran más relevantes.

##### 4.6.1.1. Caso de uso

Implementar un asistente capaz de entablar una conversación para el apoyo o asistencia de personas con déficits cognitivos en actividades de la vida diaria.

Criterio	Puntos para evaluar	Valoración	Calificación
Conversación intuitiva: Permite a los usuarios comunicar cuál es su necesidad en cualquier momento de manera natural.	<ul style="list-style-type: none"> <li>¿Los usuarios normalmente entablan conversaciones del tema o tarea?</li> <li>La interacción es corta, con poca ida y vuelta</li> </ul>	En la realización de actividades de la vida diaria no se suele entablar conversaciones, sin embargo, se conoce el contexto y el lenguaje utilizado es simple. Las interacciones implican preguntas y respuestas rápidas.	Positiva
Conversación eficiente: Permite reducir puntos de tensión brindando al usuario lo que necesita de manera rápida.	<ul style="list-style-type: none"> <li>La conversación ahorra tiempo y esfuerzo.</li> <li>Los usuarios tienen que acudir a diversas aplicaciones para completar la tarea.</li> </ul>	Un asistente de voz es una herramienta de fácil acceso en estos casos ya que los usuarios se les puede dificultar el uso de otras tecnologías que requieran mayor manipulación, sin embargo, en algunos casos no llega a ser suficiente y requiere el apoyo de un profesional.	Negativa
Multitasking: Ayuda a los usuarios cuando están ocupados	<ul style="list-style-type: none"> <li>Se puede llevar a cabo la tarea mientras se realizan otras actividades.</li> <li>Los usuarios pueden realizar esta tarea cuando sus manos o sus ojos están ocupados</li> </ul>	En general en las AVDs es de gran utilidad ya que la asistencia es para eliminar puntos de tensión que surjan en el transcurso de una actividad, contribuyendo a la finalización satisfactoria de esta.	Positiva

<b>Privacidad:</b> Las conversaciones no tocan temas personales.	<ul style="list-style-type: none"> <li>• Los usuarios se sienten cómodos hablando acerca del tema.</li> </ul>	Aunque algunas actividades como por ejemplo vestirse o bañarse son de carácter privado, la interacción no implica conversaciones con preguntas o respuestas personales, asimismo el apoyo está planteado cuando el usuario lo solicite.	Neutral
---	---	---	---------

*Tabla 1. Lista de verificación para determinar el impacto de la experiencia conversacional.*

En general se obtuvo una calificación favorable, lo que nos indica que es probable que el desarrollo de esta experiencia conversacional es relevante, por lo cual se procede al diseño y construcción esta.

#### **4.6.1.2. Identidad del asistente**

**Nombre:** Doctora

**Audiencia:** Personas con déficits cognitivos.

**Características:** Inclusión, superación personal, apoyo a sus usuarios.

**Tono de voz:**

Se debe pensar en el asistente como una persona, de esta manera nos preguntamos qué tipo de voz debe tener para que tenga una interacción exitosa con sus usuarios, algunas de las características al momento de escoger la voz que se consideran importantes son la familiaridad, el acento, debe sonar como alguien joven, amigable y de ser posible poder implementar una voz igual o parecida a la de la persona que suele acompañar al usuario final, ya sea un profesional de la salud o un familiar que conviva con ella.

**Acciones:**

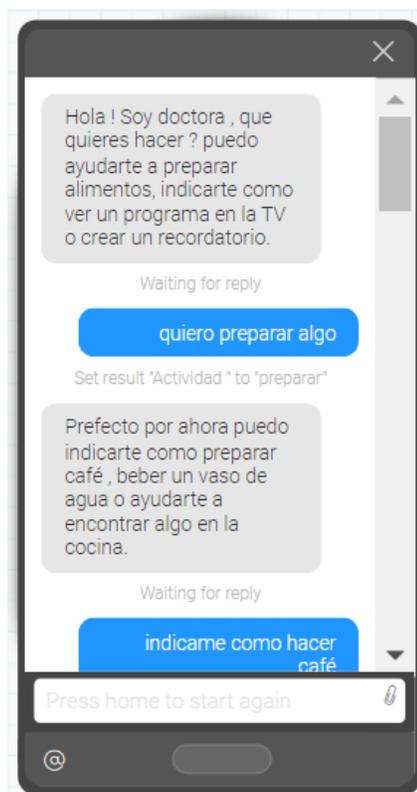
En este apartado se abordarán tres puntos que nos ayudaran a encaminar cuales son los casos de uso principales y que se quiere lograr con ellos, para ello se realiza una lista de ideas de lo que podría incluir el asistente en cada punto.

1. Servicios y conocimiento: posibles actividades principales.
  - Guías fáciles para preparación de alimentos.
  - Apoyo para la realización de actividades de la vida diaria.
  - Herramienta para profesionales de terapia ocupacional.
2. Ideas de acciones: como se podrían resolver los puntos de tensión de los usuarios.
  - Ayudar a Seguir el objetivo de una tarea hasta su finalización de una tarea no completada.
  - Orientación espacio/temporal del usuario.
  - Indicaciones para la ubicación de objetos.

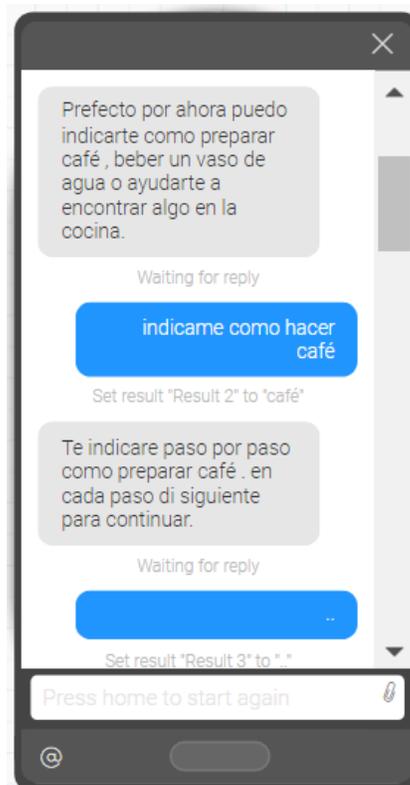
3. Acción principal: Tarea más factible para resolver el problema más urgente.
- Ayudar en la realización de las AVDs, proporcionando asistencia cuando el usuario lo solicite o cuando necesite continuar el proceso para finalizar una tarea que no ha sido completada. Esta acción se realizará indicando los pasos a seguir para efectuar la actividad requerida.

#### 4.6.1.3. Diálogos de prueba

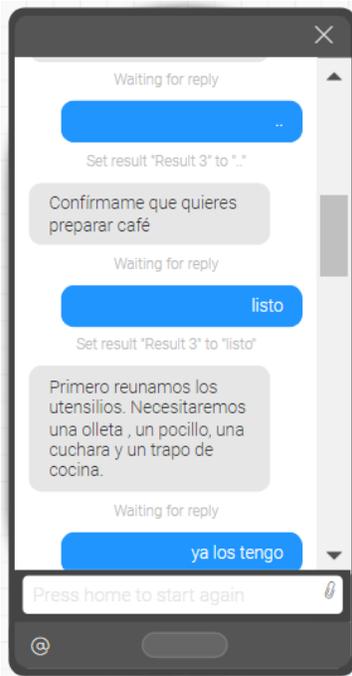
Los diálogos de prueba son el punto de partida para saber cómo sería la interacción que se está diseñando, esto ayuda a centrarse en las necesidades del usuario sin estar pensando en problemas de programación, contextos gramaticales o diagramas de flujo conversacionales más complicados, lo que nos evita perder el flujo de la conversación que podría ocasionar un diseño de interfaz poco natural.



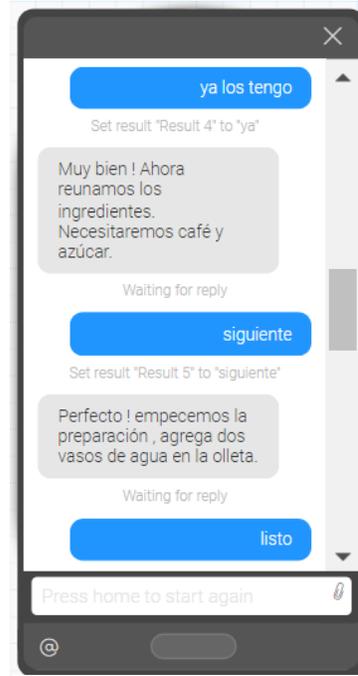
(1)



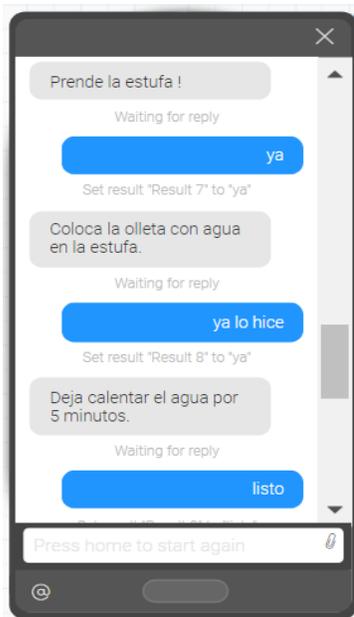
(2)



