

Asistente virtual: aprendizaje automático supervisado - IA

Grado 8°

Guía 5



TIC



Estudiantes

Apoya:



Asistente virtual: aprendizaje automático supervisado - IA

Grado 8°

Guía 5



TIC



Estudiantes



Educación



</Colombia
Programa>
{EL CÓDIGO A TU FUTURO}

**MINISTERIO DE TECNOLOGÍAS
DE LA INFORMACIÓN Y LAS
COMUNICACIONES**

Julián Molina Gómez
Ministro TIC

Luis Eduardo Aguiar Delgadillo
Viceministro (e) de Conectividad

Yeimi Carina Murcia Yela
Viceministra de Transformación Digital

Óscar Alexander Ballen Cifuentes
Director (e) de Apropiación de TIC

Alejandro Guzmán
Jefe de la Oficina Asesora de Prensa

Equipo Técnico
Lady Diana Mojica Bautista
Cristhiam Fernando Jácome Jiménez
Ricardo Cañón Moreno

Consultora experta
Heidy Esperanza Gordillo Bogota

BRITISH COUNCIL

Felipe Villar Stein
Director de país

Laura Barragán Montaña
**Directora de programas de Educación,
Inglés y Artes**

Marianella Ortiz Montes
Jefe de Colegios

David Vallejo Acuña
**Jefe de Implementación
Colombia Programa**

Equipo operativo
Juanita Camila Ruiz Díaz
Bárbara De Castro Nieto
Alexandra Ruiz Correa
Dayra Maritza Paz Calderón
Saúl F. Torres
Óscar Daniel Barrios Díaz
César Augusto Herrera Lozano
Paula Álvarez Peña

Equipo técnico
Alejandro Espinal Duque
Ana Lorena Molina Castro
Vanessa Abad Rendón
Raisa Marcela Ortiz Cardona
Juan Camilo Londoño Estrada

Edición y coautoría versiones finales
Alejandro Espinal Duque
Ana Lorena Molina Castro
Vanessa Abad Rendón
Raisa Marcela Ortiz Cardona

Edición
Juanita Camila Ruiz Díaz
Alexandra Ruiz Correa

**British Computer Society –
Consultoría internacional**

Niel McLean
Jefe de Educación

Julia Adamson
Directora Ejecutiva de Educación

Claire Williams
Coordinadora de Alianzas

**Asociación de facultades de
ingeniería - ACOFI**

Edición general
Mauricio Duque Escobar

Coordinación pedagógica
Margarita Gómez Sarmiento
Mariana Arboleda Flórez
Rafael Amador Rodríguez

Coordinación de producción
Harry Luque Camargo

Asesoría estrategia equidad
Paola González Valcárcel

Asesoría primera infancia
Juana Carrizosa Umaña

Autoría
Arlet Orozco Marbello
Harry Luque Camargo
Isabella Estrada Reyes
Lucio Chávez Mariño
Margarita Gómez Sarmiento
Mariana Arboleda Flórez
Mauricio Duque Escobar
Paola González Valcárcel
Rafael Amador Rodríguez
Rocío Cardona Gómez
Saray Piñerez Zambrano
Yimzay Molina Ramos

PUNTOAPARTE EDITORES

Diseño, diagramación, ilustración,
y revisión de estilo

Impreso por Panamericana Formas e
Impresos S.A., Colombia

Material producido para Colombia
Programa, en el marco del convenio
1247 de 2023 entre el Ministerio de
Tecnologías de la Información y las
Comunicaciones y el British Council

Esta obra se encuentra bajo una
Licencia Creative Commons
Atribución-No Comercial
4.0 Internacional. [https://
creativecommons.org/licenses/
by-nc/4.0/](https://creativecommons.org/licenses/by-nc/4.0/)

 **CC BY-NC 4.0**

“Esta guía corresponde a una
versión preliminar en proceso
de revisión y ajuste. La versión
final actualizada estará
disponible en formato digital
y puede incluir modificaciones
respecto a esta edición”

Prólogo

Estimados educadores, estudiantes y comunidad educativa:

En el Ministerio de Tecnologías de la Información y las Comunicaciones, creemos que la tecnología es una herramienta poderosa para incluir y transformar, mejorando la vida de todos los colombianos. Nos guía una visión de tecnología al servicio de la humanidad, ubicando siempre a las personas en el centro de la educación técnica.

Sabemos que no habrá progreso real si no garantizamos que los avances tecnológicos beneficien a todos, sin dejar a nadie atrás. Por eso, nos hemos propuesto una meta ambiciosa: formar a un millón de personas en habilidades que les permitan no solo adaptarse al futuro, sino construirlo con sus propias manos. Hoy damos un paso fundamental hacia este objetivo con la presentación de las guías de pensamiento computacional, un recurso diseñado para llevar a las aulas herramientas que fomenten la creatividad, el pensamiento crítico y la resolución de problemas.

Estas guías no son solo materiales educativos; son una invitación a imaginar, cuestionar y crear. En un mundo cada vez más impulsado por la inteligencia artificial, desarrollar habilidades como el pensamiento computacional se convierte en la base, en el primer acercamiento para que las y los ciudadanos aprendan a programar y solucionar problemas de forma lógica y estructurada.

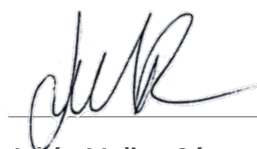
Estas guías han sido diseñadas pensando en cada región del país, con actividades accesibles que se adaptan a diferentes contextos, incluyendo aquellos con limitaciones tecnológicas. Esta es una apuesta por la equidad, por cerrar las brechas y asegurar que nadie se quede atrás en la revolución digital. Quiero destacar, además, que son el resultado de un esfuerzo colectivo:

más de 2.000 docentes colaboraron en su elaboración, compartiendo sus ideas y experiencias para que este material realmente se ajuste a las necesidades de nuestras aulas. Además, con el apoyo del British Council y su red de expertos internacionales, hemos integrado prácticas globales de excelencia adaptadas a nuestra realidad nacional.

Hoy presentamos un recurso innovador y de alta calidad, diseñado en línea con las orientaciones curriculares del Ministerio de Educación Nacional. Cada página de estas guías invita a transformar las aulas en espacios participativos, creativos y, sobre todo, en ambientes donde las y los estudiantes puedan desafiar estereotipos y explorar nuevas formas de pensar.

Trabajemos juntos para garantizar que cada estudiante, sin importar dónde se encuentre, tenga acceso a las herramientas necesarias para imaginar y construir un futuro en el que todos seamos protagonistas del cambio. Porque la tecnología debe ser un instrumento de justicia social, y estamos comprometidos a que las herramientas digitales ayuden a cerrar brechas sociales y económicas, garantizando oportunidades para todos.

Con estas guías, reafirmamos nuestro compromiso con la democratización de las tecnologías y el desarrollo rural, porque creemos en el potencial de cada región y en la capacidad de nuestras comunidades para liderar el cambio.



Julián Molina Gómez
Ministro de Tecnologías de la
Información y las Comunicaciones
Gobierno de Colombia



Guía de íconos



Equidad en el acceso
y la participación en el
mundo digital



Lógica, programación
y depuración



Prácticas de datos



Inteligencia artificial

Aprendizajes
de la guía

Con las actividades de
esta guía se espera que
puedas avanzar en:



Explicar las
diferencias entre
la programación
tradicional y la de
la IA.



Explicar un proceso
de desarrollo de una
aplicación de IA.



Reconocer la
computación como
un campo que se
ha beneficiado de
la diversidad y la
colaboración.

Resumen de la guía

Esta guía propone 5 sesiones orientadas a aprender a crear un asistente virtual en *Scratch*¹. Se comienza implementando con programación tradicional para ver sus limitaciones. Luego se procede a entrenar y validar un modelo de Inteligencia Artificial (IA) de reconocimiento de texto por medio de aprendizaje automático supervisado. Finalmente, el modelo de IA se implementa en *Scratch* incluyendo umbrales de acción.

Resumen de las sesiones

Sesión 1

En esta sesión se presentan los asistentes virtuales y se comienza a programar un asistente virtual en *Scratch* usando programación tradicional.

Sesión 2

En esta sesión se mejora el asistente virtual permitiéndole reconocer algunas variaciones de las instrucciones que conoce por medio de listas. Se continúa usando programación tradicional.

1. Adaptado Arboleda, M., Vieira, C., & Chiu, J. L. (2023). Opening the machine learning black box for multidisciplinary students: Scaffolding from GUI to coding. *IEEE Frontiers in Education Conference (FIE)*, 1-5. <https://doi.org/10.1109/FIE58773.2023.10343043>

Archivos para descargar



Archivos de esta sesión:
Usando este código QR se podrán descargar los archivos que se utilizarán en esta guía.

Sesión 3

En esta sesión se identifican patrones como introducción al aprendizaje automático. Luego se entrena un modelo por medio de aprendizaje automático supervisado de reconocimiento de texto para el asistente virtual.

Sesión 4

En esta sesión se abordan componentes de una IA con aprendizaje automático, al igual que etapas de desarrollo de un modelo de IA. La clase entrena modelos y los valida con IA colaborativa, reconociendo la importancia de la participación de personas diferentes en el desarrollo.

Sesión 5

En esta sesión se presentan la confianza del modelo y los umbrales de acción. Las y los estudiantes los agregan a su modelo de IA para finalmente implementarlo en un asistente virtual en *Scratch*.



Conexión con otras áreas

La familiarización con los asistentes digitales es un componente importante en la educación en tecnología en la actualidad. Estos dispositivos son una de las aplicaciones de IA de uso creciente disponible al público. Además, las actividades propuestas permiten fortalecer aprendizajes específicos articulados a otras áreas, entre los que se destacan:

Lenguaje

- Participar en espacios de discusión grupal, teniendo en cuenta la temática y la intencionalidad. Componer diferentes tipos de texto atendiendo a las características de sus ámbitos de uso.



Conexión con otras áreas

Ética y valores

- Aceptar y valorar la diversidad, mostrando respeto por opiniones que difieren de la propia. Considerar el impacto de las acciones en el entorno.

Matemáticas

- Reconocer patrones numéricos y no numéricos y describirlos verbalmente.

Ciencias sociales

- Comprender las razones de algunos cambios socioculturales motivados por el uso de la tecnología.



Sesión

1

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Explicar usos de la IA en la vida cotidiana.



Identificar los límites de la programación tradicional en cuanto a la comprensión del lenguaje natural.

Duración sugerida



15%

70%

15%

Material para la clase

- Anexo 1.1 (conservar durante toda la guía)
- Anexo 1.2.
- Dispositivo con acceso a Scratch y archivos previamente descargados del enlace suministrado en la introducción de la guía.



Anexos

Anexo 1.2



Los asistentes virtuales son aplicaciones de IA que pueden ayudar con la automatización de diversas tareas e incluso responder a solicitudes específicas de sus usuarios, permitiéndoles así a lo que tiene un(a) asistente personal. Alexa, Siri y el asistente de Google son algunos ejemplos de asistentes virtuales que se han hecho populares hoy en día.

Tu reto consiste en crear un proyecto en Scratch que genere un prototipo de asistente virtual que cumpla con las siguientes especificaciones funcionales:

- 1 Debe realizar cuatro acciones en el escenario de Scratch:
 - a. Prender un radio.
 - b. Apagar un radio.
 - c. Prender una lámpara.
 - d. Apagar una lámpara.
- 2 Debe recibir las instrucciones en texto.
- 3 Debe entender lenguaje natural.
- 4 No debe actuar si no está seguro de lo que le solicitan, o si considera que los efectos de un error son indeseables.

Para lograr lo anterior necesitas:

- 1 Entrenar un modelo de IA en <https://machinelearningforkids.com/>
- 2 Usar variables de acción.
- 3 Validar tu modelo de IA.
- 4 Implementar tu modelo de IA en un proyecto en Scratch.

