

“Who’s there?”

Arenas Ayala, Ramón
Lugo Cano, Daniel
Millán Pimentel, Oscar Fernando

May 25, 2020

1 Introducción

La conciencia humana es algo que a la fecha no se ha podido replicar. En gran parte lo que conocemos hoy como inteligencia artificial se debe a este término: **conciencia**. Gracias a la pregunta hecha por Alan Turing en 1950¹: “¿Puede una computadora hablar como un humano?” es que tenemos, en gran parte, el área de Inteligencia Artificial en su apogeo y por ello ahora presentamos este reporte en el que tratamos de responder esta pregunta con ayuda de la implementación de un bot de conversación.

2 Objetivo

El objetivo tanto de este reporte como de la implementación que lo acompaña es crear un bot de conversación que resuelva un problema en específico, el cual será definido más adelante, además de clarificar el poder del lenguaje y, junto con ello, los numerosos contextos. También, nos daremos cuenta de lo complejo que puede ser una conversación entre humanos en este caso específico, entre una computadora y un humano.

3 Chatbot

Citando la definición de Oracle México²: “un chatbot es un programa informático que simula y procesa conversaciones humanas (ya sea escritas o habladas) permitiendo a los humanos interactuar con dispositivos digitales como si se estuvieran comunicando con una persona real [...]”, lo que podemos tomar de esta definición es que un chatbot trata siempre de imitar el comportamiento humano en el ámbito de expresión del lenguaje. Dependiendo el objetivo del chatbot podemos decir que puede ayudar a los humanos a realizar una tarea o simplemente funcionar como una persona con la que los humanos podamos platicar de cualquier tema, todo depende de la complejidad del bot y las tareas que la programadora o el programador decida sean las correctas para que el chatbot resuelva.

¹Will we ever... pass the Turing Test for computers? [1]

²¿Qué es un chatbot? [2]

4 Problema

De acuerdo con el *Foro Económico Mundial*, en 2019 hubo una afluencia mundial de 1.45 billones de turistas alrededor del mundo³, además, este número incrementa cada año, por lo que podemos decir que viajar es una tendencia que, por lo menos en esta época, está a la alza y seguirá de esa manera. Por ello, siempre que planeamos un viaje, ya sea pequeño, en carro, por carretera a un estado fuera de la ciudad de México, hasta un viaje en avión a otro continente, se necesita cierta planeación y durante ese momento de organización vienen a la mente preguntas como: ¿Cuánto dinero debo llevar?, ¿qué hoteles estarán disponibles?, ¿qué tal esta la comida por ese lugar?, ¿a donde debo ir si quiero un lugar con música en vivo?, etc. Mil preguntas pueden surgir dependiendo de los intereses de la o las personas que realizarán el viaje, o incluso, ya estando en su destino turístico se le pueden presentar esas dudas o preguntas. Esa es la problemática que resolveremos en nuestro chatbot.

4.1 ¿Por qué usar un chatbot?

Hoy en día, tanto el Internet como el uso de las redes sociales se ha vuelto, afortunada y desafortunadamente, indispensable para una gran parte de la población mundial, ya que contamos con estas herramientas lo mejor sería aprovecharlas, por eso, un chatbot que se puede usar en herramientas como **Whatsapp** y **Facebook Messenger**, que hasta mayo de 2019 se encontraban en el top 5 de aplicaciones más usadas a nivel mundial⁴ es la mejor opción. Ahora, teniendo como base esta información, nos hace pensar que implementar un chatbot para que su uso sea a través de alguna de estas dos aplicaciones es lo más sensato, de esta manera podemos llegar a una gran parte de la población que viaja dentro de sus mismos países o al extranjero, además uno de los elementos esenciales que no puede faltar en un viaje, hoy en día, es el celular, otra razón por la cual es buena idea la solución de la problemática de esta forma.

