

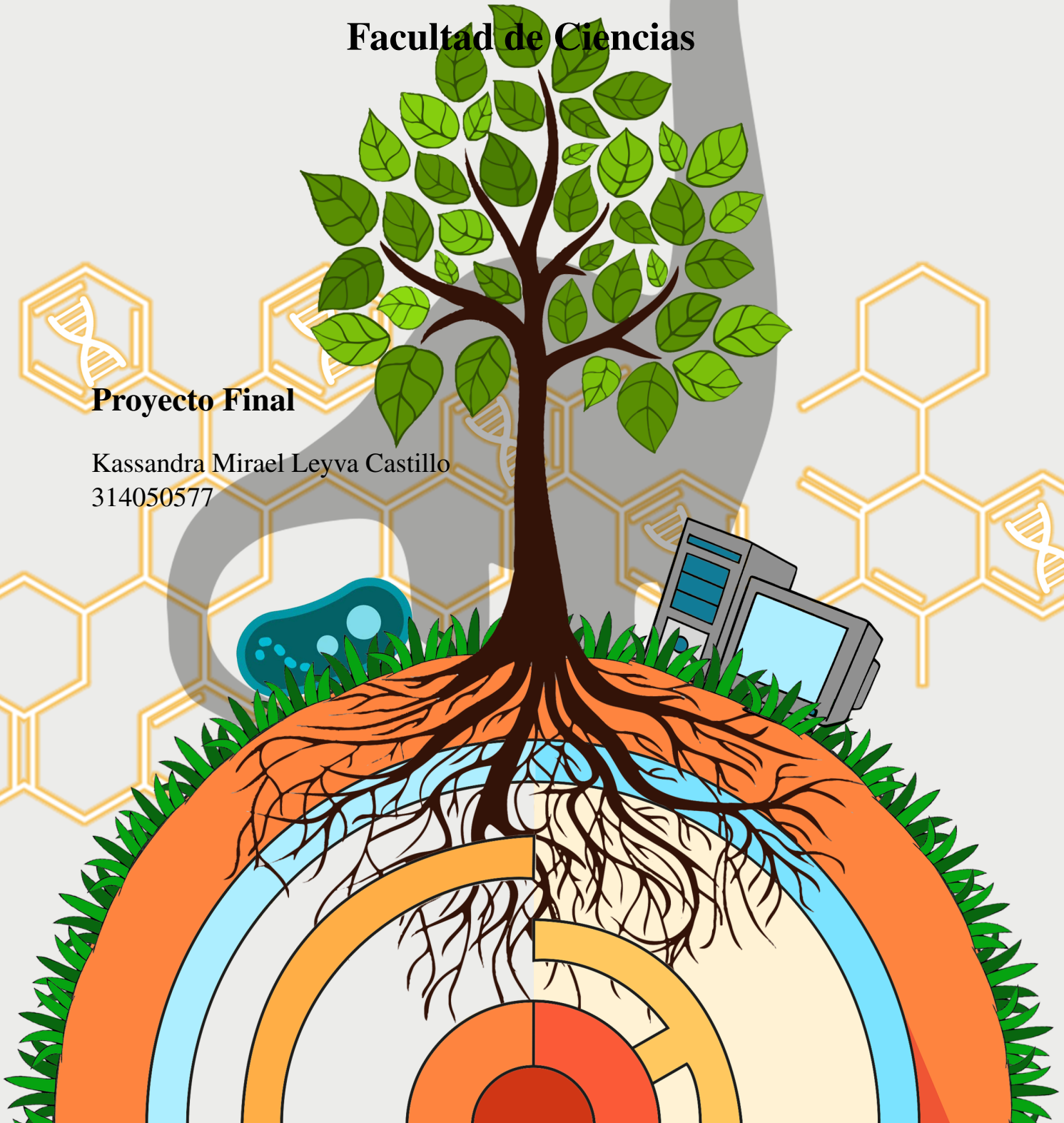


# Universidad Nacional Autónoma de México

## Facultad de Ciencias

### Proyecto Final

Kassandra Mirael Leyva Castillo  
314050577



# Introducción

Los sistemas de recomendación personalizan la experiencia del usuario al ofrecerles un artículo. Es decir, dependiendo del tipo de giro de una entidad, el sistema ofrecerá artículos similares, haciendo que la experiencia del usuario sea placentera.

Así pues, los Sistemas de Recomendación son algoritmos que intentan "predecir" los artículos que un usuario querrá adquirir.