Lo que sabemos,

lo que debemos saber



Esta sección corresponde al 15% de avance de la sesión

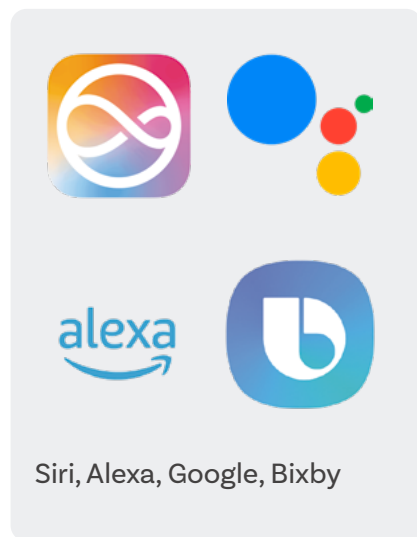
En esta guía aprenderás a crear un prototipo de **asistente virtual** que haga uso de un modelo de inteligencia artificial. Puedes leer más sobre este reto en el Anexo 1.2. Aunque solo tendrás que abordar la resolución de esta tarea al llegar a la sesión 5, a lo largo de las sesiones previas irás recordando, aprendiendo o reforzando lo que necesitas para darle solución.

Empieza reflexionando sobre estas preguntas:



¿Qué sabes sobre inteligencia artificial (IA)?
¿Ya has utilizado la IA?

Figura 1. Logos de asistentes virtuales



Siri, Alexa, Google, Bixby

Prepárate para compartir tus respuestas.

La IA es un sistema que realiza predicciones o toma decisiones a partir de información. Es un conjunto de sistemas que realizan tareas complejas para las que consideramos se requiere inteligencia. No tiene una forma específica, como la de un robot. De hecho, está más estrechamente relacionada con programas de computador.

Como sabes, la IA ya está entre nosotras y nosotros y cada vez será más probable que la uses, aun sin darte cuenta. Por ejemplo, si has utilizado un buscador de internet como Google o Bing, has usado IA. Si has escrito en un celular inteligente con predictor de texto, has usado IA. Si has visto videos o has escuchado música en internet de plataformas como YouTube, Netflix o Spotify, has usado IA.

Otro ejemplo de IA son los asistentes virtuales. ¿Qué sabes al respecto? En la Figura 1 encontrarás los logos de cuatro de ellos.

Si se requiere

- En la Guía 2 de grado 6 se abordan los aprendizajes iniciales sobre el uso de Scratch.



¿Reconoces alguno? ¿Para qué sirven?

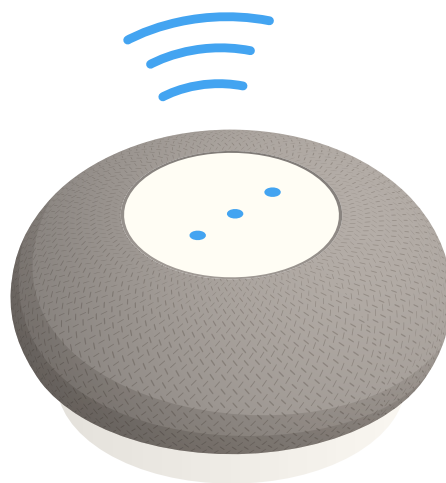
Los **asistentes virtuales** son programas que pueden realizar tareas diversas por medio de una comunicación natural con sus usuarios. Por ejemplo, pueden administrar su correo electrónico, mensajes de texto y llamadas e incluso controlar dispositivos como un televisor inteligente. Otra característica que hace que la comunicación de las y los usuarios con estos asistentes sea más fluida es que son capaces de escuchar y de hablar, como si se estuviera conversando con una persona. Las últimas versiones de estos asistentes entienden el lenguaje natural, por lo que no requieren el uso de términos especiales, como sí sucede cuando se programa.

Pregunta a tu docente si puede mostrarte un video o hacerte una demostración del funcionamiento de alguno de estos asistentes virtuales: Siri de Apple, el asistente de Google o Alexa de Amazon.

Glosario



Asistente virtual: un programa que puede realizar tareas diversas por medio de una comunicación natural con el usuario.



Nota

Scratch puede instalarse en un computador para funcionar sin conexión a internet. También se puede usar desde celulares y tabletas con internet, e incluso permite integración con tecnologías asistivas como lectores en pantalla. Pide apoyo a tu docente si necesitas hacer uso de alguna de estas opciones.

Recuerda que al comienzo de la guía se encuentra el código QR y el enlace para descargar los archivos requeridos.

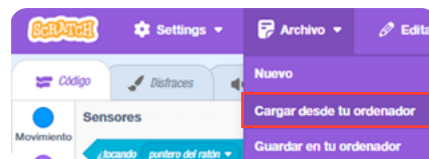
Manos a la obra**Conectadas**

Esta sección corresponde al 85% de avance de la sesión

Ahora vas a crear una primera versión del prototipo de asistente virtual en Scratch que se solicita en el reto. En este caso lograrás que controle dos dispositivos dispuestos en el escenario. Organízate en grupos de 2 a 3, de acuerdo con las indicaciones de tu docente.

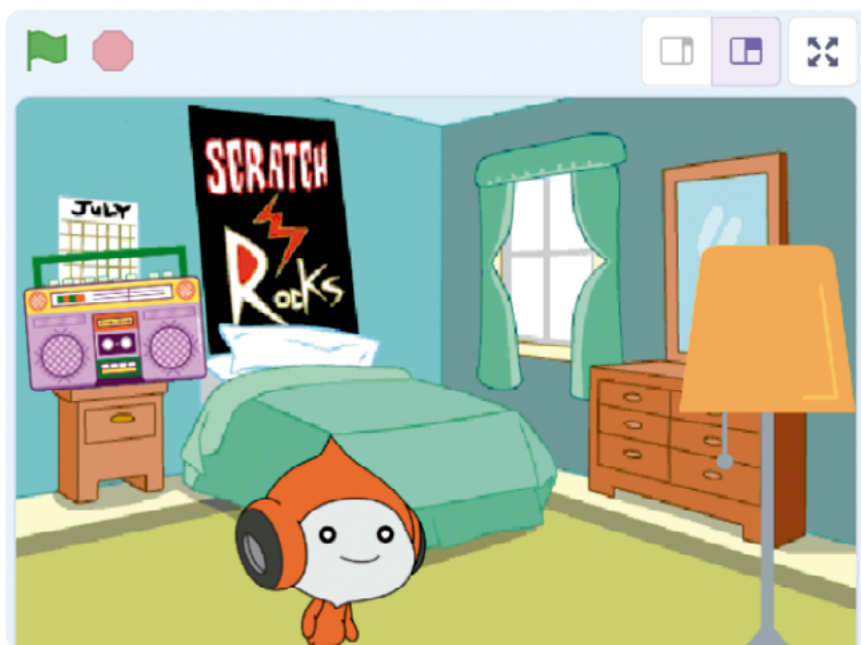
Antes de continuar, necesitan un dispositivo con acceso a Scratch y el archivo *Base.sb3*. Si ya cuentan con esto, carguen el proyecto.

Recuerden que para hacer esto deben abrir Scratch, ir al menú Archivo, seleccionar Cargar desde tu ordenador, ir al lugar donde guardaron el archivo y abrirlo.



Cuando lo hayan hecho, deberían ver la imagen de fondo de una habitación con una lámpara, un reproductor de sonido y un personaje. Este personaje será su asistente virtual.

Figura 2. Personaje de asistente virtual



Haz clic sobre la lámpara y sobre el equipo de sonido para encenderlos y apagarlos.

Haz clic en la bandera verde para iniciar el programa.



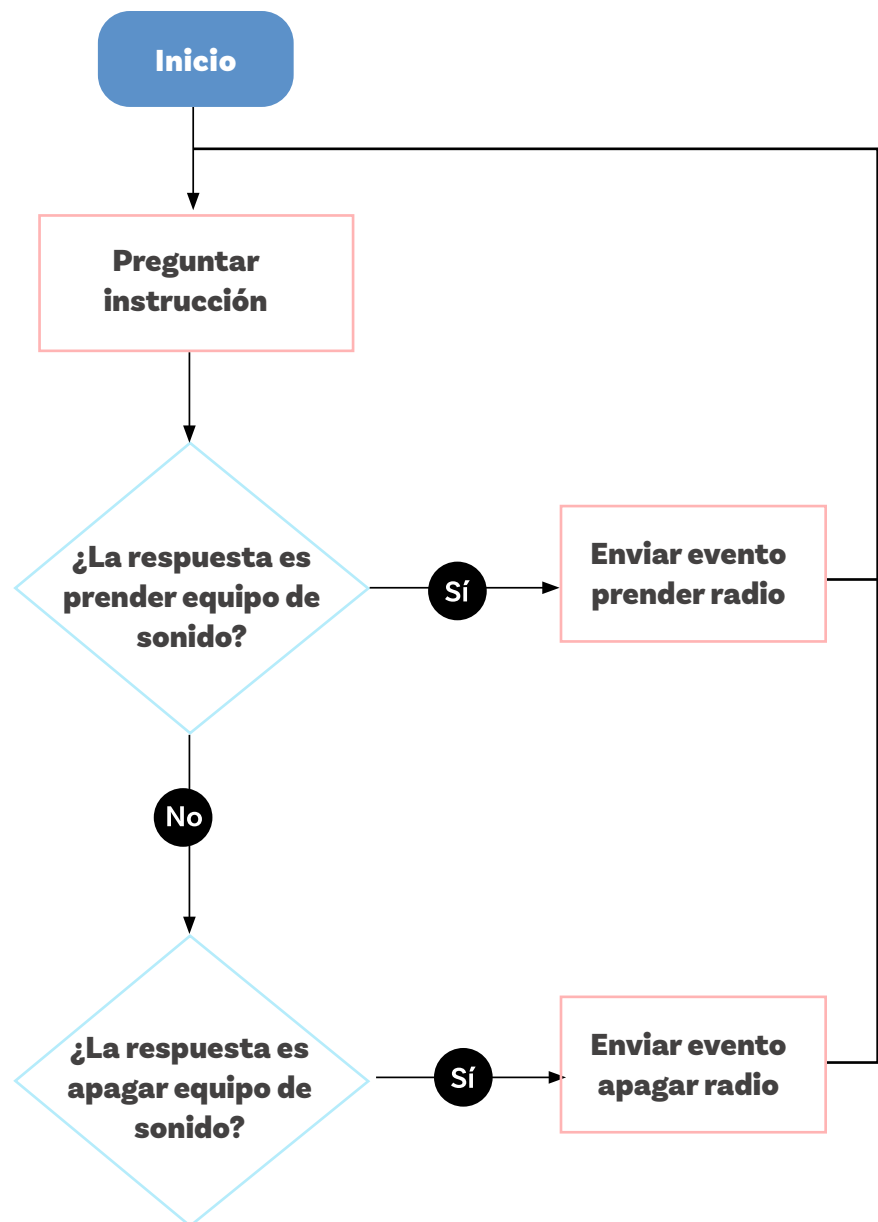
¿Cómo se llama el asistente virtual?

Después de presentarse, el asistente virtual les preguntará lo que quieren hacer y aparecerá un cuadro de texto para que escriban, como se ve en la Figura 3. Escribanle para averiguar qué sucede.

Ahora, su objetivo es que Picó, su asistente virtual, compare su respuesta con las instrucciones para prender o apagar la lámpara o el reproductor de audio (la grabadora) y luego envíe la instrucción al dispositivo correspondiente.

Figura 3. Ejemplo de interacción con el asistente virtual en Scratch





Tradicionalmente, esto implica utilizar bloques condicionales para comparar su respuesta con cada una de las posibles instrucciones. Por ejemplo, la lógica de funcionamiento del reproductor de sonido se vería como se muestra en el diagrama de flujo.