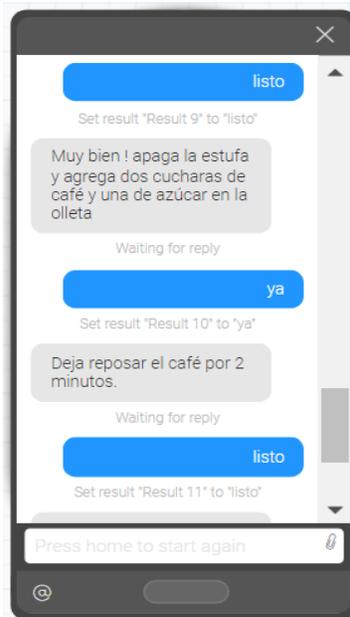
(3)



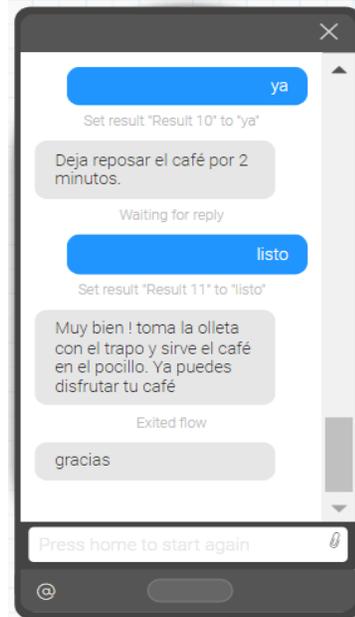
(4)



(5)



(6)



(7)

Figura 16. Secuencia de diálogo de prueba

Para la construcción del primer dialogo de prueba se toma en cuenta un caso de una conversación ideal, en la cual obviamos respuestas inesperadas que puedan desviar el flujo de la conversación.

#### 4.6.2. Construcción del Diagrama Principal

Teniendo en cuenta el dialogo de prueba se procede a construir un diagrama de flujo, el cual nos permite identificar posibles problemas que puedan surgir en el transcurso de la conversación y así poder establecer una forma de mantener al usuario dentro de la conversación establecida para completar la tarea requerida con éxito.

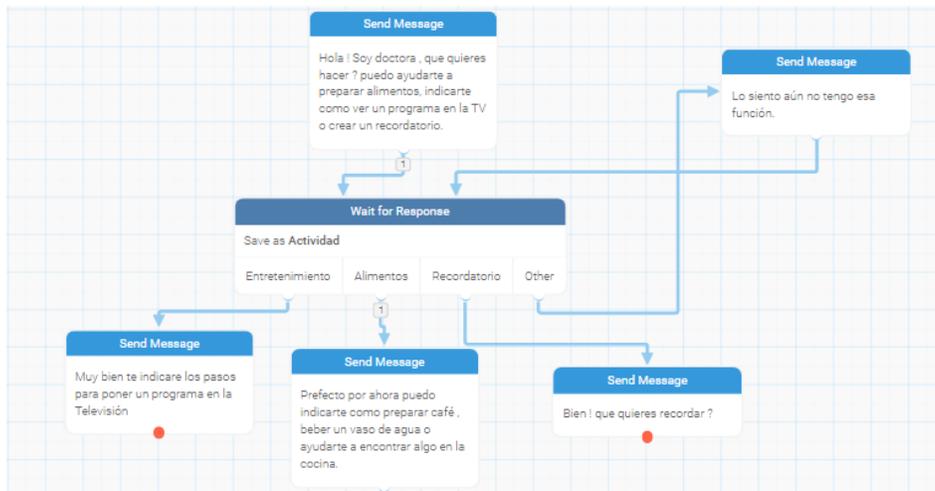


Figura 17 . Diagrama principal inicial del flujo de conversacional planteado.

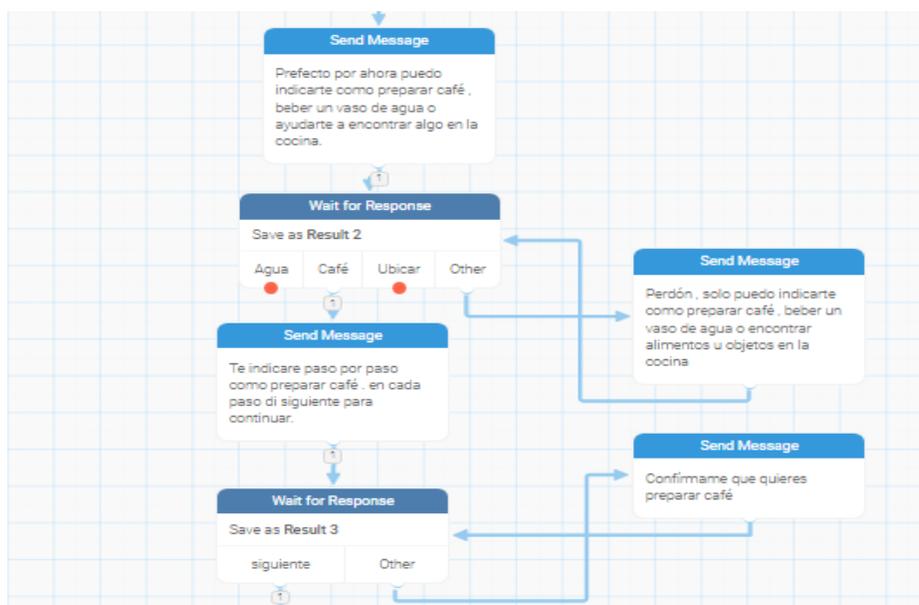


Figura 18. Lógica inicial de los pasos indicados para completar una tarea

En la Figura 17 se muestra el saludo inicial y las posibilidades que tiene el usuario para continuar la conversación, es posible escoger indicaciones para preparar alimentos, ver un programa en la TV y crear un recordatorio. En este diagrama se incluye solamente una ruta en la que el usuario desea preparar alimentos, específicamente los pasos para preparar un café. en la Figura 18 se muestra la lógica que tendría cada paso, cada uno está formado por una indicación y se espera por una confirmación, en caso de que no se dé la confirmación se indica al usuario que debe realizarla para poder continuar, esto permite mantener el flujo de la conversación antes respuestas inesperadas.

#### 4.6.3. Implementación en Dialogflow

En la sección anterior se obtuvo un diseño inicial de la experiencia conversacional, ahora este se implementará con ayuda de Dialogflow.

El primer paso es identificar las identidades necesarias que se deben crear para que el agente pueda extraer la información relevante.

Creación de entidades

Entidad	Descripción
Alimentos	Diferentes alimentos que se incluyen en las preparaciones que podemos indicar.
Lugares	Cocina
Objetos-cocina	Elementos que usualmente encontramos en la cocina como: nevera, estufa, cubiertos, platos etc.
Objetos-habitación	Cosas que podemos encontrar en una habitación como el televisor, la cama, ropa etc.
Objetos-sala	Objetos como muebles, televisión, mesas etc.
Petición	Distintas maneras en que la persona pueda dar un inicio a la conversación
Verbos	Diferentes acciones incluidas en las AVDs, por ejemplo, cocinar, comer, vestirse, ver televisión etc.

*Tabla 2 Entidades adicionales utilizadas en Dialogflow.*

#### Creación de Intents

Los Intents iniciales pueden tener modificaciones hasta obtener el resultado esperado, es indispensable hacer pruebas con cada Intent para encontrar falencias y corregirlas hasta que se obtenga el resultado esperado en el flujo conversacional establecido en el diseño inicial.

Intent	Contexto de entrada	Contexto de salida	Frases de entrenamiento	Entidades utilizadas	Respuestas
Bienvenida	Ninguno	Ninguno	Ninguna	Ninguna	¡Hola! Soy doctora, estoy aquí para ayudarte. Solo pide ayuda cuando lo necesites.
Solicitar ayuda	Ninguno	Obtener Lugar	Necesito ayuda Ayúdame	Petición	¡Me encanta ayudarte! Puedo indicarte como preparar un café, ver la televisión y hasta decirte que hora es.
Felicitación	Preparar-café paso 9	Ninguno	Ninguna		¡Lo hiciste muy bien, eres increíble!
Obtener Lugar	Ninguno	Preparar-café paso 1	Estoy en la cocina Estoy en la sala Estoy en la habitación	Lugares	Indicaste que estamos en X lugar
Obtener acción	Ninguno	Preparar-café	Quiero hacer café	Objetos-cocina Objetos-sala Objetos-habitación Verbos Alimentos	Futuras implementaciones.
		Encender la TV	Ver televisión		
		Vestirse	Quiero vestirme		
		Tomar un vaso de agua	Quiero tomar algo		
Preparar-café	Solicitar ayuda	Preparar café paso 0	Quiero preparar café Café Tomar café	Verbos Alimentos	Te indicare paso por paso como preparar café. En cada paso puedes decir palabras como listo o siguiente para continuar.
Preparar café paso 0	Preparar-café	Preparar-café paso 1	Listo Ya lo hice Si Ya estoy	Ninguna	Para preparar café debes estar en la cocina, si ya lo estas confirmalo.
Preparar-café paso 1	Preparar-café	Preparar-café paso 2	Termine Siguiete		Primero reunamos los utensilios. Necesitaremos una olleta, un pocillo, una