Anteriormente, nos guiábamos por el ranking. Obteníamos el objeto con mejor ranking, nos dejábamos guiar por la acción popular, sin embargo, hoy en día gracias al Machine Learning, nos podemos alejar del montón y poder ir directo a lo que queremos, así podemos obtener un objeto que se ajuste a nuestro parecer.

Éstos algoritmos usan distintas estrategias, por ejemplo:

- **Popularidad.** Recomienda por la popularidad, es lo mismo que se ha hecho desde hace tiempo. De acuerdo al ranking se ofrece al usuario sin igual, sin mirar lo que se quiere en verdad.
- **Contenido.** Con respecto a lo que el usuario ve, se intenta predecir que busca el usuario y ofrecer similitudes.
- **Colaborativo.** Se utiliza información en masas para identificar perfiles similares y aprender de los datos para recomendar productos de manera individual.

De acuerdo al sistema de recomendación con una técnica colaborativa buscando la satisfacción del usuario, de acuerdo a información que se va aprendiendo podrá dar una recomendación. Ahora bien podemos encontrar dos subtipos en la técnica colaborativa.

- **User-based.** Se identifican usuarios similares y de acuerdo a estas similitudes se recomienda gustos de un usuario a otro.
- **Item-based.** Se identifican similitudes entre los objetos y así poder dar una predicción

Si bien hay diferentes formas de implementar un Sistema de Recomendación, el punto esencial es que se mejora la satisfacción de las necesidades del cliente, hacer que la experiencia del usuario sea mas agradable, así sienten una atención personalizada y pues que el usuario consuma los productos que se le están recomendando de acuerdo a sus intereses.[2]

Por otro lado, los chatbots son un software, con algoritmos capaces de simular una conversación humana a través de una interfaz conversacional, mas enfocado es un software diseñado para automatizar una tarea específica, así pues, automatizar la respuesta de un usuario por medio del intercambio de mensajes en lenguaje natural.

Ya que estamos hablando de un procesamiento de lenguaje natural, humano, necesitamos que el software aprenda a poder llevar acabo ésta conversación. Una analogía es como enseñar a un bebé a hablar. Por lo cual el chatbot necesita un entrenamiento. Primeramente necesita entender lo que el usuario escribe y poder ser capaz de dar una respuesta coherente, por lo cual se implementan ciertos algoritmos de aprendizaje. Por consiguiente mediante éstos algoritmos el chatbot es capaz de de agregar nuevo entendimiento y poder dar futuras respuestas consistentes,

precisas.

Es importante destacar ciertas cualidades que trae consigo un chatbot:

- Está basado en IA
- Calidad y acierto en las respuestas
- Educar con efectividad
- Ejecutan acciones útiles
- Visibilidad e Integración
- Datos y personalización
- Específico para cada uso
- Medir y analizar
- Transparencia y principios éticos

Ya que lo que requerimos es una atención personalizada, el sistema de recomendación mediante el algoritmo de aprendizaje, podemos obtener ciertas, respuestas basadas en éstas recomendaciones, en especial sobre recetas de comida. Como anteriormente se mencionó el punto de los chatbots es automatizar acciones humanas, por ahora automatizar la búsqueda o recomendación de recetas de comida.

Así finalmente, el usuario tendrá su atención personalizada y respuestas consistentes así como precisas, que a fin, es la esencia del chatbot.

## Objetivo

El objetivo se divide en 2 partes, la primer parte enfocada en lo que viene siendo el sistema de recomendación enfocada en videojuegos y la segunda en un chatbot que recomienda recetas de cocina de acuerdo a lo que se proporcione, al final forman parte de un mismo sistema el cual estará corriendo sobre Discord para ser más amigable con el usuario.

## Sistema de Recomendación

Un usuario al querer jugar nuevos videojuegos, empieza un determinado sistema de búsqueda, lee la sinopsis, ve la portada o simplemente se lo recomendaron. Aquí es donde radica el problema ya que con todo la discriminación para elegir el juego aún puede que no le llegara a gustar. Ahora con un sistema de recomendación simplemente se le ofrecerán un grupo de juegos, de acuerdo a sus gustos, éstos debidamente procesados mediante el sistema de recomendación. Pudiendo evitar dejar juegos a medias o bien pasar por alto alguno que podría gustarle.

Así bien el sistema identificará similitudes en los ratings dados a los juegos seleccionados por los usuarios, así se tendrá una base de datos, donde se tendrá un historial de items vistos,

como el impacto obtenido con el usuario.