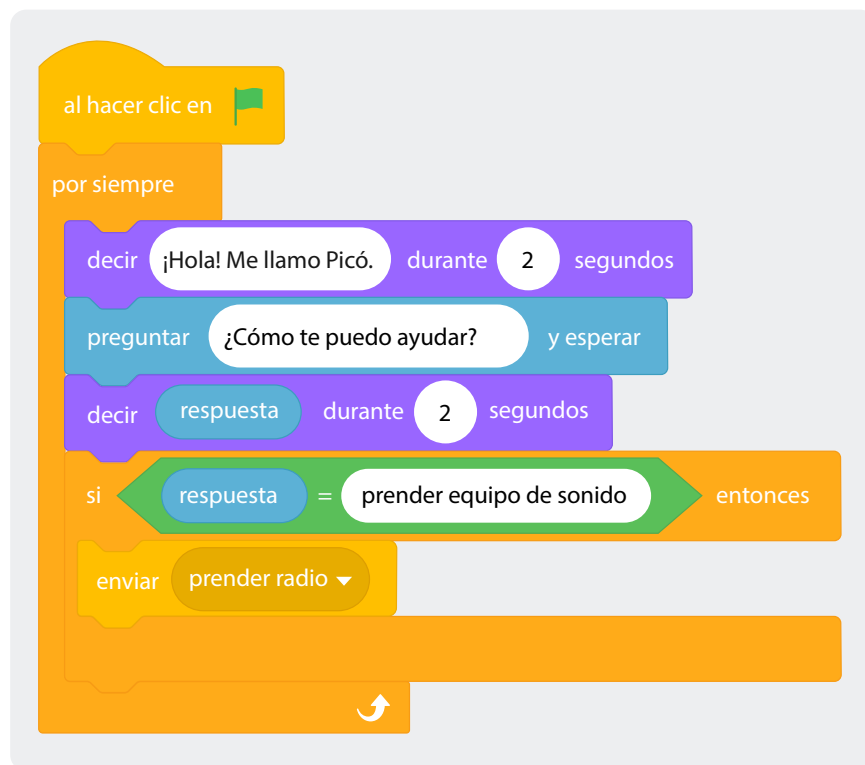
Usando condicionales, completen el programa que se presenta más abajo para que Picó encienda o apague el reproductor de sonido.

Observen que en la comparación del condicional deberán escribir la instrucción que quieren que se asocie a la acción. Esta instrucción no tiene que coincidir con el mensaje que se le envía al dispositivo.

Validen su programa y luego complétenlo para que Picó también controle la lámpara.

Cuando hayan terminado pídanle a su asistente virtual que controle los dos dispositivos usando instrucciones ligeramente diferentes, como “prende equipo de sonido” en vez de “prender equipo de sonido”. ¿Les entiende?

Figura 4. Programación de asistente virtual en Scratch



Anexo

Anexo 1.1

[illegible]

Diligencien el Anexo 1.1 con sus resultados. Tengan en cuenta que las instrucciones programadas corresponden a las frases específicas que incluyeron en su código, mientras que las instrucciones de validación corresponden a los casos de prueba que usaron.

Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

Revisa los aprendizajes de la sesión. ¿Crees que lograste alcanzarlos?

- 1 ¿Puedes explicar usos de la IA en la vida cotidiana?

☐ Sí

☐ Parcialmente

☐ Aún no

2 ¿Puedes identificar los límites de la programación tradicional en cuanto a la comprensión del lenguaje natural?

☐ Sí

☐ Parcialmente

☐ Aún no

Si tus respuestas a las preguntas anteriores fueron “Parcialmente” o “Aún no”, lee nuevamente los contenidos y toma nota de algunos usos de la IA en la cotidianidad. Amplía esta información, si es preciso, por medio de una consulta. De igual forma, revisa las instrucciones que se ingresaron al programa en Scratch para que el asistente virtual reconociera, por ejemplo, la instrucción “prender la lámpara” y reintenta dándole otras indicaciones para que encienda. Ej.: “prender la luz”, “prender el bombillo”. Registra lo que sucede.

Con tu grupo de trabajo, completen las siguientes frases:

El programa que hicimos en Scratch utilizando programación tradicional y que permite a un personaje prender y apagar una lámpara y un equipo de sonido nos ayuda parcialmente a resolver el reto porque _____

En este momento el asistente virtual que programamos tiene restricciones como _____

Cuando usamos instrucciones ligeramente diferentes, o le llamamos de forma diferente a los objetos, nuestro asistente virtual

Para que este asistente virtual comprenda estas diferentes expresiones, podríamos ajustar el programa _____

Aprovechen este espacio final para hacer un esquema gráfico que integre palabras e imágenes para resumir lo aprendido sobre la IA, los asistentes virtuales, sus características y la diferencia entre estos programas y los creados mediante programación tradicional.

Sesión 2

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Refinar la tomas de decisiones de un programa mediante la integración de condicionales con operadores booleanos.

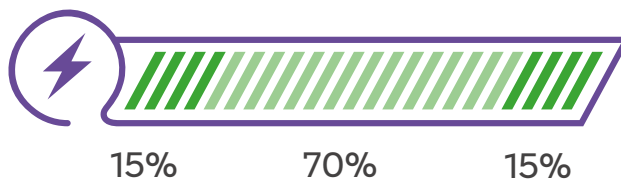


Crear y utilizar listas en Scratch.



Plantear soluciones con programación tradicional que incrementen un poco la comprensión del lenguaje natural por parte de un código.

Duración sugerida



Material por grupo:

- Anexo 2.1 (conservar durante toda la guía).
- Computador con acceso a Scratch y a los archivos previamente descargados del enlace suministrado en la introducción de la guía.



Lo que sabemos, lo que debemos saber



Esta sección corresponde al 15% de avance de la sesión

En esta sesión tú y tu equipo de trabajo continuarán mejorando su asistente virtual. En la sesión anterior lograron que prendiera o apagara un equipo de sonido o una lámpara escribiéndole una instrucción. Este es un primer paso para la creación de su prototipo, pues no tuvieron que escribir un código o instrucción extraña para que se realizara la acción, e incluso ustedes podían elegir la instrucción que querían que su asistente ejecutara.

Sin embargo, la comunicación todavía no es lo suficientemente natural y fluida. Si llegan a cambiar ligeramente la instrucción que dan, su asistente virtual no les entenderá. Por ejemplo, si en vez de escribir “prender equipo de sonido” escriben “prende equipo de sonido”, el programa ya no funciona. Ahora verás cómo plantear una posible solución a este problema.

Haz una lista de 3 variaciones de la instrucción “prender equipo de sonido”. Luego, observa el siguiente código y piensa cómo lo modificarías para incluir esas nuevas 3 instrucciones.

Figura 1. Programación de nuevas instrucciones para el asistente virtual en Scratch



Glosario



Operador booleano: Símbolo que se utiliza para relacionar variables cuyos valores son falso o verdadero. Estas variables se llaman variables booleanas.



Operador “o”: Relaciona dos variables booleanas para obtener un solo resultado, otra variable booleana. El resultado es verdadero (V) si por lo menos una de las variables (A, B) conectadas con el operador “o” es verdadera, como se ve en la tabla abajo.



Operador “y”: Relaciona dos variables booleanas para obtener un solo resultado, otra variable booleana. El resultado es verdadero (V) solo si ambas variables (A, B) conectadas con el operador “y” son verdaderas, como se ve en la tabla abajo.

Condiciones		Resultado	
A	B	A o B	A y B
V	V	V	V
V	F	V	F
F	V	V	F
F	F	F	F

Manos a la obra**Conectadas**

Esta sección corresponde al 85% de avance de la sesión

Organízate en grupos de 2 a 3 personas, de acuerdo con las indicaciones de tu docente. Comparen su lista de variaciones de la instrucción y comenten las soluciones que plantearon. Elijan 3 de las variaciones y una solución e impleméntenla en el proyecto que encuentran en el archivo *Prender_equipo.sb3*, que descargaron desde el enlace al comienzo de la guía. Recuerden que deben cargar este archivo en *Scratch* como lo hicieron en la Sesión 1.

Hay varias formas de que el programa reconozca variaciones de la instrucción. A continuación, verán dos de ellas: la inclusión de más bucles condicionales y la inclusión de varias condiciones en un mismo condicional.

En el primer caso se requiere utilizar más bucles condicionales. Esto lo pueden ver en el código que se presenta a continuación, al que se le agregaron 2 variaciones.

Nota

Al crear la lista puedes elegir si quieres que solo sea visible para el objeto del que tienes el código abierto o si quieres que todos los objetos la puedan ver. En este caso, para ser más realistas, como con el único objeto con el que te puedes comunicar con texto es Picó, solo este debería poder ver la lista. La lámpara y el equipo de sonido lo único que pueden hacer es recibir una orden específica de Picó para prenderse o apagarse.

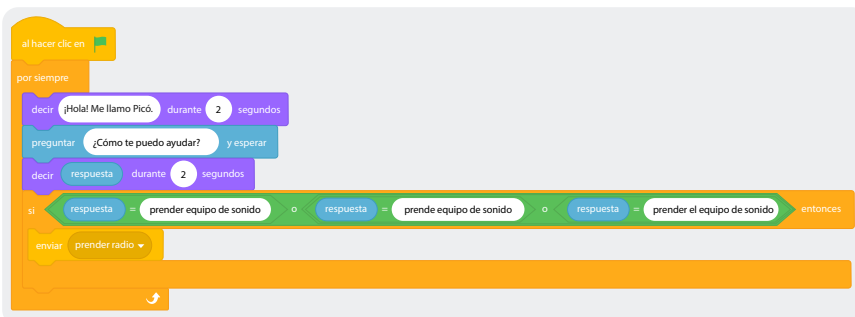
Figura 2. Programación de condicionales para el asistente virtual

Se deben agregar muchos bloques para incluir cada variación, ¿no les parece? Esto no es práctico si se quieren agregar, al menos, 3 variaciones a cada una de las 4 instrucciones.

Figura 3. Bloques para creación de lista



Otra forma es usando el operador booleano “o” para incluir varias condiciones en un mismo condicional como se ilustra a continuación:



Esta segunda opción es un poco mejor, ya que no se usan tantos bloques.



Sin embargo, ¿se imaginan agregando 10 variaciones? ¡Quedaría muy largo! Además, cada vez que quieran agregar una variación tendrían que modificar los bloques de código. ¿Será que hay una mejor manera?

Sí hay una mejor manera: usando listas. Una **lista** es una estructura de datos que permite almacenar varios datos del mismo tipo bajo un mismo nombre. Los datos se van agregando. Ahora van a ver cómo usarlas y qué ventajas tienen.

Primero, carguen de nuevo el proyecto que encuentran en el archivo *Prender_equipo.sb3*.

Teniendo abierto el código de Picó, vayan a la pestaña Código, sección Variables y seleccionen Crear una lista, como se ve en la *Figura 4*.

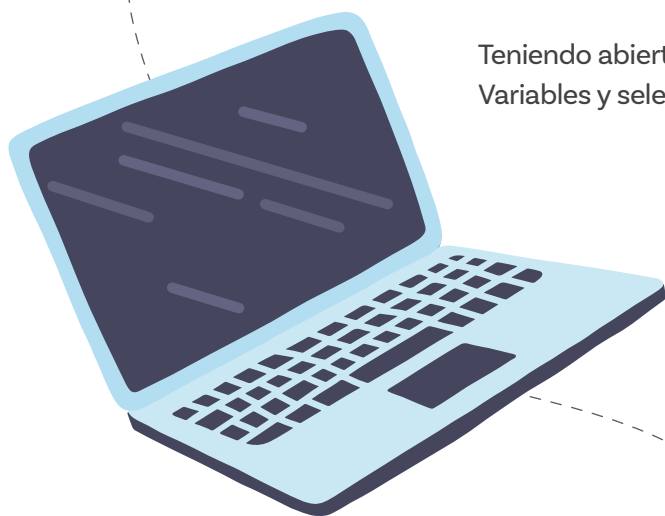


Figura 4. Bloques de variables en Scratch



Nota

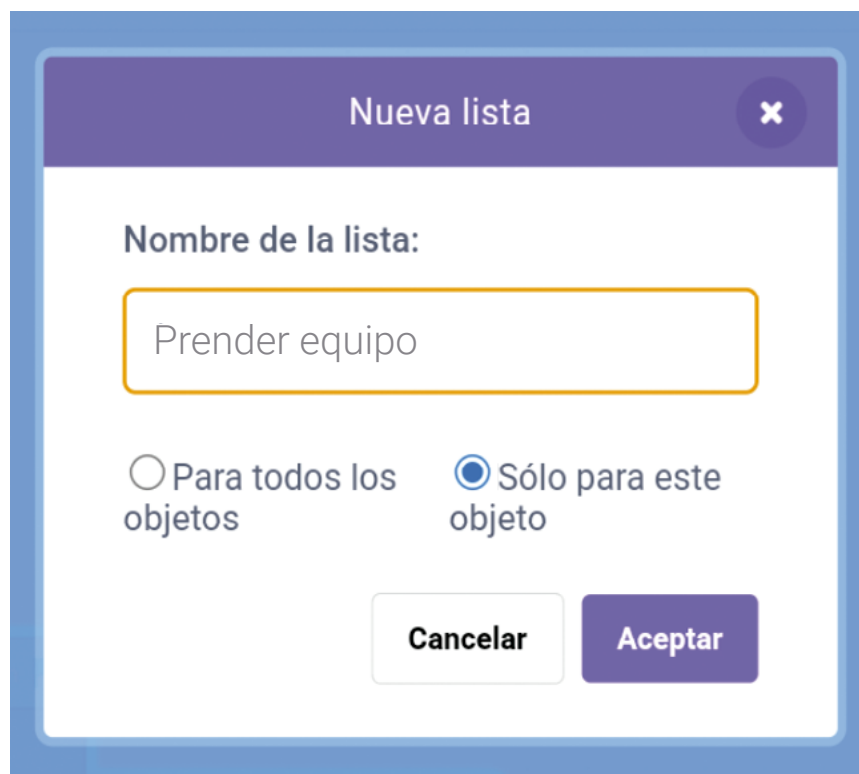
Si en el escenario la lista queda encima del radio y no les deja ver si está encendido o apagado, pueden hacer que no aparezca ahí. Para esto, teniendo abierto el código de Picó, vayan a la pestaña Código, sección Variables y deseleccionen su lista:



Prender equipo

Deberán crear una lista para cada una de las cuatro instrucciones: prender y apagar el equipo de sonido y prender y apagar la lámpara.