					cuchara y un trapo de cocina.
Preparar-café paso 2	Preparar-café paso 1	Preparar-café paso 3			¡Muy bien! Ahora reunamos los ingredientes. Necesitaremos café y azúcar.
Preparar-café paso 3	Preparar-café paso 2	Preparar-café paso 4			¡Perfecto! empecemos la preparación, agrega dos vasos de agua en la olleta.
Preparar-café paso 4	Preparar-café paso 3	Preparar-café paso 5			¡Prende la estufa!
Preparar-café paso 5	Preparar-café paso 4	Preparar-café paso 6			Coloca la olleta con agua en la estufa.
Preparar-café paso 6	Preparar-café paso 5	Preparar-café paso 7			Deja calentar el agua por 5 minutos.
Preparar-café paso 7	Preparar-café paso 6	Preparar-café paso 8			¡Muy bien! apaga la estufa y agrega dos cucharas de café y una de azúcar en la olleta
Preparar-café paso 8	Preparar-café paso 7	Preparar-café paso 9			Deja reposar el café por 2 minutos.
Preparar-café paso 9	Preparar-café paso 8	Felicitación			¡Muy bien! toma la olleta con el trapo y sirve el café en el pocillo. Ya puedes disfrutar tu café
Encender la TV	Ninguno	Ninguno	Quiero ver Televisión	Verbos Objetos-sala-habitación	Continúo aprendiendo como realizar estas tareas.
			Televisión		
Tomar un vaso de agua.	Ninguno	Ninguno	Tomar agua	Verbos Alimentos	Continúo aprendiendo como realizar estas tareas.
			Beber agua		

Tabla 3 Entidades que usa el agente en la implementación del sistema.

En este punto es posible notar que el flujo de la conversación se vuelve difícil de mantener e interpretar. es aquí cuando el uso de los contextos se vuelve indispensable, esto nos permite enlazar los distintos intents. no todos los intents o acciones mencionadas en el flujo

de la conversación podrán ser implementados en la primera versión, ya que como se puede notar se vuelve complejo diseñar conversaciones con tantos pasos, por esto se hará énfasis en el flujo que nos indica como preparar un café.

Para hacer uso de la experiencia conversacional diseñada Dialogflow nos da la opción de integrarla con diversas aplicaciones, entre la cuales se incluyen el Google Assistant el cual puede ser utilizado en diversos dispositivos como, por ejemplo, móviles Android y iPhone, computadores, Speaker inteligentes, Smart watch y en nuestro caso el AIY voice kit.

Antes de proceder a la integración explicaremos brevemente los pasos más relevantes para hacer uso del Voice kit, el detalle de estos puede ser consultado en la [página oficial](#).

#### **4.6.4. Voice kit**

##### **4.6.4.1. Armando el kit:**

El Voice kit tiene todos los materiales necesarios para la construcción de nuestro parlante inteligente, en la caja podemos encontrar una Raspberry Pi Zero, Voice Bonnet, altavoz, contenedor armable para todos los componentes y otros elementos necesarios para completar el ensamble.

Antes que nada, se debe flashear la memoria microSD con la última versión del archivo imagen de AIY voice kit, el cual encontramos en el repositorio de [git](#), que incluye el sistema operativo para la Raspberry y el software de AIY, posterior a esto se procede al ensamble de los componentes incluidos en el kit.

##### **4.6.4.2. Conexión a internet**

Una vez hemos flasheado la microSD debemos insertarla en la Raspberry y alimentar el kit con un cargador de celular micro USB, debemos esperar alrededor de 5 minutos para que el sistema cargue completamente, ahora para conectar el kit a internet se debe descargar la app AIY Projects la cual nos da las instrucciones para realizar la conexión, al final de este proceso obtendremos una dirección IP la cual utilizaremos para realizar la conexión desde el computador vía SSH al kit, se debe tener en cuenta que el computador y el kit deben estar conectados a la misma red wi-fi.

##### **4.6.4.3. Configuración del asistente**

Ahora debemos conectar el ordenador a la Raspberry Pi usando SSH en un terminal, la manera más rápida de hacerlo es descargando la extensión Secure Shell que proporciona Chrome.

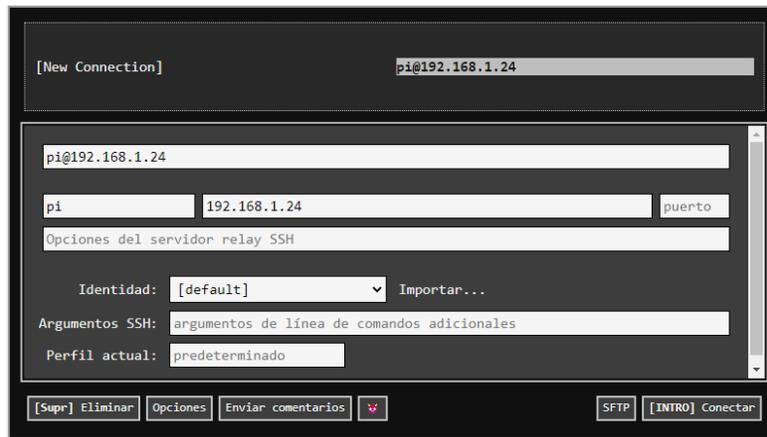


Figura 19. Ventana de Conexión usando Secure Shell.

Sustituiremos los números con la dirección IP obtenida anteriormente y daremos en el botón [ENTER] Connect, posteriormente proporcionaremos la contraseña, la cual es raspberry y si todo esta correcto ya habremos conectado nuestro ordenador a la raspberry.

Para poder utilizar las APIs del Google Assistant y Cloud Speech, debemos crear un proyecto en Google Cloud Platform, una vez creado se deben buscar las APIs requeridas, habilitarlas y crear las credenciales correspondientes, una vez creadas las descargamos y copiamos en el archivo nano assistant.json, el cual se encuentra en la raspberry y accederemos a él utilizando la terminal SSH, hasta, este momento ya podemos hacer uso de los demos que vienen incluidos por defecto en el kit.

Para iniciar con la parte final e integrar el Voice kit se debe crear un proyecto en la consola de Actions on Google, el cual nos permite interactuar con los distintos servicios de Google de los que haremos uso como, Google Assistant, Dialogflow y Firebase, este último nos da varias funcionalidades que podemos agregar al desarrollo, una de ellas es Firebase Functions, que nos permite hacer un Deploy del código que sea desarrollado en un entorno local e incluirlo al proyecto desarrollado en Dialogflow, además también brinda la posibilidad de enlazar una base de datos que puede ser de utilidad para futuros desarrollos.

Finalmente basta con correr el demo assistant\_grpc\_demo.py el cual nos permite hacer uso del Google Assistant, al que pediremos invocar la versión de prueba de la experiencia conversacional que se ha diseñado, esto se hace con la frase de invocación "Hablar con Doctora". Ahora ya podemos hablar con nuestro asistente y probar los flujos establecidos.

#### 4.6.5. Aplicación de prueba.

Se realizaron pruebas con 4 personas de distintas edades, en el rango de los 22 a los 80 años, estas personas no tenían ningún conocimiento de los flujos de conversación diseñados, solo se les indica la frase clave para invocar al asistente, luego de esto debían seguir las instrucciones del asistente y contestar las preguntas realizadas por el mismo. Cada persona tuvo que repetir varias veces la prueba, esto con el objetivo que llegaran a completar el flujo de la tarea "Pasos para preparar un café".

## 5. RESULTADOS

### 5.1. Modelo de Reconocimiento de Objetos

Las máquinas aprenden mediante una función de pérdida. Es un método para evaluar qué tan bien funciona un algoritmo en específico. Si las predicciones se desvían demasiado de los resultados reales, la función de pérdida en Machine learning arrojaría un número muy grande en nuestro algoritmo, se puede ver que a medida que el tiempo transcurre, nuestra función de pérdida disminuye.

```
Época: 0, Paso: 10/1287, Pérdida promedio: 12,4240, Pérdida de regresión promedio 3,5747, Pérdida de clasificación promedio: 8,8493
Época: 0, Paso: 20/1287, Pérdida media: 9,6947, Pérdida de regresión media 4,1911, Pérdida de clasificación media: 5,5036
Época: 0, Paso: 30/1287, Pérdida promedio: 8.7409, Pérdida de regresión promedio 3.4078, Pérdida de clasificación promedio: 5.3332
Época: 0, Paso: 40/1287, Pérdida media: 7,3736, Pérdida de regresión media 2,5356, Pérdida de clasificación media: 4,8379
Época: 0, Paso: 50/1287, Pérdida promedio: 6,3461, Pérdida de regresión promedio 2,2286, Pérdida de clasificación promedio: 4,1175

Época: 0, Pérdida de validación: 5,6730, Pérdida de regresión de validación 1,7096, Pérdida de clasificación de validación: 3,9634
Modelo guardado models/fruit/mb1-ssd-Epoch-0-Loss-5.672993580500285.pth
```

Nuestro modelo de detección de objetos fue entrenado para reconocer 10 objetos de la cocina, por esto se realizaron las pruebas en 2 cocinas.