Posteriormente en los casos donde el usuario1 y el usuario2 dieron un rating a un item1 e item2 respectivamente, entonces para el usuario3 en el caso que le de un rating similar al usuario1 o usuario2, se le recomendará al usuario3 los items del usuario con el que tuvo la similitud. Por lo tanto Una vez que se ha construido una lista de usuarios, se combinan sus preferencias para generar una lista con los N juegos mas recomendables para el usuario actual.[1][5]

El sistema de recomendación se basará en un conjunto de usuarios, donde tendrá información de las juegos jugados, así como la reacción a estos (ratings). Con éstos datos, se realizará el proceso para poder determinar similitudes y así poder dar una recomendación eficiente acorde a los intereses buscados. Estos datos son indispensables ya que según a la película o item se hará un Join (unión) con el conjunto de usuarios para obtener un match y realizar el procesamiento de datos y con esto, observar las similitudes que hay con otros usuarios.

## **ChatBot**

Si buscamos recetas en la web, claramente encontraremos varias recetas, sin embargo, éstas distribuidas en muchísimas páginas. Ahora bien si interactuamos con un agente inteligente, indicando que nos de ciertas recetas, obtendremos lo requerido sin tener que estar indagando por varias recetas, y olvidarnos de la decidía de escoger alguna.

Por otro lado nos encontramos con pros y contras relacionado a una App, puesto que las recetas que proporcionamos ya están dadas de la manera más sencilla y compacta posible, mientras que en una App pueden tener imágenes y/o muchas pantallas, cosa que podría ser molesto para algunos usuarios. Llegando al grado de dejarse guiar y querer un resultado como lo muestran, y al percatarse de un fallo llegar al punto de molestia, sin embargo el chatbot, tendrá una parte subjetiva, de que por medio de una serie de pasos el usuario pueda llegar a un resultado y esté a gusto con éste mismo.

Ahora bien, el chatbot proporcionará las recetas de 3 maneras.

- El usuario deberá escribir solicitando una receta (donde la palabra clave de aquí es - eceta”), y así se le entregará una receta aleatoriamente de las 3 categorías, ya sea, fácil, media o compleja.
- Si el suario pide una receta y después alguna de las siguiente palabras (fácil, media o compleja), se le entregará una receta según lo solicitado, cabe aclarar que será de manera aleatoria.

## **Desarrollo**

### **Sistema de Recomendación**

El sistema de recomendación se desarrollará mediante la técnica de filtrado colaborativo, así como el uso del algoritmo por items, ya que los algoritmos como "Recomendación simple" y "Recomendación por contenido", son basados en la popularidad, así como en más datos que pueden restringir más nuestra recomendación. Sin embargo el hecho de realizar una recomendación, es no sólo guiarnos en algo subjetivo como la popularidad, si no que se pueda recomendar

con similitudes para con cualquier juego, no importando el director o año.[2] Ahora bien, para el sistema de recomendación se necesita la correlación con los vecinos, dado que hay sistemas con millones de usuario, compararlo con todos sería muy difícil, sin embargo, se establece una meta y así se toma un subconjunto de usuarios para la recomendación.[5]

## Análisis Exploratorio de Datos (EDA)

Para el sistema de recomendación se utilizó una lista finita de juegos, así mismo los ratings de estos mismos datos por múltiples usuarios.

## Preparación de Datos

Nuestro sistema recibe dos hojas de datos, una que contiene los juegos y otra que contiene los Ratings de estos, de aquí se hace un filtrado de datos quitando los datos innecesarios.

Así pues se toman las similitudes con otros usuarios, y así poder darles una recomendación de una película que a otro usuario le haya gustado. Claramente, éstos usuarios estarán en un grupo donde se haya hecho un match de un cierto juego que les haya gustado. Cabe aclarar que surge un problema, los gustos son muy subjetivos y por ende una persona puede ser muy optimista e indicar un juego con un rating de 5, mientras que otra puede dejarlo sólo en 3, sin embargo estamos tomando en cuenta, que todo esto se encuentra en un sólo nivel.

## Modelado de Datos y Propuesta de Solución

Al usuario actual se le muestra una lista de 5 juegos como recomendación de acuerdo al algoritmo ya procesado.

El sistema primero pide la calificación de algunos juegos, suponiendo que ya los jugaron[1]

```
1      print("Califica del 1 al 5 los siguientes juegos")
2      calif1 = input("Bioshock")
3
4      calif2 = input("Just Cause")
5
6      calif3 = input("Residen Evil 5")
7
8      calif4= input("Dead Space 2")
9
10     calif5 = input("Mario Party 4")
```

Seguido del sistema de recomendación y procesando toda la información podemos obtener una lista con los juegos que se recomiendan.