Comiencen a modificar la opción de prender el equipo de sonido.



Escriban un nombre para la lista, por ejemplo: “Prender equipo”, seleccionen la opción Solo para este objeto. Luego, hagan clic en Aceptar, como pueden observar en la figura anterior.

Los bloques del menú Variables se debieron actualizar, incluyendo la lista que acaban de crear, como se ve en la Figura 4. Verifiquen que el bloque Prender equipo esté seleccionado. En caso de que no lo esté, selecciónenlo.

Figura 5. Ejemplo de interacción con el asistente virtual en Scratch

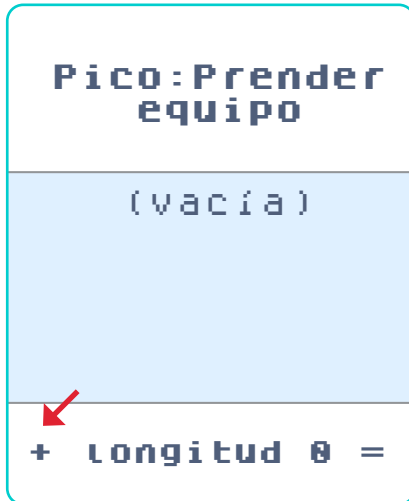
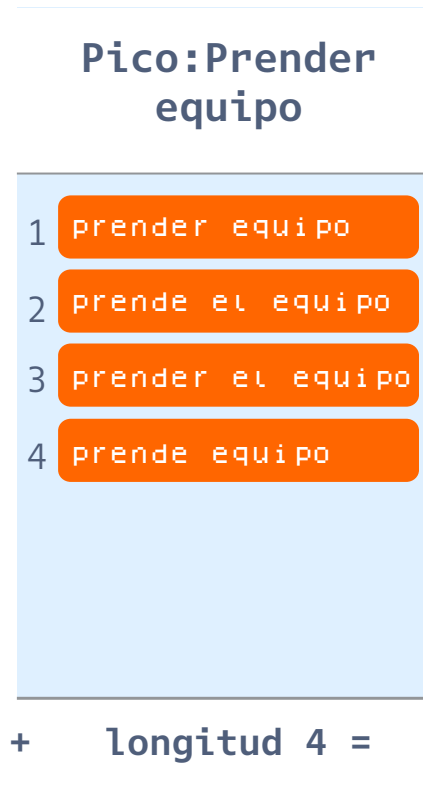


Figura 6. Variaciones de la instrucción agregadas a la lista Prende equipo



En el escenario les debe aparecer un recuadro que representa la lista, como ven en la *Figura 5*. Hagan clic en el + para agregar una de las instrucciones que quieren que Picó reconozca para prender el equipo de sonido y luego presionen la tecla Intro o Enter. Continúen agregando las otras variaciones de la instrucción de la misma manera. Al final deben obtener algo como lo que se muestra en la *Figura 6*.

Ya crearon la lista, ahora deben usarla en el código de Picó para que compare lo que le escriben con cada uno de los datos de la lista.

Para esto deben remplazar el bloque de comparación con el de igualdad, ya que este solo puede comparar un número con otro. Exploren los bloques en la sección de operadores.



¿Cuál operador creen que serviría para saber si un texto está en la lista?

En la *Figura 7*, pueden ver el código para que Picó tenga en cuenta la lista de instrucciones para prender el radio.

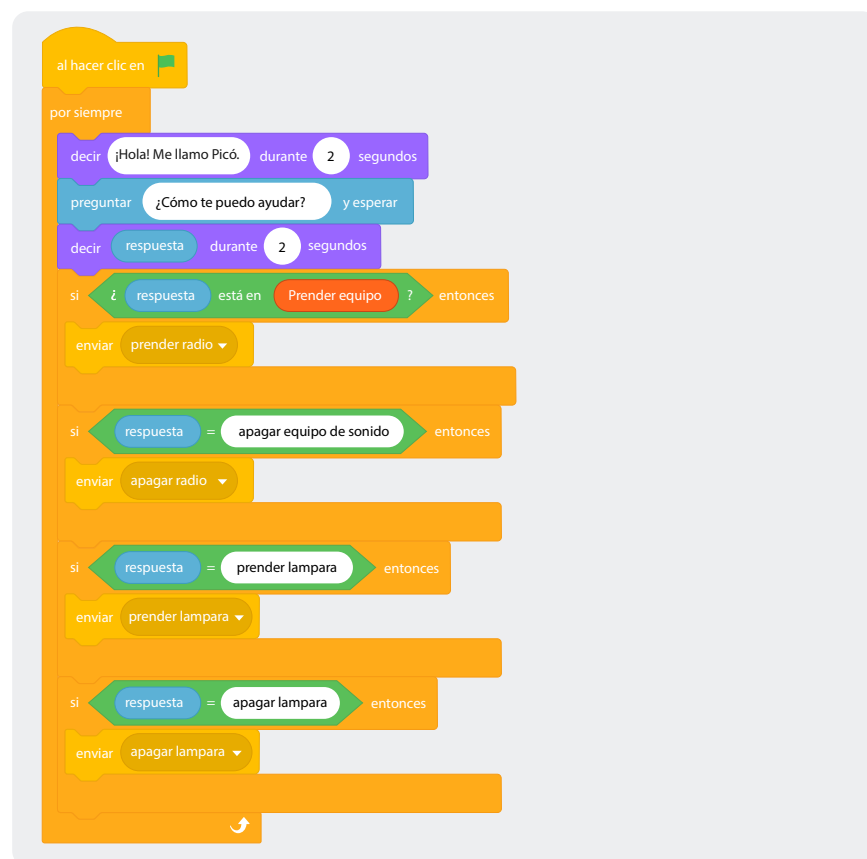


Anexo

Anexo 2.1

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Correcto?
Prender lámpara			
Apagar lámpara			

Figura 7. Programación del asistente virtual que incluye la lista Prender equipo



¿Qué pasa si le preguntan algo de una forma ligeramente diferente a las variaciones que le incluyeron?

Por ejemplo, si le escriben: “Hazme el favor de encender la radio”, si Picó no les entiende, podrían agregar esa variación a la lista. Pero ¿podrías agregar todas las variaciones posibles? Registren sus resultados en el Anexo 2.1.

Glosario



Lista: Estructura de datos que permite almacenar varios datos bajo un mismo nombre. Todos los datos deben ser del mismo tipo: por ejemplo, números o letras. En otros lenguajes de programación, como en MakeCode, se le conoce como arreglo.

Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

Revisa los aprendizajes de la sesión. ¿Crees que lograste alcanzarlos?

- 1 Refinar las tomas de decisiones de un programa mediante la integración de condicionales con operadores booleanos
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no
- 2 Crear y utilizar listas en Scratch
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no
- 3 Plantear soluciones con programación tradicional que incrementen un poco la comprensión del lenguaje natural por parte de un código
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no

¿Cómo te fue? Si tus respuestas a las preguntas anteriores fueron “Parcialmente” o “Aún no”, revisa nuevamente los contenidos de esta sesión, tratando de seguir paso a paso las instrucciones que se ofrecen. Mientras lo haces, toma nota de las preguntas que te surjan, y consulta con tu docente para aclararlas. Ten presente también que, si pudiste agregar mediante listas las variaciones de las instrucciones, lo que hiciste fue plantear una solución con programación tradicional para mejorar la comprensión del lenguaje natural por parte de tu prototipo de asistente virtual.

Hoy aprendiste a utilizar listas y programación tradicional para que tu asistente virtual pueda prender y apagar una lámpara y un equipo de sonido. ¿Cómo te ayuda eso a ir dando solución al reto?

Con tus compañeras y compañeros de grupo, discutan las siguientes preguntas:



¿Qué restricciones identifican en el asistente virtual que acababan de programar?

Si usan instrucciones ligeramente diferentes, o si le dicen de forma diferente a los objetos, ¿les entiende Picó?

¿Qué pasa si se equivocan en una letra de la instrucción?

¿De cuántas formas diferentes le pueden pedir a este asistente que haga cada una de las acciones? ¿Mejóro con respecto a la sesión anterior? Comparen sus resultados de las dos sesiones.

¿Les parece que su asistente virtual ya es lo suficientemente inteligente? En caso negativo, ¿qué le faltaría?

Escriban una breve conclusión a partir de sus respuestas a estas preguntas.

Finalicen creando una actividad breve que les permita repasar de forma divertida lo aprendido en esta sesión en cuanto a listas, operadores booleanos y limitaciones que hayan detectado en la programación tradicional. ¿Podrían crear tarjetas con preguntas para usar en un juego de mesa? ¿Se les ocurre alguna historia o cuento donde usen estos conceptos de forma correcta? ¿Qué tal si hacen un cuestionario que puedan intercambiar con otro grupo? Compartan su actividad, siguiendo las indicaciones de su docente.

Sesión 3

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Identificar patrones en números y en texto.

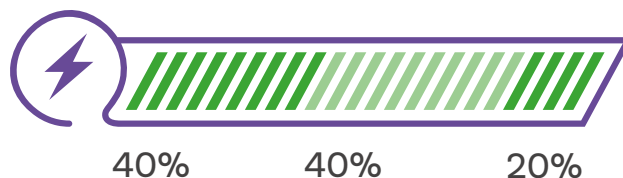


Entrenar una IA, mediante aprendizaje automático supervisado, para que reconozca texto.



Comparar las diferencias entre el aprendizaje automático y la programación tradicional.

Duración sugerida



Material para la clase

- ☐ Anexo 3.2 (conservar durante toda la guía) y Anexo 3.1
- ☐ Computador con acceso a internet.



Lo que sabemos, lo que debemos saber



Esta sección corresponde al 40% de avance de la sesión

En esta sesión vas a aprender cómo hacer la comunicación con tu asistente virtual aún más natural y fluida. En la sesión anterior usaste una lista para incluir diferentes formas de decir la misma instrucción.



¿Crees que sería una buena idea continuar agregando instrucciones a tu lista hasta que tu asistente virtual entienda todas las posibles formas de decir una instrucción?

Quizás te hayas dado cuenta de que hay demasiadas formas de pedir lo mismo. Por ejemplo, podrías agregar al principio “Por fa”, “Por favor”, “Hazme el favor”, “Hágame el favor”, “¿Podrías”, “¿Serías tan amable de”, etc. Incluso podrías agregar algunas de las anteriores al final. Además, hay diferentes formas de referirse a los objetos, como radio, grabadora, equipo de sonido, reproductor de sonido o lámpara y luz. También hay sinónimos de la acción como prender y encender. Si te pones a pensar en todas las posibles combinaciones, te darás cuenta de que son demasiadas. Además, no tienes forma de garantizar que hayas incluido absolutamente todas las variaciones posibles.