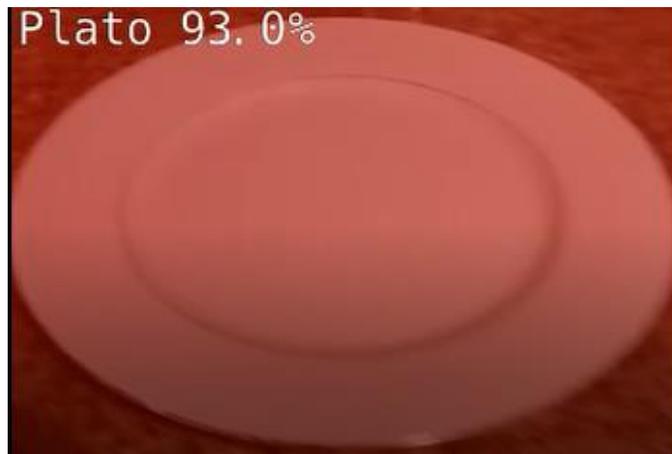


Figura 20. Prueba detección de objetos

Nuestro modelo ubica el objeto mediante un bounding box y lo clasifica mediante una etiqueta y un score en la parte que superior, que indican la clase del objeto y la precisión de esta clasificación, si se presenta un objeto que no está incluido en nuestras clases el modelo no deberá mostrar nada sobre este objeto, y se tendría que tomar como un error de clasificación.

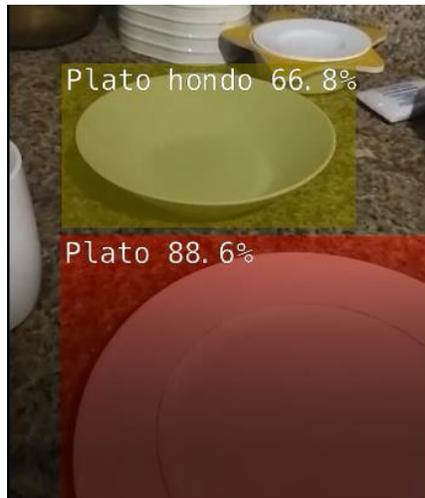


Figura 21. Prueba detección de objetos

## 5.2. Asistente de Voz

En las pruebas realizadas es posible notar que algunas palabras no son captadas correctamente por el asistente, ocasionando problemas inicialmente con el llamado, ya que era necesario que se reconociera específicamente la frase “hablar con doctora”, sin embargo, a medida que se realizaron distintas pruebas las palabras fueron captadas con mayor facilidad e incluso se logró invocar el asistente con palabras como "Necesito hablar con la doctora" o "Hola doctora" que inicialmente no funcionaban. Por otra parte, se obtuvo una realimentación de las personas a las que se les aplicó la prueba, en el caso de la persona 2, se identifica que los problemas para seguir el flujo de conversación son debido a la formulación de las frases, ya que estas eran principalmente afirmativas pero no incluían una pregunta que obligaran de alguna manera que la persona respondiera para continuar con el flujo, una vez aplicada esta modificación en las respuestas se procede aplicar la prueba en la persona 3, en la cual se pudo notar indisposición e impaciencia para interactuar con el asistente ya que en el diseño inicial se establecen algunos parámetros requeridos para continuar la conversación, pero no se incluyeron frases de sugerencia que permitieran a la persona identificar dichos parámetros. Finalmente, la persona 4 tuvo una experiencia más satisfactoria gracias a las frases de sugerencia incluidas, además de esto logró identificar por su propia cuenta la estructura que debía usar, sin embargo, menciona que en un principio se es muy insistente e inflexible con estas frases.

A continuación, la Tabla 4 . Realimentación de las pruebas aplicadas y observaciones. detalla lo sucedido con cada una de las personas a las que se aplicó la prueba.

Sujeto	Comentarios	Observaciones
Persona 1: Hombre de 22 años	Manifiesta que es fácil y entretenido de utilizar pero que no lo usaría en su vida diaria.	Realizó una interacción exitosa con el asistente, no tuvo problemas para seguir las indicaciones y el contexto que planteaba al flujo conversacional.
Persona 2: Mujer de 28 años.	La mujer expresa que durante la interacción no sintió que el asistente le guiara por un flujo ya que no se vio obligada a dar una respuesta a la frase expresada por el asistente.	Esta persona tuvo algunos problemas para continuar el flujo de la conversación, al empezar la interacción el asistente saluda y sugiere que puede ayudar, sin embargo la mujer no solicitó la ayuda, por lo cual tuvo que darse la indicación de solicitar ayuda para continuar con el proceso, seguidamente el asistente responde con una serie de tareas en las que podría brindar asistencia pero nuevamente la mujer no escogió ninguna tarea en específico y tuvo que darse la indicación correspondiente para continuar, una vez la conversación estaba en el punto de indicar como hacer la tarea de preparar café, la mujer no tuvo problema en seguir el flujo conversacional establecido para completar todos los pasos dados por el asistente.
Persona 3: Mujer de 54 años.	Esta persona sugirió que el asistente tardaba mucho en dar respuesta a sus peticiones y le causaba impaciencia puesto que no sabía si volver a preguntar o esperar.	Esta persona siguió bien el flujo de la conversación hasta el momento en que el asistente indica que dará los pasos necesarios para preparar café, ya que en ese punto la mujer no espera el tiempo suficiente entre cada indicación que da el asistente y empieza hacer preguntas insistentemente, además a la hora de proporcionar las confirmaciones que pide el asistente en cada paso, las repite muchas veces lo que impide que el asistente pare de tomar su respuesta para procesarla y en cambio sigue captando todo lo que la mujer dice, lo cual entorpece el flujo y no logra completar todos los pasos.
Persona 4: Hombre de 80 años.	Este hombre se muestra conforme con la interacción, sin embargo, expresa que el asistente puede ser insistente en las frases de confirmación y que una persona de su edad abandonaría	Logra llamar correctamente al asistente, solicitar ayuda y pedir las indicaciones para preparar un café, sin embargo cuando el asistente le solicita que debe estar en la cocina para preparar el café la persona responde afirmativamente pero el asistente sigue repitiendo que debe estar en la cocina, esto debido a que la frase de confirmación incluida en el diseño debe estar formada por cualquier palabra afirmativa como si, listo o ya,

	fácilmente la tarea de no poder avanzar.	pero también debe incluir la palabra cocina, a pesar de esto ante la insistencia del asistente en la frase "Para preparar café debes estar en la cocina, si ya lo estas confirmalo." el hombre logra avanzar indicando "Ya estoy en la cocina", seguidamente el asistente continua con la frase "Primero reunamos los utensilios, necesitaremos una olleta, un pocillo, una cuchara y un trapo de cocina" a lo que el hombre responde afirmativamente que ya los tiene, pero no es posible avanzar hasta que se den los parámetros requeridos en la frase de confirmación utilizada para mantener a la persona dentro del flujo, es aquí cuando el asistente responde con la frase "Recuerda que son cuatro utensilios cuchara, pocillo, olleta y trapo, cuando los tengas todos puedes decir ya tengo los utensilios", el hombre logra identificar la palabra sugerida en la instrucción y continuar con el flujo. El siguiente paso se indica en la frase "¡Muy bien! Ahora reunamos los ingredientes, necesitaremos café y azúcar.", y el hombre logra continuar con el flujo sin necesidad de sugerir la frase de confirmación respondiendo "Ya tengo los ingredientes". respuesta que tiene la misma estructura sugerida previamente en el flujo, de esta manera el hombre logra completar exitosamente toda la conversación.
--	--	--

*Tabla 4 . Realimentación de las pruebas aplicadas y observaciones.*

### **5.2.1. Diagrama final**

Después de las pruebas realizadas se obtuvo el diagrama final, en el que se incluyen los pasos, solicitudes de usuario, toma de decisiones, clasificación de intenciones y frases utilizadas por el agente, no se incluyen todos los pasos para no hacerlo tan extenso, se pueden tomar como referencia los pasos 2 y 3 que tienen la misma estructura que los siguientes.

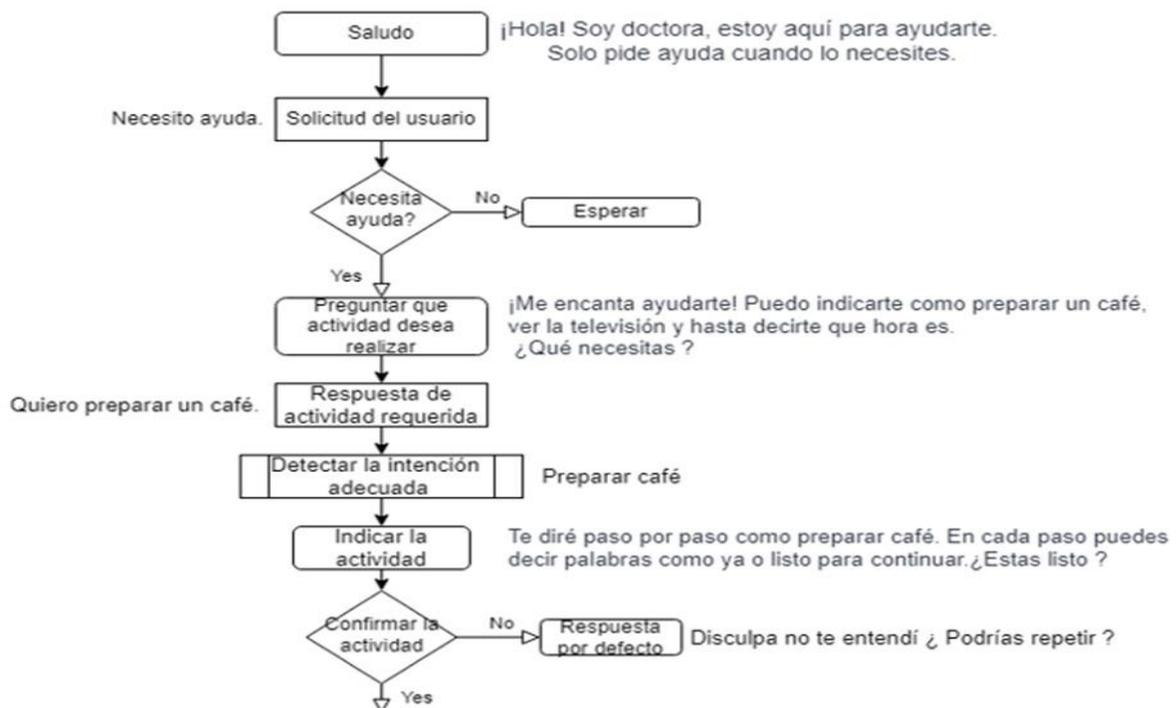


Figura 22. Diagrama de flujo que establecerá la lógica final que implementará el agente parte a.