4.2 Alcance

El alcance de este chatbot abarca una parte muy pequeña de la problemática presentada, se decidió que solamente serán destinos turísticos mexicanos, específicamente algunas playas de México: Huatulco, Cancún y Acapulco. El chatbot podrá resolver dudas y realizar acciones como: recomendaciones de comida, recomendaciones de restaurantes, recomendaciones de actividades, recomendaciones de atractivos turísticos, dar el clima del destino seleccionado, dar información de hospedaje y del vuelo al destino. Todo esto basándonos en las tareas que hace una agencia de viajes pero con un enfoque más personal. Por lo tanto el usuario que use el chatbot podrá preguntar dudas acerca de sus vacaciones en cualquier momento de su estadía en dicho lugar.

5 Tipo de agente

Estamos hablando de un **agente lógico** también conocido como agente basado en conocimiento⁵. Esto se debe a que la implementación tiene una **base de conocimiento** con la cual podrá responder a las entradas que le dé el usuario, siguiendo un procedimiento para realizar una respuesta acorde a la entrada recibida, esto para que el usuario sienta que es una conversación fluida y lo más parecida a una conversación humano-humano. Para ello es muy importante que la implementación cuente con **inferencias**, que no es más que realizar preguntas o sentencias basadas en sentencias anteriores.

³How Many People will Travel the World? [3]

⁴Las redes, webs y apps más usadas del mundo [4]

⁵Agentes lógicos [5]

Nuestro chatbot (agente) es capaz de representar estados y acciones, además, puede deducir qué respuesta es apropiada a la pregunta o petición que su entrada reciba, por ello decimos que se trata de un agente basado en conocimiento.

6 Entorno de trabajo

Entorno de Trabajo		Justificación
Observable	Parcial	Es parcial al ser un agente que necesita de una segunda parte para funcionar y de esta parte, al ser un usuario humano, se espera que siempre esté recibiendo sentencias, lo cual no siempre sucederá. Habrá veces en las que el usuario no escriba nada al chatbot, mientras no se tenga una sentencia para leer por parte del bot, no se puede dar una respuesta.
Determinista	Determinista	Es determinista ya que el próximo estado del agente está determinado inmediatamente después del estado anterior, es decir, siempre tendrá una respuesta para cualquier sentencia introducida por el usuario.
Episódico	Episódico y secuencial	Es episódico cuando se hacen preguntas o consultas al chatbot para las que su respuesta es inmediata, por ejemplo, cuando se da la bienvenida al usuario la respuesta será con otra bienvenida, esto no afectará a estados futuros del chatbot. Se vuelve secuencial cuando se entra en un <i>bloque</i> , es decir, cuando un intent esta dentro de otro, por ejemplo, cuando el usuario quiere hacer una reservación de hotel se entra a otro intent para saber cuántas personas se hospedarán, luego entra a otro para verificar en qué hotel se quiere hospedar; el último intent de ese bloque depende totalmente de que el usuario antes haya preguntado por hacer una reservación, por ello se vuelve secuencial.
Discreto	Discreto	Es discreto ya que la base de conocimiento del chatbot es finita, tiene un determinado número de acciones y percepciones, es decir, las implementadas por las y los programadores.
Agente	Multiagente	Es claro ver que para tener una conversación se necesitan 5 factores indispensables que la lleven a cabo: un emisor, un receptor, un medio de comunicación, un mensaje y un contexto. De estos 5 elementos destacamos el emisor y receptor, sin los cuales no se podría llevar a cabo la conversación del chatbot, por ello hablamos de que nuestro agente es multiagente.

Table 1: Entorno de trabajo⁶.