Anexo

Anexo 3.1

Nombre: _____

1. ¿Qué número creen que sigue en la secuencia: 4, 8?
2. ¿Por qué?
3. ¿Es la única respuesta que se ajusta a la secuencia?
4. ¿Qué se necesita para poder estar seguros del patrón en este caso?
5. Completen la siguiente tabla:

Secuencia	Siguiente valor	Patrón
1, 4, 7, 10		
123, 101, 106, 99		
13, 21, 55, 111		
2, 6, 26, 80, 342		

6. Separa estas palabras en dos grupos escribiéndolas en las dos casillas de la primera fila de la tabla que aparece abajo: molesto, mal, malgenado, enfurecido, triste, cansado, feliz, exaltado, emocionado, encantado, contento y enérgico.
7. Luego escribe el patrón, lo que tiene en común cada grupo en la segunda fila.

Entre los sistemas de IA, hay algunos que no requieren que se programe cómo debe responder el sistema a cada instrucción, en cambio entrenamos estos sistemas para que aprendan. A esto se le conoce como **aprendizaje automático**. Estos sistemas tienen algoritmos para detectar patrones en conjuntos de datos y aprender por sí mismos lo que deben hacer: predecir algo o tomar una decisión.

Los seres humanos somos buenos encontrando patrones. Veamos algunos ejemplos.

En esta parte de la sesión vas a trabajar en grupos. Reúnete en grupos de 2 a 3 personas, según las indicaciones de tu docente y respondan las primeras 3 preguntas del Anexo 3.1 sobre la secuencia numérica 4, 8.



¿Será que la respuesta que dieron es el único patrón que se ajusta a esos dos números?

*¿Consideraron el doce como el número siguiente?
¿Cuál es el patrón en este caso?*

*¿Consideraron el dieciséis como el número siguiente?
¿Cuál es el patrón en este caso?*



Continúen respondiendo las preguntas 4 y 5 del Anexo 3.1.

Como se habrán dado cuenta, es necesario contar con varios datos para poder deducir el patrón de una secuencia. En algunos casos, si se cuenta con pocos datos, puede que más de un patrón corresponda. En otros, al tener más datos es más fácil darse cuenta del patrón y verificarlo.

Por otro lado, observen que al aprender el patrón ya podrían saber si cualquier número pertenece a la secuencia, sin necesidad de haberlos listado todos.

Vean otro ejemplo de patrones.



Observen las palabras en el punto 6 del Anexo 3.1 y diligencien la tabla.

Al igual que en el caso de los números, una vez han identificado el patrón es posible clasificar palabras nuevas en uno de los dos grupos.

Sin embargo, el patrón de las palabras tiene características diferentes al de los números. En el caso del patrón con números la respuesta era única, si había suficiente información. La pertenencia o no a la secuencia numérica no se presta a interpretación, mientras que la clasificación de una palabra dentro de uno u otro grupo sí. Por ejemplo, quizás para algunas personas la palabra “meditativa” esté más relacionada con una emoción positiva, mientras que para otras lo está con una negativa. Por otro lado, en las secuencias de palabras, además de las palabras se pueden establecer unos nombres de los grupos o categorías a las que estas deben pertenecer.

Una forma de entrenar a una inteligencia artificial para clasificar es por medio del aprendizaje automático supervisado. En este se le presentan ejemplos etiquetados a la IA durante el entrenamiento.

Glosario



Aprendizaje automático: Capacidad de sistemas de IA de realizar predicciones o tomar decisiones basadas en patrones que los mismos sistemas detectan durante una fase de entrenamiento, en vez de ser programados tradicionalmente, caso por caso.



Patrón: Regla que determina la pertenencia de un elemento a un grupo.



Aprendizaje automático supervisado: Tipo de aprendizaje automático en el cual se le presentan ejemplos al algoritmo de aprendizaje durante el entrenamiento. Los ejemplos constan de un dato y la etiqueta que le corresponde.

Enlace



Manos a la obra

Conectadas



Esta sección corresponde al 80% de avance de la sesión

Ahora van a entrenar una IA usando el aprendizaje automático supervisado para mejorar la comprensión del lenguaje natural del asistente virtual que están programando en *Scratch*. Continúen trabajando en grupos según las indicaciones de su docente.

- 1 Vayan al siguiente enlace:
<https://machinelearningforkids.co.uk/>
- 2 En caso de que la página no les aparezca en español, cambien el idioma haciendo clic en la esquina superior derecha y seleccionando español. Luego hagan clic en Empezar, como se muestra en la *Figura 1*.

Figura 1. Opción para cambio de idioma

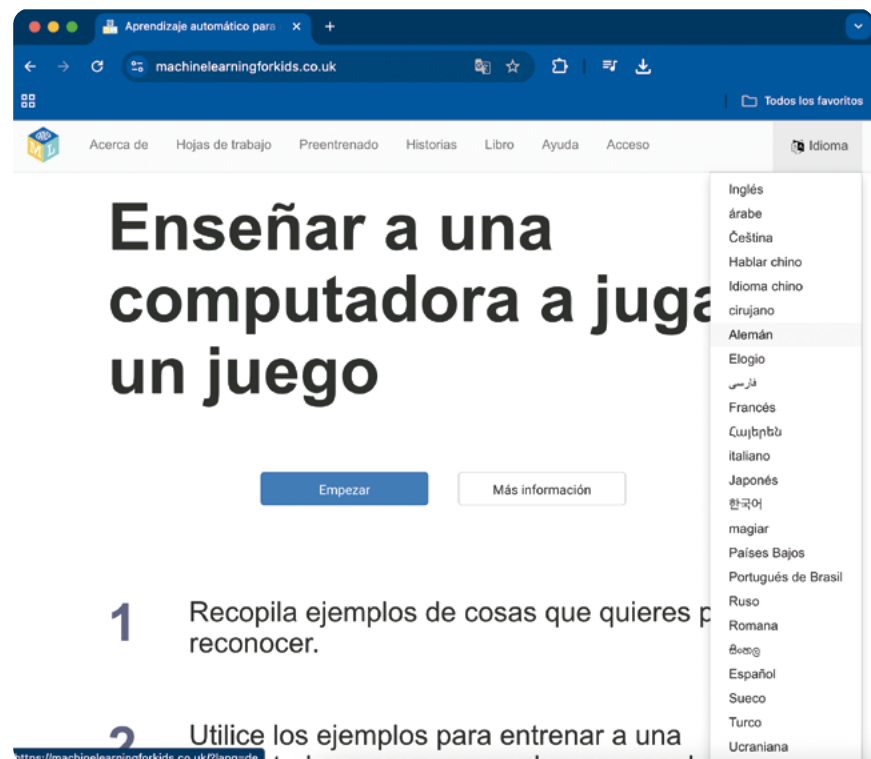


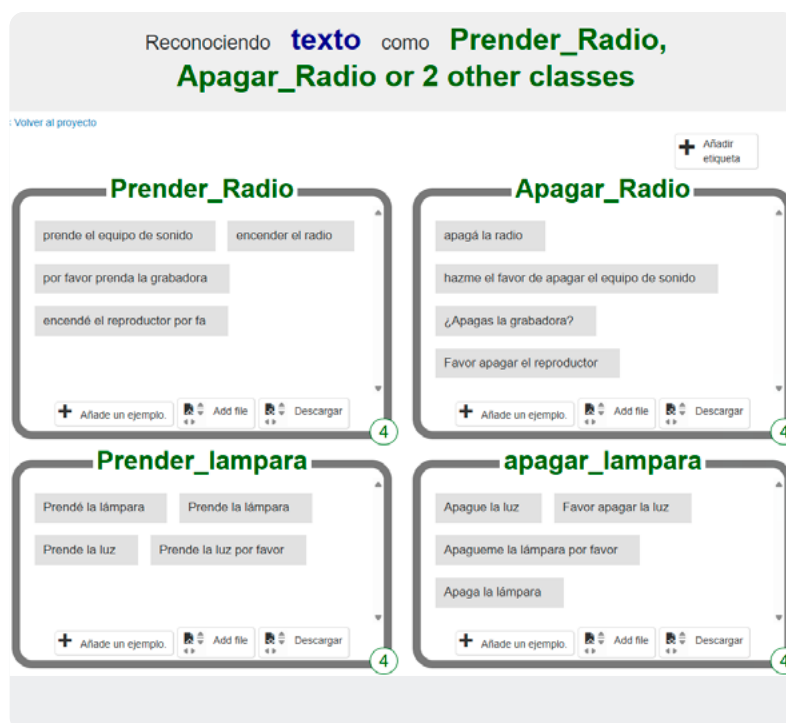
Figura 2. Representación gráfica de las instrucciones en el punto 3



3 En la Figura 2 encuentran imágenes correspondientes a las instrucciones que se encuentran a continuación.

- En la siguiente pantalla seleccionen la opción Pruébalo ahora.
- Luego añadan un proyecto nuevo.
- Después escriban el nombre del proyecto, por ejemplo, "Asistente virtual".
- Seleccionen la opción Reconociendo texto como tipo de proyecto, Spanish como idioma, In your web browser como tipo de almacenamiento y hagan clic en Crear.
- Con estas instrucciones se creó el proyecto. Ahora hagan clic dentro del recuadro del proyecto para entrar en él.
- Seleccionen Añadir etiqueta, escriban "prender radio" y agréguenla.

4 Luego hagan clic en Añade un ejemplo, escriban una de las formas en las que se puede pedir prender el equipo de sonido y seleccionen **agregar**. Repitan el procedimiento para tener cuatro variaciones de las cuatro instrucciones de tu asistente virtual: prender/apagar el equipo de sonido/la lámpara. Al terminar deberían tener algo como lo que se muestra a continuación. Hagan clic en Volver al proyecto.



- 5 Deberían llegar a una pantalla como la que se muestra abajo. Seleccionen Aprender y Probar.



- 6 Luego les aparecerán un par de mensajes. Uno de ellos, que pueden ver la Figura 2, les dice que no tienen suficientes ejemplos en cada etiqueta.

Figura 3. Mensaje que indica que no hay suficientes ejemplos de etiqueta en el modelo

¿Y ahora qué hay que hacer?

¡Seguimos!

Vuelve a la página [Entrenar](#) y añade más ejemplos para cada etiqueta .

Cuanto más añadas, mejor aprenderá, pero como mínimo, necesitas five ejemplos.

- 7 La herramienta pide un mínimo de 5 ejemplos por etiqueta y además les dice que mientras más tengan, mejor aprenderá el modelo. Vuelvan a la sección Entrenar para agregar por lo menos 5 ejemplos en cada etiqueta y vuelvan a Aprender y Probar.

¿Y ahora qué hay que hacer?

¿Lista para empezar con el entrenamiento?

Haz clic en el botón de abajo para empezar a entrenar el modelo de aprendizaje automático con los ejemplos que se han agregado hasta ahora.

(O vuelve a la página [Entrenar](#) si quieres añadir antes más ejemplos.)

Información del entrenamiento:

Entrena un nuevo modelo

- 8 Observen que el mensaje cambió y que ahora aparece la opción Entrenar un nuevo modelo. Selecciónenla. Aparecerá un recuadro informativo como el que se ve en la Figura 3.



Figura 4. Información del modelo entrenado

Información del entrenamiento:

Modelo se inició started training at: Tuesday, April 29, 2025 7:50 AM

Estatus actual: Training

Último chequeo del modelo: less than a minute ago ⓘ

Cancelar el entrenamiento



Agregue texto para ver cómo lo identifica tras el entrenamiento.

Introduzca un texto de prueba aquí

Probar

Información del entrenamiento:

Modelo se inició comenzó a entrenar en: Martes, 29 de abril de 2025, 7:52 a. m.

Estado actual: Disponible

El modelo se borrará automáticamente cuando pasen: : Martes, 29 de abril de 2025, 11:52 a. m.

Borra este modelo

Entrena un nuevo modelo

- 9 Unos minutos después, cuando haya terminado el entrenamiento, el recuadro se actualizará y aparecerá otro en el que pueden escribir el texto para que su asistente lo clasifique, como se ve a continuación. Debajo de este cuadro aparece la predicción.