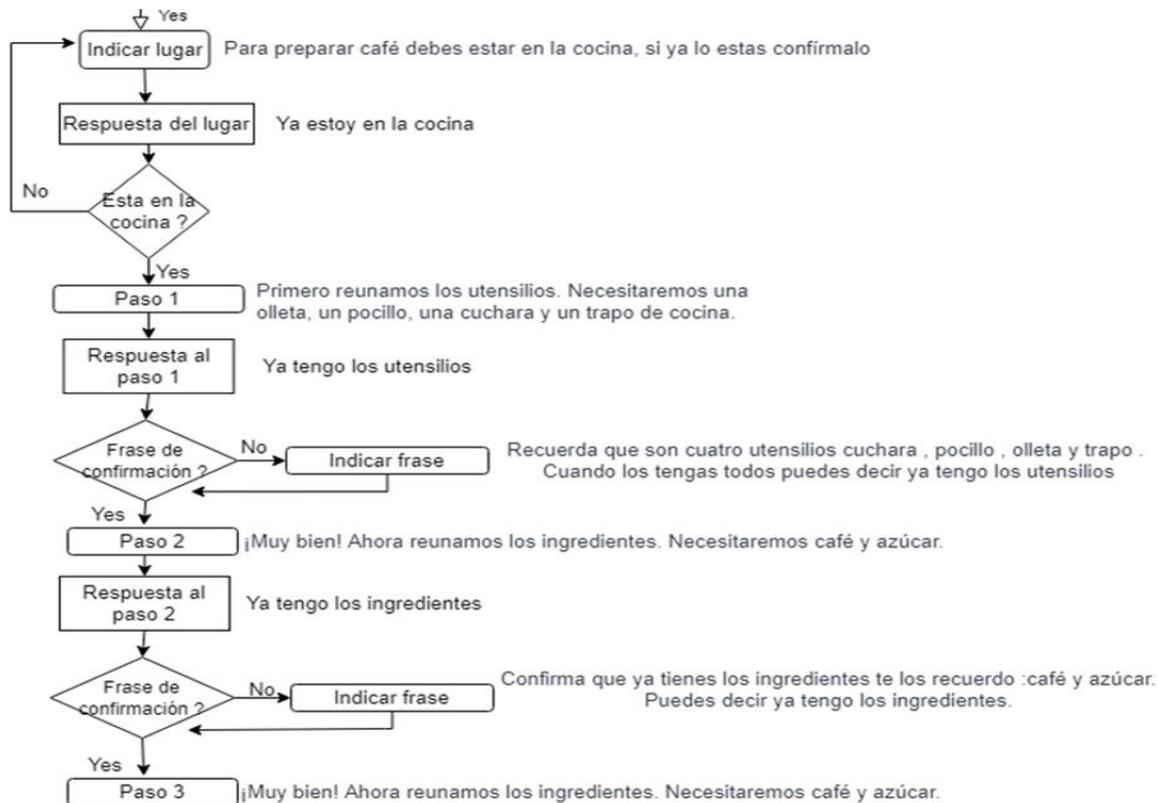


Figura 23. Diagrama de flujo que establecerá la lógica final que implementará el agente parte b.

```

pi@raspberrypi: ~/AIY-projects-python/src/examples
INFO:aiy.assistant.grpc:You said: "necesito hablar".
INFO:aiy.assistant.grpc:End of audio request detected.
INFO:aiy.assistant.grpc:You said: "medicamento".
INFO:aiy.assistant.grpc:You said: "Necesito hablar con".
INFO:aiy.assistant.grpc:You said: "Necesito hablar con la".
INFO:aiy.assistant.grpc:You said: "Necesito hablar con la vaca".
INFO:aiy.assistant.grpc:You said: "Necesito hablar con la doctora".
INFO:aiy.assistant.grpc:Recording stopped.
DEBUG:aiy.assistant.grpc:Updating conversation state.
INFO:aiy.assistant.grpc:Assistant said: "Vale. Voy a buscar la última versión de Doctora. ¡Hola! Soy doctora, estoy aquí para ayudarte. Solo pide ayuda cuando lo necesites."

```

Figura 24. Prueba de la tarea “preparar café” diseñada en Dialogflow utilizando el Voice kit parte a.

```

pi@raspberrypi: ~/AIY-projects-python/src/examples
INFO:aiy.assistant.grpc:You said: "Necesito ayuda".
INFO:aiy.assistant.grpc:You said: "Necesito ayuda".
INFO:aiy.assistant.grpc:Playing started.
DEBUG:aiy.assistant.grpc:Updating conversation state.
INFO:aiy.assistant.grpc:Assistant said: "¡Me encanta ayudarte! Puedo indicarte como preparar un café, ver la televisión y hasta decirte que hora es. ¿Qué necesitas ?"
INFO:aiy.assistant.grpc:Expecting follow-on query from user.

```

Figura 25. Prueba de la tarea “preparar café” diseñada en Dialogflow utilizando el Voice kit parte b.

```

pi@raspberrypi: ~/AIY-projects-python/src/examples
INFO:aiy.assistant.grpc:Recording started.
INFO:aiy.assistant.grpc:You said: "prende".
INFO:aiy.assistant.grpc:You said: "prepa".
INFO:aiy.assistant.grpc:You said: "preparar".
INFO:aiy.assistant.grpc:You said: "preparar un".
INFO:aiy.assistant.grpc:You said: "preparar un café".
INFO:aiy.assistant.grpc:You said: "preparar un café".
INFO:aiy.assistant.grpc:End of audio request detected.
INFO:aiy.assistant.grpc:You said: "preparar un café".
INFO:aiy.assistant.grpc:Recording stopped.
INFO:aiy.assistant.grpc:Playing started.
DEBUG:aiy.assistant.grpc:Updating conversation state.
INFO:aiy.assistant.grpc:Assistant said: "Te diré paso por paso como preparar café. En cada paso puedes decir palabras como ya o listo para continuar. ¿Estas listo ?"

```

Figura 26. Prueba de la tarea “preparar café” diseñada en Dialogflow utilizando el Voice kit parte c.

## 6. DISCUSIÓN

Los resultados obtenidos para nuestro modelo de detección de objetos son bastante cercanos a lo esperado comparándolo con el modelo entrenado por Dustin Franklin [34]. El modelo reconoce de manera adecuada y con un puntaje de precisión elevado los objetos para los que fue entrenado. Todo esto a pesar base de datos no era homogénea en la cantidad de imágenes que fueron descargadas por clase. esto genera que el entrenamiento por clase no sea igual para todas las clases ya que se sabe que entre más datos de entrenamiento y más épocas mejor resultará el modelo. [36] [34]



Figura 27. Prueba de distancia y orientación

Las imágenes anteriores corresponden a los mismos objetos, pero captados por la cámara a distintas distancias y con distintas orientaciones y cómo podemos observar influye en gran medida la clasificación de estos.



Figura 28. Clasificación errónea

También podemos observar la presencia de clasificaciones erróneas del modelo, aunque con un puntaje bajo que nos indicaría que nuestro modelo no tiene una certeza muy alta sobre la clase de objeto.

En cuanto al asistente de voz, en primera instancia al evaluar los resultados obtenidos en las pruebas de la experiencia conversacional diseñada, se puede apreciar que es difícil mantener la conversación debido a que las personas no conocen el flujo ni el contexto establecido. El principio de Pareto aplicado al flujo conversacional nos sugiere que, 80% de los usuarios usan 20% de los caminos conversacionales [39], por esto cuando se procedió a modificar la estructura de las frases en la cuales se incluyó la respuesta dentro de las preguntas realizadas se logró condicionar en cierta medida las posibles respuestas y mejoró notablemente el rumbo de la conversación.