7 REAS

Existen distintos tipos de sistemas que utilizamos día a día, los cuales poseen características llamadas REAS que es un acrónimo para **R**endimiento, **E**ntorno, **A**ctuadores y **S**ensores; en nuestro

⁶Properties of task environments[6]

proyecto se pueden ver en la siguiente tabla:

REAS	
Agente	Chatbot de conversación
Rendimiento	La prueba de Turing ⁷
Entorno	Plataforma con función de chat
Actuadores	Base de conocimiento
Sensores	Entrada del humano (sentencias)

Justificación

Ahora describiremos el porqué nuestra REAS se definió de esta manera, basándonos en las definiciones de Russell y Norvig⁸

- **Rendimiento:** Se eligió esta medida de rendimiento porque lo único que evalúa si nuestro chatbot es bueno o malo es el mismo usuario. Si tiene una conversación fluida, con respuestas acertadas de acuerdo a las preguntas realizadas por el mismo usuario, significa que el usuario pensará que está hablando con otro humano.
- **Entorno:** El entorno es donde se desenvuelve el agente, es decir, el único lugar donde puede interactuar el usuario con el chatbot es en una plataforma tipo chat, donde se pueda tanto recibir como enviar texto. En nuestro caso el entorno es una terminal de computadora, aquí el usuario interactuará con el bot y viceversa.
- **Actuadores:** La base de conocimiento es parte esencial para el bot ya que todo su comportamiento se basa en este elemento, se eligió como actuador porque sin la base de conocimiento el bot no podría funcionar, no daría respuesta alguna, no sería capaz de interactuar.
- **Sensores:** Las cadenas de texto que el usuario introduzca en el entorno del agente serán “leídas” por el bot, esto activará una respuesta que tomará de la base de conocimiento y si está bien implementado dará una respuesta adecuada a la entrada del usuario.

8 Representación del conocimiento

8.1 Esquema

La forma en la que representamos el conocimiento consta de varias partes la primera es una clasificación por **intención** de la sentencia introducida por el usuario, por su traducción en inglés en el código lo llamamos **intent** esto se refiere a qué es lo que quiere llegar el usuario con la sentencia dada, por ejemplo:

Usuario: Quiero una pizza grande de pepperoni. (1)
intent → “ordenar pizza”

Esto nos dice que con la sentencia (1) el usuario quiere pedir una pizza, no nos interesa (aún) cuánto cuesta, dónde se entrega, etc. Simplemente se sabe que se quiere ordenar una pizza.

⁸ Agente inteligente, REAS y Entorno de trabajo [8]

Lo siguiente que compone nuestro esquema de conocimiento es una expresión regular esto es de utilidad ya que no queremos atrapar casos redundantes como artículos o preposiciones del lenguaje, buscamos palabras clave dentro de la sentencia del usuario, por ejemplo:

Usuario: (Qué||Cuál||Dime||Dame).* (categorías).*

Queremos atrapar el **intent**: “dar categorías”, por ello solo necesitamos saber qué es una pregunta o una petición y la palabra clave *categorías*, de esta manera quitamos casos innecesarios para que el compilador atrape y de paso facilitamos la búsqueda y acortamos líneas de código basura.

La última parte que conforma nuestro esquema es la **respuesta**, que no tiene gran complejidad, simplemente es una lista de **strings** que el bot responderá de manera aleatoria, esto funciona porque estarán dentro del mismo **intent**, solo basta asegurarse de que las respuestas sean igualmente adecuadas para la sentencia que atrapa el compilador.

9 Actividades del chatbot

Como ya se dijo anteriormente, el chatbot sigue un esquema de **intents** y de expresiones regulares, por lo que es sencillo ubicar qué actividades puede hacer el chatbot si prestamos atención a las intenciones que pretende realizar dependiendo de la sentencia recibida por el usuario. Hablando específicamente de nuestro chatbot, las actividades que puede realizar son las comunes de una agencia de viajes combinada con un asistente personal, es decir, puede realizar las siguientes actividades, incluyendo qué expresiones regulares pueden caer en cada una de estas actividades:

- Dar la bienvenida
Expresiones regulares: ‘Hola (.*)’, ‘.*Hola.*’, ‘Hola’, ‘.*Buen(a|o)s (días|tardes|noches).*’
- Ayudarte a reservar hotel en los destinos disponibles
Expresiones regulares: ‘Quiero (.*) hotel (.*)’, ‘Quisiera (.*) hotel (.*)’
- Ayudarte a revisar costos de vuelo y hospedaje
Expresiones regulares: ‘(precio|costo) .* vuelo .*’, ‘(Cuanto|Cuánto) .* vuelo .*’, ‘(Cuanto|Cuánto) .* vuelo .*’, ‘(precio|costo) .* vuelo .*’
- Ayudarte a revisar el clima en el destino seleccionado
Expresiones regulares: ‘(Qué|Cuál|Dime|Dame|Cuáles|Cuales|Que|Cual|Quiero|A que| A qué).* (clima|tiempo).*’, ‘.* clima .*’
- Decir qué hoteles hay disponibles
Expresiones regulares: ‘(Que|Qué|Cuáles|Cuales|Quiero|Quisiera|Dime) (.*) hoteles (.*)’
- Dar recomendaciones de comida del destino
Expresiones regulares: ‘.* que .* (comer|comida) .*’, ‘.* comida tipica .*’, ‘.* comida .*’, ‘.* comida .* huatulco’, ‘.* comer .*’

- Dar recomendaciones de restaurantes en el destino
Expresiones regulares: ‘.* (restaurante|restaurantes) .*’, ‘.* (restaurantes|restaurante) .*’, ‘(Donde|Dónde|En que lugar|En qué lugar|Algún lugar|Algún lugar) .* comer .*’
- Dar recomendaciones de qué actividades hacer en ese destino
Expresiones regulares: ‘(Que|Que) actividades.*’, ‘(Que|Qué) puedo hacer en .*’, ‘(Que|Que) actividades .*’, ‘(Que|Qué) .* hacer .*’, ‘(Que|Qué) .* hacer .*’

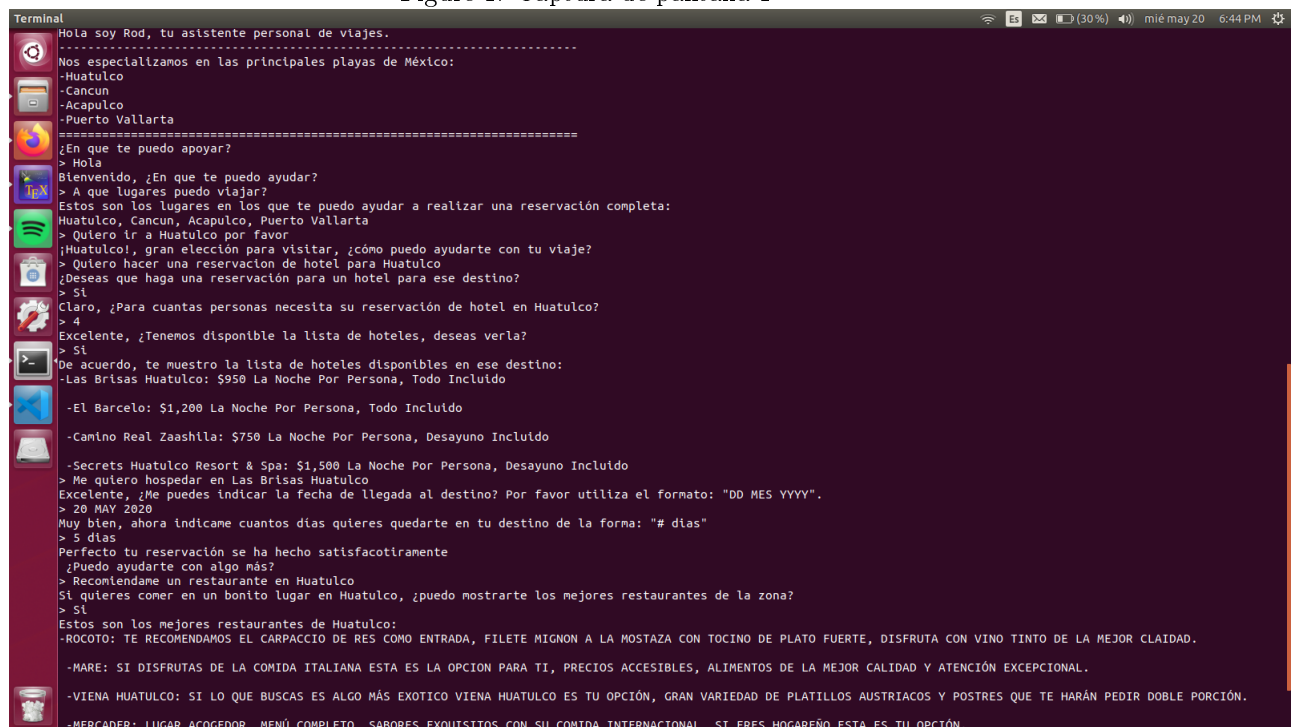
Cuando se ve (.*) o .* en la expresión regular significa que no nos interesa que haya palabras en esa parte de la sentencia. Cada expresión regular diferente está dividida por comas (,) y encerrada por comillas simples (‘’).