Agregue texto para ver cómo lo identifica tras el entrenamiento.

Introduzca un texto de prueba aquí

Probar

Anexo

Anexo 3.2

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Aprendió?
Encendido lámpara			
Apagado lámpara			

10 Validen varias formas de pedirle a su asistente que realice las cuatro acciones para las que lo entrenaron y registren tus resultados en el Anexo 3.2:

- ☐ Intenten validarlo usando variantes de las instrucciones con las que lo entrenaron. ¿Funciona?
- ☐ Ahora intenten usando variantes de las instrucciones que no estaban en el entrenamiento. ¿Funciona?
- ☐ Intenten usando formas muy diferentes, por ejemplo, funciona si le dicen:
 - Está muy oscuro (debería prender la lámpara)
 - Hay mucha luz (debería apagar la lámpara)
 - Silencio (debería apagar el radio)
 - Música (debería prender el radio)

Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

Revisa los aprendizajes de la sesión. ¿Crees que lograste alcanzarlos?

1 ¿Puedes identificar patrones en números y en texto?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

2 ¿Puedes entrenar una IA, mediante aprendizaje automático supervisado, para que reconozca texto?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

3 ¿Puedes comparar las diferencias entre el aprendizaje automático y la programación tradicional?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

Si tus respuestas a las preguntas anteriores fueron “Parcialmente” o “Aún no”, devuélvete, paso a paso, a realizar la práctica propuesta. Consulta con tus compañeras y compañeros del grupo y con tu docente, las inquietudes que te surjan.

Con tu grupo de trabajo nuevamente, encuentren las respuestas a estas preguntas.

Considerando el reto propuesto, ¿para qué les servirá haber entrenado una IA por medio de aprendizaje automático supervisado para reconocer texto?

¿Ven alguna relación entre el reconocimiento de patrones y la posibilidad de que su asistente virtual entienda más instrucciones?

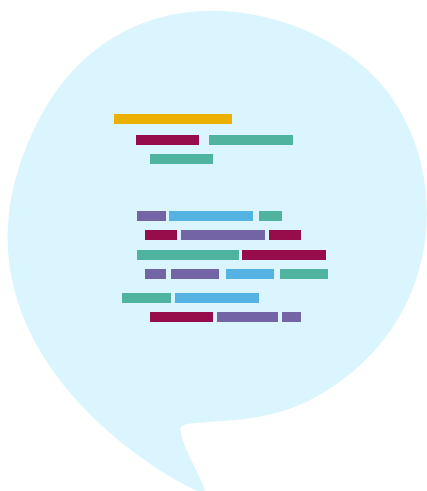
Si usan instrucciones ligeramente diferentes, o si le dicen de forma diferente a los objetos, ¿les entiende su asistente?

¿Qué pasa si se equivocan en una letra de la instrucción?

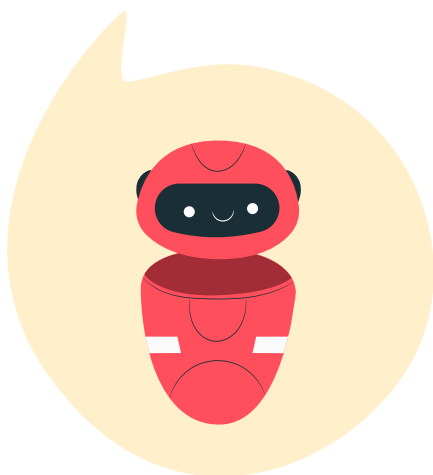
¿De cuántas formas le pueden pedir a su asistente que haga cada una de las acciones? ¿Crees que hay un límite?

Finalmente, completen esta tabla comparativa entre la programación tradicional y el aprendizaje automático.

Programación tradicional	Aprendizaje automático



Pueden, igualmente, escribir algunas de las definiciones de palabras y de conceptos que recordaron con esta sesión. Ej.: patrones, aprendizaje automático y aprendizaje automático supervisado.



Sesión 4

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Explicar la relación entre los componentes de una IA de aprendizaje automático y la creación de un modelo.

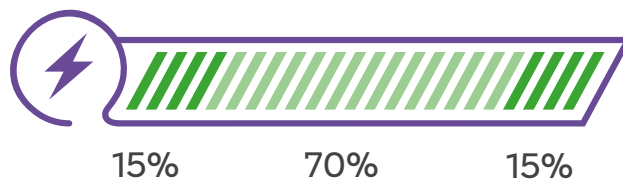


Explicar en qué consisten las etapas de entrenamiento y validación de un modelo de IA.



Explicar cómo el desarrollo de la IA se beneficia de la participación de personas con perspectivas diferentes.

Duración sugerida



Material para la clase

- ☐ Anexos 4.1 (conservar para la siguiente sesión) y 4.2.
- ☐ Computador con acceso a internet.



Lo que sabemos, lo que debemos saber



Esta sección corresponde al 15% de avance de la sesión

En la sesión anterior entrenaste una IA para que reconociera un patrón en varias versiones de las instrucciones para tu asistente virtual. En esta sesión mejoraremos aún más esa IA entrenada. Continúa para ver cómo.

Un sistema de IA de aprendizaje automático tiene tres componentes principales que se muestran en la siguiente imagen.



A diferencia de la programación tradicional, en estos sistemas no basta con tener un algoritmo. Requieren aprender a partir del conjunto de datos antes de poder ser usados para generar sus decisiones o predicciones. Por esta razón, el conjunto de datos se suele llamar datos de entrenamiento. Estos datos son los que determinarán lo que hará, lo que aprenderá la IA. En la sesión anterior los datos de entrenamiento eran los ejemplos que le diste de cada instrucción.

El primer algoritmo es más general y se llama algoritmo de aprendizaje. En la sesión pasada usaste un algoritmo de aprendizaje de reconocimiento de texto. Este algoritmo también podría usarse para detectar correo no deseado, o para catalogar las palabras en los dos grupos en el ejercicio de reconocimiento de patrones.

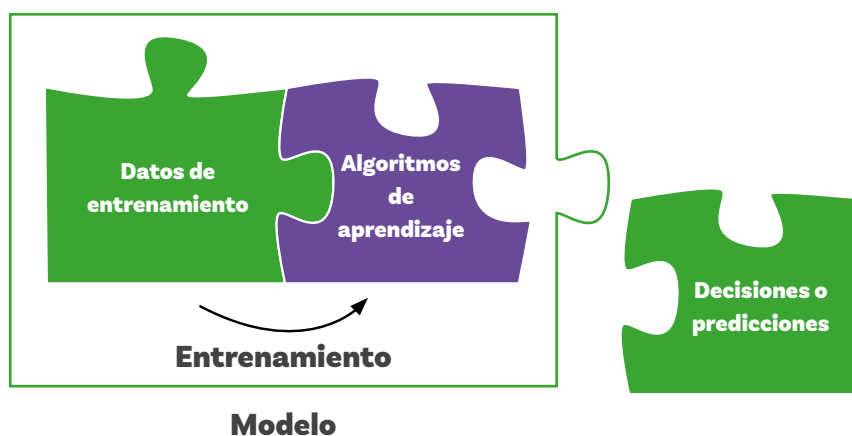
Luego, la etapa en la que el sistema está aprendiendo a detectar los patrones se llama entrenamiento. Una vez se ha entrenado al sistema con unos datos de aprendizaje se obtiene lo que se suele llamar un **modelo**.

El modelo es un programa, otro algoritmo, un código, diferente del algoritmo de aprendizaje. Durante el entrenamiento, el algoritmo de aprendizaje ajusta el modelo para que sus predicciones o decisiones sean más precisas.

Por ejemplo, en la sesión pasada, el programa para reconocer cuatro instrucciones es un modelo creado a partir de un algoritmo de aprendizaje para reconocimiento de texto. Fue entrenado con ejemplos de instrucciones, etiquetados como un conjunto de datos de aprendizaje.

Si agregamos el modelo a los tres componentes principales de un sistema de IA de aprendizaje automático, tendríamos algo como lo que se muestra en la *Figura 1*.

Figura 1. Elementos involucrados en el entrenamiento de un modelo de IA



Sin embargo, como te habrás dado cuenta en las sesiones anteriores, realizamos un paso más antes de considerar que nuestro sistema está listo. Este paso ha hecho que nos devolvamos para continuar mejorándolo.

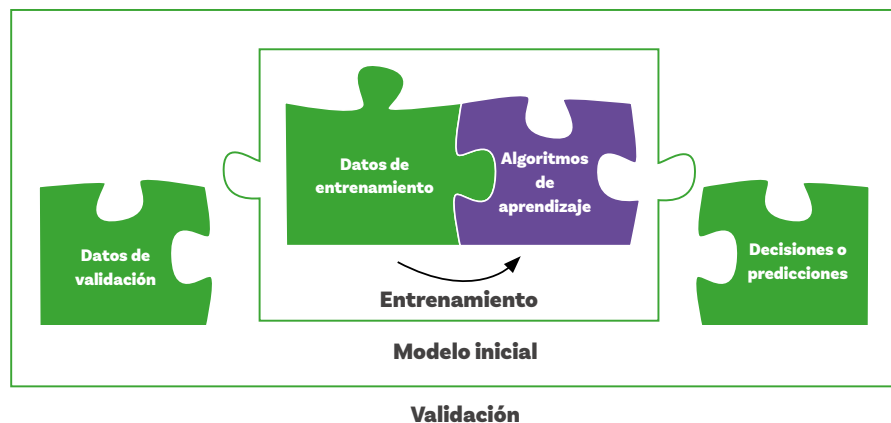
Este paso consiste en una primera verificación de que el modelo esté funcionando bien. Podríamos llamar a esta etapa **validación** y, por ende, los datos usados en ella se llaman datos de validación. En caso de que los resultados de la validación no sean satisfactorios, se puede volver a entrenar el modelo con más datos de entrenamiento.



¿Crees que los datos de validación deben ser los mismos que los de entrenamiento?

Recuerda que, a diferencia de la programación tradicional, en el aprendizaje automático la idea es que el modelo detecte patrones para que no sea necesario especificarle todos los casos posibles. De esta forma el modelo estará en la capacidad de hacer predicciones a partir de datos diferentes a los del entrenamiento. Así que, para validarlo, se deben usar datos diferentes, como se muestra en la *Figura 2*.

Figura 2. Validación del modelo entrenado



Glosario



Modelo: Programa que ha sido creado entrenando un algoritmo de aprendizaje con unos datos de entrenamiento para un fin específico.



Validación de un modelo: Evaluación inicial del modelo, usándolo, con el fin de mejorarlo.

Anexo

Anexo 4.1

	Instrucciones programadas
Prender radio	
Apagar radio	

Manos a la obra

Conectadas



Esta sección corresponde al 85% de avance de la sesión

Vamos a ver la importancia de validar un modelo. Organízate en grupos de 2 a 3 personas, de acuerdo con las indicaciones de tu docente. Vuelvan a crear un proyecto de reconocimiento de texto en el siguiente enlace: <https://machinelearningforkids.co.uk/>, tal como lo hicieron en la sesión pasada. Van a volver a crear las cuatro etiquetas correspondientes a las cuatro acciones que su asistente virtual debe realizar. En cada una escriban tantos ejemplos de variaciones de la instrucción como crean conveniente. Pueden usar el Anexo 3.2. Escriban sus datos de entrenamiento en el Anexo 4.1. Luego entrenen su modelo.

A pesar de que se sientan muy seguros de su modelo, es necesario validarlo para ver si ya está listo. Dejen la ventana del navegador de internet maximizada en la caja para ingresar las instrucciones para validar su modelo visible. Una vez lo hagan, deberían ver algo así:

Agregue texto para ver cómo lo identifica tras el entrenamiento.

Introduzca un texto de prueba aquí

Probar

Identificado como **prender_radio**
con un 76% de confianza

Información del entrenamiento:

Modelo se inició comenzó a entrenar en: Jueves, 30 de enero de 2025 14:10 horas

Estado actual: Disponible

El modelo se borrará automáticamente cuando pasen: Jueves, 30 de enero de 2025 18:10 horas

Borra este modelo

Entrena un nuevo modelo

Ahora, dejen en su puesto de trabajo una copia del Anexo 4.2, pero guarden el Anexo 4.1.