Por otro lado, se pudo observar que al repetir la prueba varias veces los sujetos se adaptaron a la capacidad de entendimiento del agente y lograron identificar por sí mismas cómo debían responder para establecer una conversación continua, lo cual está relacionado con la capacitación para tareas específicas utilizada en la rehabilitación cognitiva para mejorar el desempeño de tareas funcionales por medio de la práctica dirigida a objetivos y la repetición. [3]

Con respecto al rendimiento del agente, se pudo observar que requiere ajustes para mejorar su rendimiento, una vez identificadas las falencias e implementado el diagrama final se logra reconocer adecuadamente las intenciones programadas, cabe destacar que el uso de los contextos de entrada y salida fueron indispensables al momento de indicar paso por paso la tarea especificada, ya que, aunque se detectara el Intent correspondiente de no ser incluidos el agente no seguía el orden esperado y proporcionaba los pasos al azar o saltaba algunos de ellos. En cuanto a los parámetros requeridos utilizados en las frases de entrenamiento, no es viable tratar de cubrir absolutamente todos los casos dentro de las frases, la manera más sencilla es agregando indicaciones que el agente proporcionara a sus usuarios sobre cuál es el parámetro que falta y que debería incluir en su respuesta. Por otra parte, la pronunciación de las frases se escucha fluida a pesar ser una voz artificial, se mantienen las pausas, acentos e incluso se aprecian los tonos específicos para hacer afirmaciones o preguntas.

## **7. RECOMENDACIONES Y TRABAJOS FUTUROS**

Para los trabajos futuros se busca como primera medida la integración del modelo de detección de objetos con el asistente de voz, de tal manera que permita ofrecer una ayuda más completa a las personas con déficits cognitivos leves y graves brindando asistencia a la persona en la orientación o lugar de la casa donde se encuentra ubicado, también verificar tiene los objetos indicados para realizar la actividad. Por ejemplo: las personas con déficits cognitivos pueden presentar dificultades para seleccionar los objetos requeridos para las tareas, mantenerse en la actividad inhibiendo los estímulos externos, y seguir el objetivo hasta su finalización. Este tipo de tecnología merece ser explorado en contextos reales y verificar su usabilidad.

Se busca que la herramienta sea compacta y portátil, que permita al usuario final llevarla a todas partes sin que le genere incomodidad, para esto debe contar con una cámara web, una batería adecuada y con el sistema de audio vendido Nvidia para la Jetson Nano, todo esto ensamblado dentro de un recipiente que nos permita transportar todo con facilidad.

Este trabajo es una prueba de cómo se comportan estas tecnologías para una sola actividad de la vida diaria en la cocina, pero para trabajos futuros, una vez integrada la tecnología del asistente virtual con la visión por computadora, podemos implementar otras actividades de entretenimiento o aseo personal.

## 8. CONCLUSIONES

En este trabajo se propone La construcción de herramientas de bajo costo para uso de profesionales de terapia ocupacional en la implementación de principios de la rehabilitación cognitiva para lograr una cognición funcional. se requiere del trabajo en conjunto de personal de ingeniería y profesionales de la salud para lograr los mejores resultados posibles. Para la implementación del asistente de voz es indispensable realizar previamente un trabajo de diseño conversacional en el que se debe enfatizar en varios aspectos clave, primeramente, debemos evitar sobrediseñar los diálogos iniciales ya que no es eficiente tratar de cubrir todos los casos que puedan presentarse, es mejor invertir en mejorar los flujos que tendrán mayor impacto para ayudar en la tarea inicialmente planteada. Probar la experiencia conversacional con personas que no están familiarizadas con el diseño nos permite identificar problemas rápidamente, esta realimentación es una de las partes más importantes, ya que nos ayudó a reconocer puntos que un principio no se tomaron en cuenta, identificando que lograr un flujo adecuado es más fácil si toda la información está contenida en los diálogos que en un principio eran cortos y concisos.

Al evaluar la tecnología utilizada podemos apreciar que la posibilidad que ofrece Dialogflow de proporcionar distintas respuestas ante una misma solicitud nos permite tener una experiencia más natural sin caer en el seguimiento de un flujo establecido a manera de único guion. Sin embargo, el uso de entidades que requieran una estructura muy específica tiene limitaciones en plataformas como esta que tienen modelos de reconocimiento de voz pre entrenados, esta tarea requiere de un nivel de experiencia mucho mayor en técnicas de ML para la síntesis de audio que se escapa del alcance del proyecto.

Reconocer las AVDs por medio de voz es una tarea sencilla ya que basta con una simple pregunta para identificar lugares y acciones realizadas, en cambio, proporcionar asistencia paso por paso de una actividad es complejo ya que es fácil desviarse del flujo establecido, por esto es necesario trabajar de la mano de profesionales de la salud para caracterizar el flujo de conversación según la discapacidad, la edad del usuario, o cualquier otra variable que pueda afectar el flujo.

Es innegable que en un futuro cercano necesitaremos de todas las herramientas que la inteligencia artificial nos brinda, ya sea desde autos que se conducen solos, hasta asistentes en el hogar que estén pendientes las 24 horas del día de las actividades que realizamos. Todos buscamos una constante optimización de las tareas que realizamos y la inteligencia artificial es la clave para lograr esto. El proyecto planteado en este documento tiene un gran potencial para ayudar a las personas con alteraciones cognitivas. Pero se tiene un gran reto por delante al querer integrar la detección de objetos con el asistente de voz, reto del que necesitamos ayuda de los profesionales que tratan todos los días con estos pacientes, para que nos guíen por el mejor camino. Nvidia y otras marcas como Google están ayudando en gran medida al avance y crecimiento de la inteligencia artificial, por medio del desarrollo de soluciones en tiempo real y de bajo costo, En nuestro proyecto validamos el uso de Google Coral, Raspberry PI y Jetson Nano para la implementación de herramientas que hagan uso de modelos de inteligencia artificial.

## Bibliografía

- [1] A. González, «Deterioro cognitivo y calidad de vida en ancianos de una clínica de medicina familiar de la ciudad,» archivos en medicina familiar, ciudad de mexico, 2008.
- [2] P. Noomi Katz, Introduction to Cognition and Participation, Kiryat Ono, Israel: EBSCOhost, 2008.
- [3] Y. Goverover, «Cognitive Rehabilitation: Evidence-Based Interventions,» EBSCO, 2018, p. 51–66.
- [4] E. Guerrero, J. C. Nieves y H. Lindg, «ALI: An assisted living system for persons with mild cognitive impairment,» CBMS, 2013.
- [5] A. M. Seelye, M. Schmitter, B. Das y D. J. Cook, pplication of Cognitive Rehabilitation Theory to the Development of Smart Prompting Technologies,, IEEE Reviews in biomedical engineering, 2012.
- [6] MFMER, «Deterioro cognitivo leve,» MAYO CLINIC, 2020. [En línea]. Available: <https://www.mayoclinic.org/es-es/diseases-conditions/mild-cognitive-impairment/symptoms-causes/syc-20354578>. [Último acceso: 2022].
- [7] H. a. R. D. Pirsiavash, «Detecting activities of daily living in first-person camera views,» de 2012 *IEEE conference on computer vision and pattern recognition*, 2012, pp. 2847-2854.
- [8] K. Cicerone, C. Dahlberg, J. Malec y D. Langenbahn, «Evidence-based cognitive rehabilitation: updated review of the literature,» Archives of physical medicine and, 2005, p. 1681–1692.
- [9] S. Campbell, P. Halder, S. Campbell y D. Phillips, «Cross Cultural Differences in Cognition: A Study on How Culture Affects the Way We Think,» SIASAT, 2020, pp. 9-15.
- [10] L. Boman, K. Tham, A. Granqvist, A. Bartfai y H. Hemmingsson, «Using electronic aids to daily living after acquired brain injury: a study of the learning process and the usability,» Disability and Rehabilitation: Assistive Technology, 2007, pp. 23-33.
- [11] H. Thomas, «(IA: Inteligencia Artificial).,» *Polis*, vol. 1, nº 2, p. 0, 2001.
- [12] J. Negrete, «Inteligencia artificial y la demostración analógica: Estrategia, soluciones y trascendencia educativa.,» *Revista Mexicana De Biodiversidad*, vol. 15, nº 1, pp. 89-103, 2011.
- [13] W. Rahman, AI and Machine Learning, New Delhi, India : Sage Publications Pvt. Ltd. , 2020.
- [14] thoth, «Qué es el Deep Learning (Aprendizaje Profundo),» 2017. [En línea]. Available: <http://formatalent.com/que-es-el-deep-learning-aprendizaje-profundo>. [Último acceso: 8 5 2022].
- [15] M. Balastegui, «Inteligencia Artificial: cómo crear tu propio chatbot,» 2019. [En línea]. Available: <https://djc.es/d3X01>. [Último acceso: 8 5 2022].
- [16] J. Aleman, M. Ruiz y E. Aguilar, «Retrospectiva al Desarrollo de Chatbots y Procesamiento del Lenguaje Natural,» 2019. [En línea]. Available: <http://redicces.org.sv/jspui/handle/10972/3782>. [Último acceso: 8 5 2022].
- [17] A. I. Harabagiu, «Desarrollo de chatbots en la plataforma de Facebook,» , 2017. [En línea]. Available: <http://repositori.uji.es/xmlui/handle/10234/174196>. [Último acceso: 8 5 2022].
- [18] J. Eisenstein, Introduction to natural language processing., London: The MIT Press, 2019.