Se puede agregar tanta información del lugar como lo necesite el programador, la base de conocimiento puede ser, de igual forma, tan extensa como se necesite.

10 Flujo de diálogo

A continuación se muestra una pequeña ejecución del chatbot, donde se describirán los *intents* a los que corresponden las sentencias introducidas.

Figure 1: Captura de pantalla 1



El bot comienza dando una pequeña bienvenida diciendo con qué destinos cuenta para que el usuario pregunte por cosas relacionadas a un viaje a esos destinos. A continuación se presentan las sentencias utilizadas:

- *Hola*
`intent` = bienvenida
Explicación: La expresión regular que atrapa este caso es cualquiera que tenga que ver con un saludo, por ejemplo: hola, buenas tardes, buenos días. Además, atrapa expresiones que contengan únicamente la palabra “hola”.
 - *A que lugares puedo viajar?*
`intent` = dar destinos
Explicación: Esta sentencia cae en este caso cuando se atrapa de la forma: (Qué|Cuál|Dime|Dame|Cuáles|Cuales que|A qué).* (destinos|lugares).* cuando el usuario quiere saber qué destinos están disponibles para poder viajar.
 - *Quiero ir a Huatulco por favor*
`intent` = ir huatulco
Explicación: La sentencia cae en este intent cuando se selecciona un lugar para viajar.
 - *Quiero hacer una reservación de hotel para Huatulco*
`intent` = reservar hotel
Explicación: Si el usuario quiere realizar una reservación para el destino seleccionado, entonces cae en este caso. A lo que el bot le pregunta:
return: ¿Deseas que haga una reservación para un hotel para ese destino?
A lo cual el usuario debe responder la pregunta de manera afirmativa o de manera negativa, en este caso contesta de manera positiva a la pregunta y este `intent` va relacionado directamente con el siguiente `intent` que es ‘reservar hotel personas’ que va encaminado a saber precisamente para cuántas personas es la reservación.
 - Si
`intent` = confirmación
Explicación: Cada vez que se ve un “sí” o “no” van al mismo `intent` pero va a depender el **contexto** cuál será la respuesta que dará el bot
 - 4
`intent:` reservar hotel personas
Explicación: el bot caerá en este caso cuando se detecte un número inmediatamente de haber pasado por el `intent` pasado ya que forman un bloque de respuestas dentro de la implementación.
- Al afirmar o negar siempre se tendrá una respuesta por parte del bot. Lo que sigue es que el bot pregunta si el usuario desea ver la lista de hoteles disponibles en esa zona, al contestar de manera afirmativa se muestra la lista de hoteles, esto correspondiente al `intent:` `que hotel`, al indicar el nombre del hotel donde se desea hospedar tenemos los siguientes resultados:
- **Me quiero hospedar en Las Brisas Huatulco**
`intent:` que hotel
A continuación se pregunta por la fecha de llegada al destino y cuántos días quiere hospedarse el usuario, de la siguiente manera:

- 20 MAY 2020
`intent: hotel fecha inicio`
Explicación: Toma exactamente ese formato de fecha, no puede leer otro diferente, este se activa una vez que dices en qué hotel te quisieras hospedar.
- 5 días
`intent: hotel fecha fin`
Explicación: al detectar la palabra “días” se activa este intent, que viene anidado con el intent anterior.
- Recomiendame un restaurante en Huatulco
`intent: recomendaciones restaurante`
Explicación: Cuando el usuario pide recomendaciones sobre algún lugar para comer por la zona, el bot detecta la palabra *restaurante*, *lugar*, *huatulco*, así puede preguntar si el usuario se refiere a ver restaurantes por la zona, a lo que el usuario responde que sí, a continuación muestra la recomendación además de la lista de algunos lugares.

Se puede tomar como ejemplo el destino a Huatulco, el flujo de diálogo será siempre el mismo, lo único que cambia es el destino e información referente al mismo. Por ello tenemos **intents** generales, el usuario podrá elegir el destino de su preferencia desde el inicio del diálogo y a partir de ahí toda la información que pida será de ese destino, a menos que especifique que quiere información de otro destino.

Aclaración

La información presentada acerca del lugar, como restaurantes, hoteles, comida típica y actividades recreativas, es verídica y puede ser consultada en: <https://www.tripadvisor.com.mx>

11 Análisis de comportamiento

El chatbot se comporta como un asistente personal para viajar, realiza todas aquellas tareas que una persona necesita hacer cuando va de viaje a algún lugar, por lo tanto el tema de las palabras clave para que las expresiones regulares sean atrapadas es con base al tema de viajes. Por ejemplo atrapa expresiones que contengan palabras como: **comida**, **restaurante**, **viajar**, **vuelo**, **precio**, **hotel**, **hospedaje**, **clima**, por mencionar algunos. No se encuentra dentro del chatbot respuestas ajenas a este tema, si se da el caso tenemos un `intent: desconocido` para dar a entender al usuario que no sabe nada sobre la sentencia introducida, que por favor reescriba su pregunta o si lo puede decir con otras palabras.

12 ¿Qué tan adecuada es la respuesta del chatbot?

Muchos de los `intent` se basan en confirmación, es decir, el bot vuelve a preguntar, de alguna forma, si se refería a cierto tema, a esto el usuario contesta afirmativa o negativamente y es cuando se da la respuesta a las sentencias del bot. Como se mencionó en el análisis de comportamiento,

el chatbot entiende las sentencias encaminadas al tema de **viajar**, por ese lado el chatbot contesta correctamente a todas las preguntas referentes a recomendaciones durante su viaje, reservaciones, vuelos, comida, actividades, etc. A fin de cuentas quien juzgará qué tan adecuada es la respuesta del bot es la usuario que la use cuando sea el momento.

13 Ventajas del agente

Las ventajas de usar un agente basado en conocimiento es que, al contar con una base de conocimiento, lo vuelve capaz de contener tanta información como se le agregue, siendo parte esencial de este agente, es el núcleo del agente. Otra ventaja es que este agente tiene la capacidad de aprender del ambiente en el que se desenvuelve, crear acciones nuevas y respuestas nuevas si así lo ve necesario, es claro que este chatbot aún no es capaz de hacerlo, pero por el simple hecho de ser un agente basado en conocimiento se podría implementar el aprendizaje más adelante.

14 Desventajas del agente

Una desventaja que se ve a simple vista, tiene que ver con la base de conocimiento, al ser un agente que aún no cuenta con aprendizaje autónomo, se vuelve muy limitada la base de conocimiento. Quien programe un agente de este tipo sin el aprendizaje tendrá que agregar suficientes elementos a la base de conocimiento para que el agente sea robusto y pueda dar el resultado esperado. Otra desventaja relacionada con lo anterior es la programación de funciones generalizadas que te lleven a un mismo resultado por parte del agente al ambiente, es decir, si se desea agregar conocimiento nuevo y las funciones son específicas de un solo estado o una sola acción, el agregar nuevo conocimiento se vuelve repetitivo y tedioso.