Según les indique su docente, levántense de sus espacios de trabajo y vayan a reubicarse en el computador de otro grupo. Tienen ahora la

Anexo

Anexo 4.2

Instrucciones de validación		¿Averigüé?	Confianza
Prender radio			
Apagar radio			

misión de escribir instrucciones en el modelo creado por otro grupo para validarlo. La idea es que sean creativos y le pregunten de formas no evidentes, pero que un ser humano entendería. Entre los textos que escriban para validar, incluyan errores, por ejemplo, de ortografía. Además, incluyan también un texto que no tenga relación alguna con las cuatro acciones que debe realizar el asistente virtual. A medida que validan el modelo de otro grupo, diligencien el Anexo 4.2 que se encuentra en ese puesto de trabajo.

Llenen la última columna con el porcentaje que aparece debajo de la predicción, marcado como *confidence*, como se ve en la Figura 3.

Sigan las instrucciones de su docente para pasar de un grupo a otro y validar su modelo tantas veces como sea requerido. Luego, regresen a su puesto de trabajo.

Observen los resultados consignados en su Anexo 4.2.

Figura 3. Nivel de confianza del modelo

Agregue texto para ver cómo lo identifica tras el entrenamiento.

prendeme la radio

Identificado como **prender_radio**
con un 76% de confianza



¿Su modelo logró interpretar correctamente todas las instrucciones pertinentes?

¿Encontraron las personas que validaron su modelo formas de dar la instrucción en las que su grupo no había pensado antes?

Si lo consideran necesario, agreguen datos de entrenamiento adicionales a su modelo y vuelvan a entrenarlo. Agreguen estos nuevos datos al Anexo 4.1.

Figura 4. Pasos para descargar el modelo entrenado



Mientras entrena el modelo, discutan como grupo: ¿Les pareció útil la participación de sus compañeros en la validación de tu modelo? ¿Por qué?

Acaban de utilizar las instrucciones de varias personas para enriquecer su modelo. Al ser personas diferentes, probablemente se les hayan ocurrido formas nuevas de pedir las instrucciones. Esta técnica se usa en diferentes etapas del desarrollo en IA y recibe diferentes nombres dependiendo de la etapa. En términos generales, se le puede llamar **IA colaborativa**, pues varias personas colaboraron en alguna parte del proceso.

Además de enriquecer el modelo gracias a la diversidad de las personas que intervinieron, la **IA colaborativa** puede tener otras ventajas. Se podría solicitar la ayuda del público o de los usuarios por internet, por ejemplo. En este caso, se tendría una mayor diversidad por el acceso a personas con perspectivas diferentes en todo el mundo. Además, se obtiene otra ventaja que es la velocidad y la cantidad de datos. Se podrían obtener una gran cantidad de datos muy rápidamente, ya que se podrían estar recibiendo de varias partes al mismo tiempo.

Asegúrense de tener todos los datos de entrenamiento en el Anexo 4.1, ya que en la próxima sesión continuarán trabajando con ese modelo. De ser posible, para ahorrar tiempo en la próxima sesión, descarguen los ejemplos de cada etiqueta.

Para esto, hagan clic en Volver al proyecto y Entrenar. Luego, en cada etiqueta, en la esquina inferior derecha hagan clic en Descargar y guarden el archivo en un lugar donde lo puedan recuperar en la próxima sesión. En la Figura 4 encuentran la secuencia de instrucciones ilustrada.

Glosario



IA colaborativa: Inteligencia artificial que utiliza la inteligencia colectiva de un gran grupo de personas en diferentes etapas del desarrollo.

Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

Revisa los aprendizajes de la sesión. ¿Crees que lograste alcanzarlos?

- 1 ¿Puedes explicar la relación entre los componentes de una IA de aprendizaje automático y la creación de un modelo?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no
- 2 ¿Puedes explicar en qué consisten las etapas de entrenamiento y validación de un modelo de IA?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no
- 3 ¿Puedes explicar cómo el desarrollo de la IA se beneficia de la participación de personas con perspectivas diferentes?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no

Si tu respuesta a las preguntas anteriores fue “Parcialmente” o “Aún no”, vuelve a revisar los contenidos de la sesión, tratando de buscar correlaciones entre la información que se presenta en la primera parte y la práctica grupal realizada. Discute con tus compañeras y compañeros de grupo las inquietudes que te surjan. Si lo requieres, pide apoyo a tu docente.

Con tu grupo, compartan sus opiniones sobre las siguientes preguntas:



Teniendo presente el reto planteado, ¿para qué les servirá haber validado su modelo usando IA colaborativa?

¿Encuentran alguna relación entre la validación con la IA colaborativa y la posibilidad de que su asistente virtual entienda más instrucciones?

Después del ejercicio realizado, si usan instrucciones ligeramente diferentes, o si le dicen de forma diferente a los objetos, ¿les entiende su asistente?

Hagan un mapa conceptual que conecte las definiciones de palabras y de conceptos claves que has visto en esta sesión. Considera términos como modelo, validación e IA colaborativa.



Sesión 5

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Explicar la confianza de un modelo de IA.

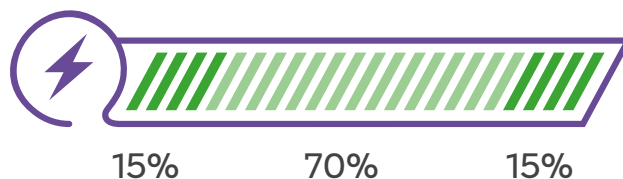


Justificar la utilización de umbrales de acción en un modelo de IA.



Entrenar un modelo de IA y desarrollar un asistente virtual de Scratch que incorpore el modelo entrenado.

Duración sugerida



Material para la clase

- ☐ Anexos 1.2, 5.1 y 5.2
- ☐ Computador con acceso a internet.



Anexo

Anexo 1.2



Los asistentes virtuales son aplicaciones de IA que pueden ayudar con la automatización de diversas tareas e incluso responder a solicitudes específicas de sus usuarios, asemejándose así a lo que haría un(a) asistente personal. Alexa, Siri y el asistente de Google son algunos ejemplos de asistentes virtuales que se han hecho populares hoy en día.

Tu reto consiste en crear un proyecto en Scratch que genere un prototipo de asistente virtual que cumpla con las siguientes especificaciones funcionales:

- 1 Debe realizar cuatro acciones en el escenario de Scratch:
 - a. Prender un radio.
 - b. Apagar un radio.
 - c. Prender una lámpara.
 - d. Apagar una lámpara.
- 2 Debe recibir las instrucciones en texto.
- 3 Debe entender lenguaje natural.
- 4 No debe actuar si no está seguro de lo que le solicitan, o si considera que los efectos de un error son indeseables.

Para lograr lo anterior necesitarás:

- 1 Entrenar un modelo de IA en <https://machinelearningforbirds.com/ai/>.
- 2 Usar verbales de acción.
- 3 Validar tu modelo de IA.
- 4 Implementar tu modelo de IA en un proyecto en Scratch.

Anexo 4.2

Instrucciones de validación		¿Asertó?	Confianza
Prender lámpara			
Apagar lámpara			

Lo que sabemos, lo que debemos saber



Esta sección corresponde al 15% de avance de la sesión

Es el momento de resolver el reto que se encuentra en el Anexo 1.2. Vuelve a leerlo antes de seguir.

En la sesión anterior validaste tu modelo usando la ayuda de varias de tus compañeras y compañeros de clase. Ahora le haremos un ajuste final a ese modelo y luego lo cargaremos en tu asistente virtual Picó en Scratch.

Primero compara brevemente los resultados de la programación tradicional que hiciste en las sesiones 1 y 2 con los modelos de IA, realizados en las sesiones 3 y 4.



¿Qué diferencias observas?

¿Qué pasa cuando le das una instrucción al asistente virtual y este no la entiende o la instrucción dada no tiene nada que ver con las acciones esperadas?

¿Hace algo el asistente virtual o simplemente no hace nada?

Como te habrás dado cuenta, el modelo de IA, por defecto siempre da un resultado.



¿Te parece que es una buena idea que el modelo siempre haga algo?

Si se equivoca, ¿es grave?

¿Se puede hacer algo para evitar que el asistente actúe si se equivoca?

¿Recuerdas que en la sesión pasada se pidió que se registrara un porcentaje en la última columna del Anexo 4.2?

Ese porcentaje nos indica qué tan seguro está el modelo de su propia predicción. A esto se le llama **confianza del modelo**. Evidentemente este número dependerá de los datos de entrenamiento que se hayan usado. Mientras más cerca se encuentre el texto al patrón que el modelo detectó, la confianza será más alta. Observa los porcentajes de tu modelo.



En general, ¿crees que la confianza del modelo es un buen indicador de cuándo se acierta? Si es así, ¿notaste que cuando acierta, la confianza del modelo suele ser alta, mientras que cuando se equivoca, su confianza es baja?

Teniendo en cuenta lo anterior, podrías definir un porcentaje de confianza mínimo para que tu asistente virtual realice la acción, es decir, un umbral de acción. Es más, podrías decidir que dependiendo de la acción se necesite o no un **umbral de acción** o incluso cambiar ese porcentaje.

Aquí no hay una respuesta única. Depende de tu preferencia y de las consecuencias de que tu asistente no te entienda. Por ejemplo, puede que no quieras que el asistente se vaya a equivocar y prenda el equipo de sonido por error, ya que puede ser tarde en la noche y despertarías a las demás personas en tu hogar. En cambio, puede que no te moleste que se equivoque y apague el radio por error. Es más, quizás quieras que sea fácil apagarlo y que no te vaya a “desobedecer” al no entender una instrucción. En ese caso podrías usar un umbral de acción para prenderlo y no usar un umbral de acción para apagarlo.

En esta parte de la sesión vas a trabajar en grupos. Reúnete en grupos de 2 a 3 personas, según las indicaciones de tu docente y discutan sobre el uso o no de umbrales de acción para cada instrucción, los valores que tendrían en caso de usarlos y la justificación de sus decisiones.

También deben definir con su grupo qué hacer en caso de usar y alcanzar un umbral de acción y en caso de que no se alcance. Por ejemplo, volviendo al caso de prender la radio, supongan que usan un umbral de acción de 80% y que una instrucción que le dieron obtuvo un 75% de confianza. ¿Qué debería hacer Picó? Quizás pueda decir no te entendí, por ejemplo.

Anexo

Anexo 5.1

Inst.	Umbral	Acción si no se alcanza el umbral	Justificación
Predicción			
Predicción			
Predicción			
Predicción			

Llenen el Anexo 5.1 con las conclusiones a las que lleguen como grupo. En la segunda columna escriban el porcentaje del umbral de acción si decidieron usarlo, o escribe “no” en caso de que no lo usen en esa instrucción. En la tercera columna escriban la acción que el asistente virtual deberá realizar en caso de que la confianza de la predicción no alcance el umbral de acción. En la última columna debe estar la justificación de todas las decisiones relacionadas con esa instrucción.

Glosario



Confianza del modelo: Evaluación interna del modelo de la exactitud de su predicción.



Umbral de acción: Valor mínimo de confianza del modelo para autorizar la realización de la acción predicha.