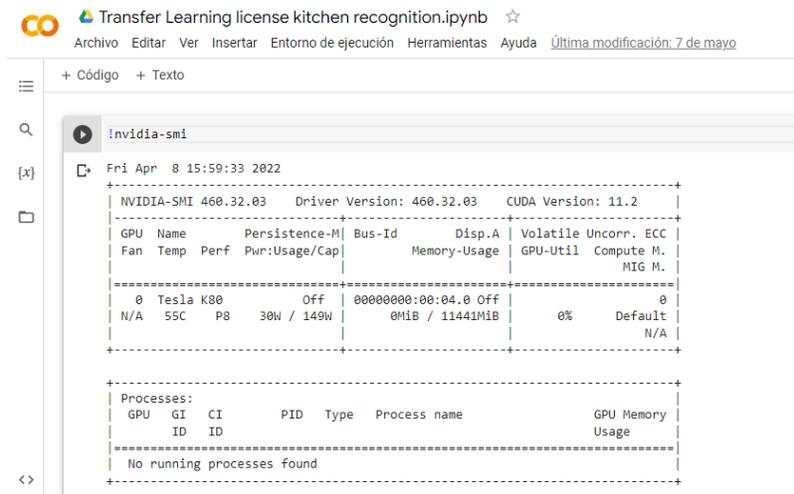
- [19] A. Chopra, A. Prashar y C. Sain, «Natural Language Processing,» *international journal of technology enhancements and emerging engineering research*, vol. I, nº 4, pp. 1-3, 2013.
- [20] G. Marcus, F. Rossi y M. Veloso, «Beyond the Turing Test,» *Association for the Advancement of Artificial Intelligence*, 2016.
- [21] M. Ballesteros y S. Cózar, *Automatización de respuestas a las dudas de puntuación más frecuentes dirigidas a la RAE en Twitter. Propuesta de un asistente conversacional.*, Madrid, 2021.
- [22] M. Vidal., «Voicebots: la atención al cliente de la era millennial,» *Contact Center Call Center & IP solutions*, nº 90, pp. 21-64, 2018.
- [23] Microsoft, «Microsoft Bot Framework,» 2019. [En línea]. Available: <https://dev.botframework.com/>. [Último acceso: Marzo 2022].
- [24] Google, «Conceptos básicos de Dialogflow ES,» 16 12 2021. [En línea]. Available: <https://cloud.google.com/dialogflow/es/docs/basics>.
- [25] Platzi, «Curso de DialogFlow: Programación de Chatbots por voz,» [En línea]. Available: <https://platzi.com/cursos/dialogflow/>.
- [26] Amazon, «Amazon Lex : Crear chatbots con IA conversacional,» Amazon Web Services, 2022. [En línea]. Available: <https://aws.amazon.com/es/lex/>. [Último acceso: 2022].
- [27] F. Rueda, «¿Qué es la computación,» *Revista Sistemas*, vol. 112, pp. 72-80, 2009.
- [28] Maplink, «Descubre qué es Google Cloud Platform y sus ventajas,» 2021. [En línea]. Available: <https://maplink.global/blog/es/que-es-google-cloud/>.
- [29] Amazon, «Amazon Web Services,» 2022, Amazon Web Services, 2022. [En línea]. Available: <https://djc.es/s1f1c>. [Último acceso: 2022].
- [30] Microsoft, «¿Qué es Azure?,» Microsoft 2022, 2021. [En línea]. Available: <https://djc.es/WoeYh>. [Último acceso: 2022].
- [31] Google, «Voice Kit,» 2022. [En línea]. Available: <https://aiyprojects.withgoogle.com/voice/>.
- [32] A. Vera, «Curso Profesional de Computer Vision con TensorFlow,» Platzi, 1 01 2022. [En línea]. Available: <https://platzi.com/cursos/computer-vision-tensorflow/>. [Último acceso: 08 04 2022].
- [33] E. Robotics, «8 aplicaciones de la visión artificial,» 2020. [En línea]. Available: <https://www.edsrobotics.com/blog/ocho-aplicaciones-vision-artificial/>. [Último acceso: 2022].
- [34] D. Franklin, «dusty-nv,» Nvidia, 18 junio 2019. [En línea]. Available: <https://github.com/dusty-nv/jetson-inference>. [Último acceso: 15 03 2022].
- [35] R. Gandhi, «R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms,» *Towards Data Science*, 2018. [En línea]. Available: <https://djc.es/qzSga>. [Último acceso: 2022].
- [36] A. Bochkovskiy, C. Yao y H. Yuan, «YOLOv4: Optimal Speed and Accuracy of Object Detection,» *Academia Sinica, Taiwan*, 2020.
- [37] «NVIDIA JETSON NANO,» NVIDIA Corporation, 2022. [En línea]. Available: <https://djc.es/hpaeF>. [Último acceso: 15 04 2022].

- [38] G. I. o. Technology, «Open Images Dataset V6,» Facebook AI, 2018. [En línea]. Available: <https://storage.googleapis.com/openimages/web/extras.html>. [Último acceso: 2022].
- [39] J. González, «Ley de pareto 80/20,» *Departamento de Economía, Contabilidad y Finanzas.*, 2007.
- [40] particleman14, «Transfer Learning for Edge Devices,» GitHub, Inc., 2021. [En línea]. [Último acceso: 2022].
- [41] M. Glover, «Cuáles son los procesos cognitivos básicos,» *Psicología-Online*, 2022. [En línea]. [Último acceso: 2022].
- [42] C. G. Katzelnick, *Orthostatic Blood Pressure and Arterial Stiffness in Persons with Spinal Cord Injury: The Effect of the Renin Angiotensin Aldosterone System*, 2020.
- [43] M. B. Hernández y J. M. Gómez, «Aplicaciones de procesamiento de lenguaje natural,» 2013. [En línea]. Available: <https://djc.es/wmCxp>. [Último acceso: 8 5 2022].
- [44] L. Galvis, H. Arguello y D. Mateus, «Tratamiento digital de imágenes e inteligencia artificial aplicados a la perforación de pozos petroleros,» *Revista Fuentes*, vol. 9, n° 1, p. 3, 2011.
- [45] Biblioteca Nacional de medicina,, «Deterioro cognitivo leve,» *MedlinePlus*, 2021. [En línea]. Available: <https://medlineplus.gov/spanish>. [Último acceso: 2022].
- [46] K. Dinerman, A. a. Josman, N. a. Roe y David, «The use of cognitive strategies among people with schizophrenia: a randomized comparative study,» *The Open Journal of Occupational Therapy*, 2019, pp. 1--12.
- [47] M. Teresa, S. Alicia y M. Juan, «Estrategias de compensación en adultos mayores,» *scielo*, Murcia, 2015.
- [48] P. O. Yael Goverover, *Cognitive Rehabilitation: Evidence-Based Interventions*, israel: EBSCO, 2018.
- [49] J. Pineau, M. Montemerlo, M. Pollack, N. Roy y S. Thrun, «Towards robotic assistants in nursing homes: Challenges and results,» *Information Technol. Biomedicine*, 2003.
- [50] S. Fickas, M. Sohlberg y P. Hung, *Route-following assistance for travelers with cognitive impairments: A comparison of four prompt modes*, *International Journal of Human-Computer Studies*, 2008.

## 9. ANEXOS

### 9.1. GOOGLE COLAB

El entrenamiento del modelo utilizado para este proyecto se realizó en la nube con GPU discretas, específicamente en colab debido a los grandes conjuntos de datos que se utilizan. A continuación, indicaremos un link al código usado en nuestro modelo un archivo con el nombre Transfer Learning license kitchen recognition.ipynb.



```
Transfer Learning license kitchen recognition.ipynb ☆
Archivo Editar Ver Insertar Entorno de ejecución Herramientas Ayuda Última modificación: 7 de mayo

+ Código + Texto

!nvidia-smi

Fri Apr 8 15:59:33 2022
+-----+
| NVIDIA-SMI 460.32.03      Driver Version: 460.32.03      CUDA Version: 11.2      |
+-----+-----+-----+
| GPU   Name      Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|  Memory-Usage | GPU-Util  Compute M. |
|                                           MIG M.         |
+-----+-----+-----+
|  0   Tesla K80           Off      | 00000000:00:04:0 Off |    0          0     |
| N/A   55C    P8     30W / 149W |  0MiB / 11441MiB |      0%      Default  |
+-----+-----+-----+
|                                           |
+-----+-----+-----+
| Processes:                                                       GPU Memory |
|  GPU   GI    CI        PID   Type   Process name                  Usage    |
|-----+-----+-----+
| No running processes found                                     |
+-----+-----+-----+
```

## 9.2. GITHUB JETSON INFERENCE NVIDIA

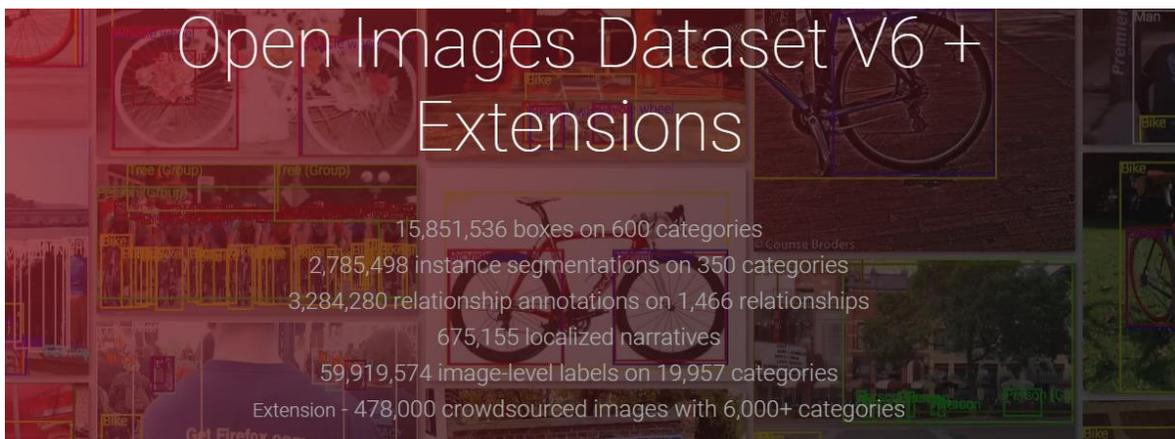
El siguiente link nos conduce al github hecho por desarrolladores de nvidia, en el repositorio se proporciona una biblioteca de redes de aprendizaje profundo aceleradas por TensorRT para reconocimiento de imágenes, detección de objetos con localización (es decir, cuadros delimitadores) y segmentación semántica.