15 Conclusión

El implementar un agente basado en conocimiento, en este caso el chatbot, requiere de conocimiento no solo en el lenguaje de programación en el que se esté implementando, sino requiere conocimiento en la lengua, en este caso el español, ya que extraer de una oración o sentencia un significado exacto puede llegar a ser complicado. Se puede ver como referencia una simple conversación entre dos personas, una hablante nativa de inglés y la otra de español, si no están sincronizados usando el mismo lenguaje es muy difícil entablar la conversación o saber qué es lo que necesita la otra persona.

Más adentrado a nuestra implementación el bot contesta de manera adecuada a un cierto grupo de sentencias que tengan que ver con el tema con el que fue implementada su base de conocimiento, con viajes. Por lo tanto si el objetivo de un usuario es comprar su despensa y le pide ayuda a nuestro chatbot, es claro que ninguna respuesta será adecuada para ese usuario, haciendo énfasis en que depende mucho en qué es lo que busca el usuario del chatbot. Si busca información acerca de destinos para vacacionar y todo lo que conlleva el realizar un viaje, el chatbot contesta de manera adecuada, se hizo pensando en que el usuario pudiera cometer faltas de ortografía y aún así saber que se refiere a un mismo **intent**.

16 Ejecución de implementación

Se creó con el lenguaje de programación Python y es ejecutado por medio del intérprete de Python3 usando el comando: `python3 chatbot.py`.

17 Bibliografía

References

- [1] Pavlus, J. (2014, noviembre 18). Will we ever... pass the Turing Test for computers? Recuperado 5 de mayo de 2020 15:16, de <https://www.bbc.com/future/article/20120516-can-computers-ever-think-like-us>
- [2] Oracle. (s. f.). ¿Qué es un chatbot? Recuperado 5 de mayo de 2020 15:28, de <https://www.oracle.com/mx/solutions/chatbots/what-is-a-chatbot/>
- [3] Traveling Cook. (s. f.). How Many People will Travel the World? Recuperado 5 de mayo de 2020 16:31, de <https://traveling-cook.com/how-many-people-will-travel-the-world/>
- [4] ReasonWhy. (2019, marzo 4). Las redes, webs y apps más usadas del mundo. Recuperado 5 de mayo de 2020 18:44, de <https://www.reasonwhy.es/actualidad/redes-webs-apps-mas-usadas-2019>
- [5] Norvig, P. & S. J. Russell (2009). Artificial Intelligence: A Modern Approach 3rd edition. Capítulo 7, sección 7.1: Knowledge based agents (3.a ed., Vol. 1, pp. 219-220). Recuperado de <https://www.cin.ufpe.br/~tfl2/artificial-intelligence-modern-approach.9780131038059.25368.pdf>
- [6] Russell, S., & Norvig, P. (2020). Artificial Intelligence: A Modern Approach (4th Edition) [PDF] (4.a ed.). Capítulo 2, sección 2.3.2: Properties of task environments. (pp. 42-44) Recuperado de <https://www.cin.ufpe.br/~tfl2/artificial-intelligence-modern-approach.9780131038059.25368.pdf>
- [7] TED-Ed. (2016, abril 25). The Turing test: Can a computer pass for a human? - Alex Gendler [Archivo de vídeo]. YouTube. Recuperado de <https://www.youtube.com/watch?v=3wLqsRLvV-c> el 20/mayo/2020 18:04
- [8] Russell, S., & Norvig, P. (2009). Artificial Intelligence: A Modern Approach 3rd edition. Capítulo 2, sección 2.3: The Nature of Environments (3.a ed., Vol. 1, pp. 40-41). Recuperado de <https://www.cin.ufpe.br/~tfl2/artificial-intelligence-modern-approach.9780131038059.25368.pdf>