Manos a la obra

Conectadas

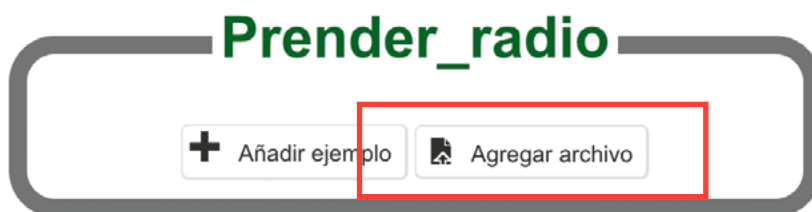
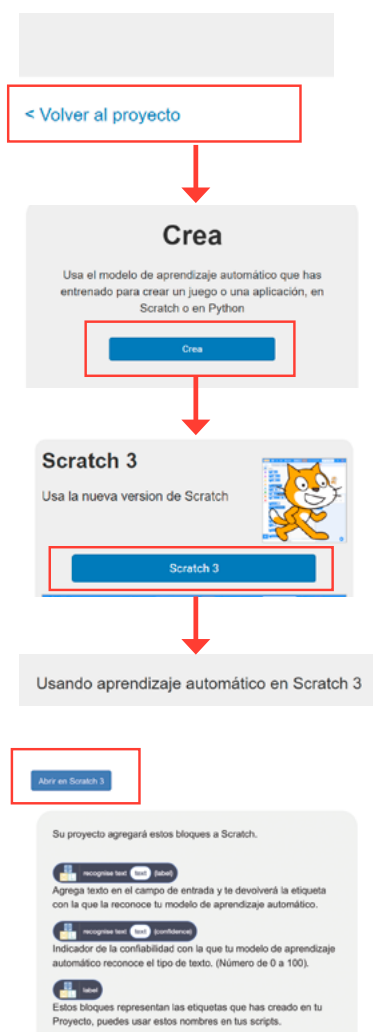


Esta sección corresponde al 85% de avance de la sesión

Ahora que su grupo ya validó su modelo, que lo reentrenaron, de ser necesario, y que fijaron umbrales de acción de considerarlos necesarios, su modelo está listo para cargarse en Scratch y ser usado por Picó.

Para esto, deben volver a crear un proyecto de reconocimiento de texto en el siguiente enlace: <https://machinelearningforkids.co.uk/>, tal como lo hicieron en las dos sesiones anteriores. Van a volver a crear las cuatro etiquetas correspondientes a las cuatro acciones que su asistente virtual debe realizar. De ser posible, carguen en cada una el archivo con los ejemplos que guardaron en la sesión anterior haciendo clic en Agregar archivo en la parte de abajo. De lo contrario escriban los ejemplos que registraron en el Anexo 4.1. Luego entrenen su modelo.

Figura 1. Pasos para cargar el modelo entrenado a Scratch



Ahora, hagan clic en Volver al proyecto en la esquina superior derecha. Luego seleccionen Crea, Scratch 3. En esta pantalla se presentan los bloques que se añadirán a Scratch para utilizar el modelo que acaban de entrenar. Revísenlos y finalmente hagan clic en Abrir en Scratch 3. En la Figura 1 encuentran la secuencia de instrucciones ilustrada.

Una vez en Scratch carguen el proyecto del archivo *Controlar_todo.sb3*, tal como lo hicieron en la Sesión 1. Les debe aparecer el escenario de la habitación y Picó, la lámpara y el equipo de sonido, cada uno con su código tradicional usando condicionales simples y una sola instrucción.

Como grupo deben cambiar el código para utilizar su modelo e implementar los umbrales de confianza y las acciones en caso de que no se alcance el umbral. En el Anexo 5.2 encuentran un ejemplo de implementación de un modelo de IA y un umbral de acción del 70% para prender el radio. En caso de que no se alcance el umbral, Picó dice "Disculpa, no te entendí."

Ahora es su turno. Después de terminar, prueben su asistente. Si les queda todavía algo de tiempo antes del cierre de la clase, lo ideal sería repetir el ejercicio de la sesión anterior para validar este nuevo modelo y poder comparar mejor los resultados.

Anexo

Anexo 5.2



Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

Revisa los aprendizajes de la sesión. ¿Crees que lograste alcanzarlos?

1 ¿Puedes explicar la confianza de un modelo de IA?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

2 ¿Puedes justificar la utilización de umbrales de acción en un modelo de IA?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

3 ¿Puedes entrenar un modelo de IA y desarrollar un asistente virtual de Scratch que incorpore el modelo entrenado?

- ☐ Sí
- ☐ Parcialmente
- ☐ Aún no

Vuelve ahora al reto que tenías.



¿Cómo te fue dándole solución?
¿Qué fue difícil para ti? ¿Por qué?

Piensa igualmente en las siguientes preguntas:

¿Ves alguna relación entre la confianza del modelo, el umbral de acción y la posibilidad de que tu asistente virtual realice menos acciones equivocadas?

Si usas instrucciones ligeramente diferentes, o si le dices de forma diferente a los objetos, ¿te entiende tu asistente?

¿Qué pasa si te equivocas en una letra de la instrucción?

¿De cuántas formas le puedes pedir que haga cada una de las acciones? ¿Crees que hay un límite?

¿Te parece que ya es lo suficientemente inteligente tu asistente virtual? En caso negativo, ¿qué le faltaría?

Imagina que te han invitado a hacer una publicación para un canal de videos educativos breves, tipo TikTok, y esperan que les cuentes lo que aprendiste con esta práctica sobre la IA. Prepara el guion de lo que dirías en tu video. Recuerda que máximo deberías hablar por 1 minuto y que la idea es que menciones las conclusiones más importantes que sacaste, por ejemplo, con relación a la confianza del modelo, el umbral de acción y la importancia e incluso necesidad de los umbrales de acción para evitar resultados incorrectos.



Anexo 1.1 Instrucciones únicas

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Acertó?
Prender lámpara			
Apagar lámpara			

Anexo 1.1 Instrucciones únicas

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Acertó?
Prender radio			
Apagar radio			

Anexo 1.2 Reto

Los asistentes virtuales son aplicaciones de IA que pueden ayudar con la automatización de diversas tareas e incluso responder a solicitudes específicas de sus usuarios, asemejándose así a lo que haría un(a) asistente personal. Alexa, Siri y el asistente de Google son algunos ejemplos de asistentes virtuales que se han hecho populares hoy en día.

Tu reto consiste en crear un proyecto en *Scratch* que genere un prototipo de asistente virtual que cumpla con las siguientes especificaciones funcionales:

- 1 Debe realizar cuatro acciones en el escenario de *Scratch*:
 - a. Prender un radio.
 - b. Apagar un radio.
 - c. Prender una lámpara.
 - d. Apagar una lámpara.
- 2 Debe recibir las instrucciones en texto.
- 3 Debe entender lenguaje natural.
- 4 No debe actuar si no está seguro de lo que le solicitan, o si considera que los efectos de un error son indeseables.

Para lograr lo anterior necesitarás:

- 1 Entrenar un modelo de IA en <https://machinelearningforkids.co.uk/>
- 2 Validar tu modelo de IA.
- 3 Usar umbrales de acción.
- 4 Implementar tu modelo de IA en un proyecto en *Scratch*.

Anexo 2.1 Instrucciones en lista

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Acertó?
Prender lámpara			
Apagar lámpara			

Anexo 2.1 Instrucciones en lista

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Acertó?
Prender radio			
Apagar radio			

Anexo 3.1 Patrones

Nombres: _____

1 ¿Qué número creen que sigue en la secuencia: 4, 8? _____

2 ¿Por qué? _____

3 ¿Es la única respuesta que se ajusta a la secuencia? _____

4 ¿Qué se necesita para poder estar seguros del patrón en este caso?

5 Completen la siguiente tabla:

Secuencia	Siguiente valor	Patrón
1, 4, 7, 10		
123, 117, 108, 99		
13, 27, 55, 111		
2, 8, 26, 80, 242		

6 Separa estas palabras en dos grupos escribiéndolas en las dos casillas de la primera fila de la tabla que aparece abajo: molesta, mal, malgeniado, enfurecida, triste, cansado, feliz, exaltada, emocionado, encantada, contento y enérgica.

7 Luego escribe el patrón, lo que tiene en común cada grupo en la segunda fila.

Anexo 3.2 IA entrenada con aprendizaje automático supervisado

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Acertó?
Prender lámpara			
Apagar lámpara			

Anexo 3.2 IA entrenada con aprendizaje automático supervisado

	Datos de entrenamiento Instrucciones programadas	Datos de validación Instrucciones de validación	¿Acertó?
Prender radio			
Apagar radio			

Anexo 4.1 Datos de entrenamiento – parte A

	Instrucciones programadas
Prender lámpara	
Apagar lámpara	

Anexo 4.1 Datos de entrenamiento – parte B

	Instrucciones programadas
Prender radio	
Apagar radio	

Anexo 4.2 Datos de validación – parte A

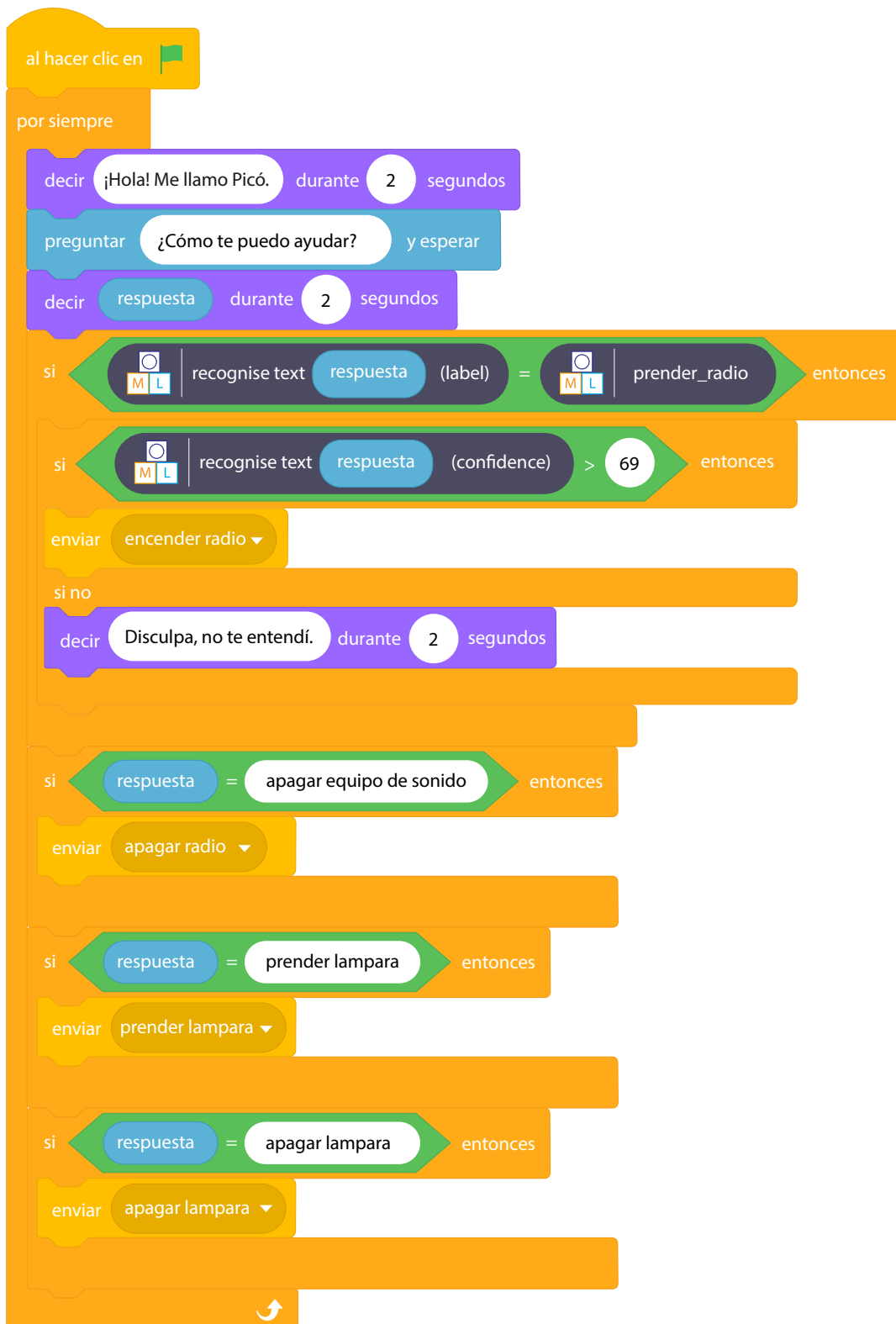
	Instrucciones de validación	¿Acertó?	Confianza
Prender lámpara			
Apagar lámpara			

Anexo 4.2 Datos de validación – parte B

	Instrucciones de validación	¿Acertó?	Confianza
Prender radio			
Apagar radio			

Anexo 5.1 Umbrales de acción

Instr.	Uso umbral	Acción si no se alcanza el umbral	Justificación
Prender lámpara			
Apagar lámpara			
Prender radio			
Apagar radio			

Anexo 5.2 Ejemplo de implementación de modelo de IA en Scratch



TIC



Apoya:



Educación