Es una guía rápida al mundo de la visión computarizada de la mano de expertos en el tema con pasos bien explicados y videos que ayudaran ante cualquier duda, además se esto se actualiza constantemente.



## 9.3. Open Images dataset v6

Open Images dataset v6 es un conjunto de datos de 9 millones de imágenes anotadas con etiquetas de nivel de imagen, cuadros delimitadores de objetos, máscaras de segmentación de objetos, relaciones visuales y narraciones localizadas.

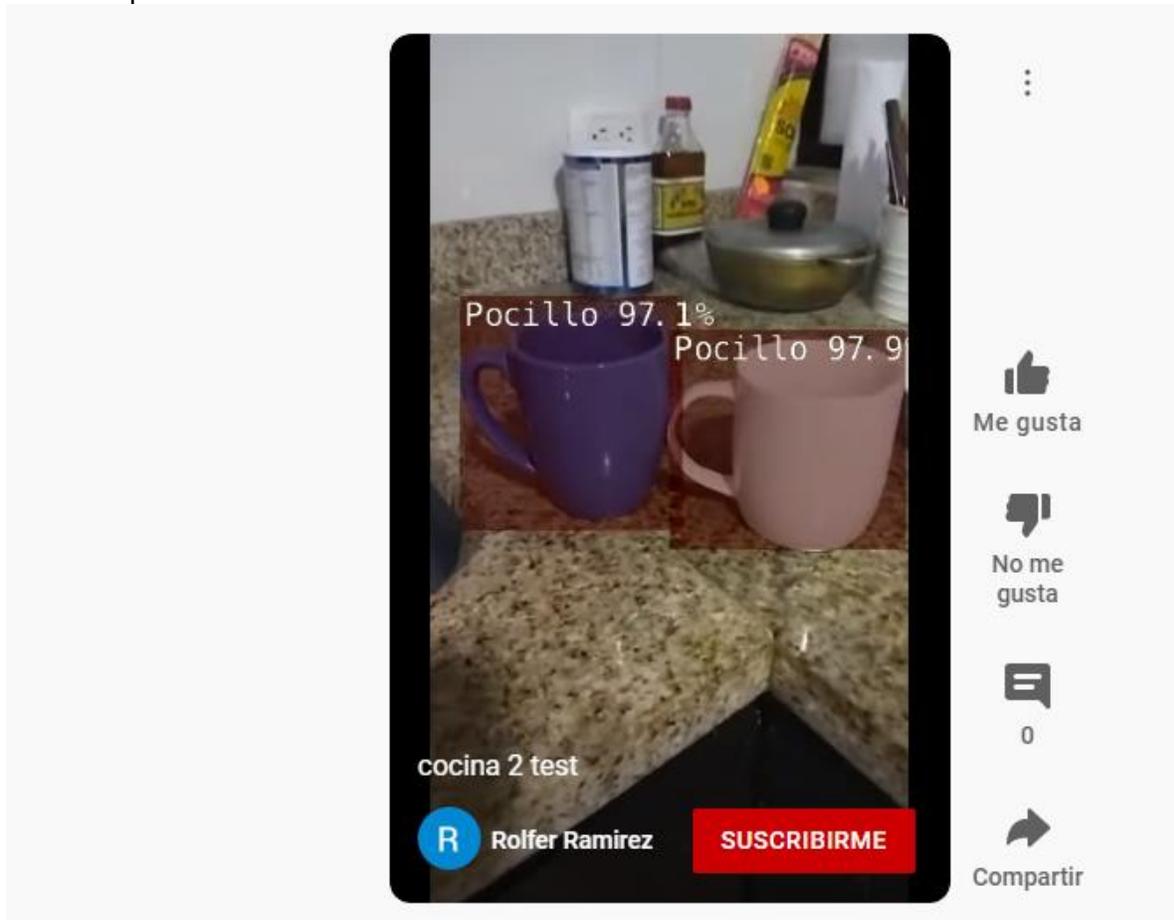


Es Un conjunto de datos públicos para la clasificación de imágenes a gran escala con múltiples etiquetas y clases, creado en 2017 por Krasin I., Duerig T., Alldrin N., Ferrari V., Abu-El-Haija S., Kuznetsova A., Rom H., Uijlings J., Popov S., Kamali S., Mallocci M., Pont-Tuset J., Veit A., Belongie S., Gomes V., Gupta A., Sun C., Chechik G., Cai D., Feng Z., Narayanan D., Murphy K.

#### 9.4. VIDEOS DE PRUEBA

Los siguientes enlaces son videos de prueba realizados por nosotros en los que corremos nuestro modelo de detección de objetos, estos videos fueron subidos a you tube para tener un fácil acceso a ellos.

Video de prueba del modelo:



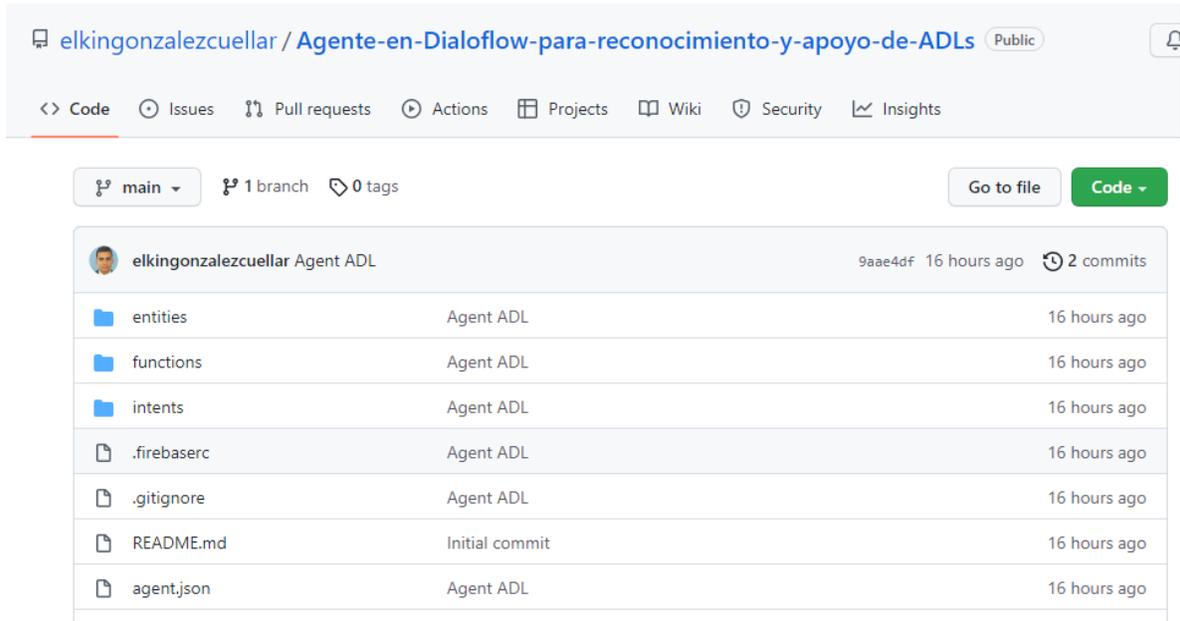
Links de acceso:

<https://youtube.com/shorts/2uDafCCWGVY?feature=share>

<https://youtube.com/shorts/jyhQYD7B9kU?feature=share>

## 9.5. Repositorio asistente de voz

Este enlace nos lleva a nuestro repositorio donde tenemos todo nuestro trabajo realizado en nuestro asistente virtual,



The screenshot shows a GitHub repository page. At the top, the repository name is 'elkingonzalezcuellar / Agente-en-Dialoflow-para-reconocimiento-y-apoyo-de-ADLs' with a 'Public' label. Below the repository name are navigation links: '<> Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', and 'Insights'. The repository is currently on the 'main' branch, with '1 branch' and '0 tags' indicated. There are two buttons: 'Go to file' and 'Code'. The repository content is displayed as a table of files and folders:

File/Folder	Commit Message	Commit Time
elkingonzalezcuellar Agent ADL	9aae4df	16 hours ago
entities	Agent ADL	16 hours ago
functions	Agent ADL	16 hours ago
intents	Agent ADL	16 hours ago
.firebaserc	Agent ADL	16 hours ago
.gitignore	Agent ADL	16 hours ago
README.md	Initial commit	16 hours ago
agent.json	Agent ADL	16 hours ago