Dado que se busca medir la similitud entre los items mediante el ranking que tienen éstos, si bien una característica importante del sistema de recomendación es que puede medir éstas características, en particular se emplea la correlación de Pearson. Para poder tener las similitudes entre dos juegos, tenemos que:

$$d_{cor}(p, q) = 1 - \text{correlacion}(p, q)$$

Donde la correlación, es la correlación de Pearson para buscar la distancia o similitud entre los juegos.

Ahora bien, tenemos el sistema de predicción

$$\text{Predicción}(u_x, i_5) = \frac{\text{corr}(i_5, i_1) \times R_{u_x, i_1} + \text{corr}(i_5, i_4) \times R_{u_x, i_4} + \text{corr}(i_5, i_3) \times R_{u_x, i_3}}{\text{corr}(u_x, i_1) + \text{corr}(u_x, i_4) + \text{corr}(u_x, i_3)}$$

Donde  $u_x$  será la calificación que un usuario está dando a un juego, y realizando la medición, comparación con otros juegos para poder determinar una recomendación satisfactoria. Así bien, una vez tratadas las calificaciones otorgadas a los primeros 5 juegos, podemos obtener resultados como los siguientes:

	gameId	title	year
430	434	Resident Evil 4	2005
729	742	Metroid Prime 2	2004
2227	2311	Call of Duty: Black Ops 2	2012
2309	2393	Alice: Madness Returns	2011
3930	4023	Monster Hunter 4 Ultimate	2014
5429	5525	Postal 2	2003
7007	7118	Apex Legends	2019
9505	27808	The Legend of Zelda: Twilight Princess	2006

Al ser un agente inteligente sus **REAS** son las siguientes:

- **R:** Porcentaje de actividad en la recomendación de juegos.
- **E:** Conexión con la base de datos donde se almacenan los datos de los juegos, así como las calificaciones que le dan a los mismos(Rating).
- **A:** Sistema de aprendizaje y clasificación de información (Algoritmo de Recomendación).
- **S:** Sistema de filtración de información y sistema de recomendación.

También podemos ver que es un esquema basado en redes, ya que podemos ver ciertas conexiones con los ratings similares entre los usuarios, y así poder llevar a cabo una recomendación. Es decir, el algoritmo mediante items relaciona los juegos mediante los ratings, así poder encontrar similitudes entre uno y otro.

Siendo un esquema de representación una forma de representar la realidad en un ordenador, contamos con la parte estática y la parte dinámica

- **Parte estática**  
Es el conocimiento de la base de datos y con la cual empezamos a realizar el matching o a correlacionar los ratings que los usuarios han dado.
- **Parte dinámica**  
Es la implementación del algoritmo y con el cual empezamos a obtener datos sobre los juegos a recomendar.  
Procedimiento que permite interpretar los datos del problema a partir del conocimiento dado y así poder adquirir un nuevo conocimiento

El esquema de representación de conocimiento es procedimental ya que dado la inclusión de información (base de datos), se sabe como se usará mediante la filtración o proceso para llevar un match de gustos con otros usuarios. Es decir, de acuerdo a las calificaciones que se le otorgan, empieza a realizar una filtración buscando similitudes de esos juegos y así iniciar el filtrado en la base de datos para poder tener una recomendación eficaz mediante el algoritmo de items.

## **ChatBot**

Con la finalidad de dar una solución al problema planteado, se desarrolla un agente inteligente, es decir un chatbot, por lo cual es requerido ciertos componentes.

## **Análisis Exploratorio de Datos (EDA)**

Para nutrir se utilizo un archivo Json, previamente llenado con la info necesaria, esta incluye en su mayor medida recetas de cocina separadas por tags, asi mismo incluye cosas basicas del chatbot, como saludos, despedidas y ayudas.

## **Preparación de Datos**

Primeramente requerimos de un espacio de conocimiento, es decir, una base de datos, que contenga recetas de comidas las cuales fueron agregadas manualmente y se están abarcando recetas complicadas y sencillas. De esto se puede concluir que la mayor parte de los datos ya vienen pre procesados y en su mayoría “limpios”.

## **Modelado de Datos y Propuesta de Solución**

Para procesar el anterior conocimiento, requerimos del chatbot, un agente inteligente que está basado en el servicio al usuario, por medio de una comunicación mediante texto, es un chatbot ITR (Respuestas de Interacción de Texto) los cuales son muy específicos ya que respecto a lo que requiere el usuario; el sistema de procesamiento del texto el agente va a poder dar otorgar una respuesta en lenguaje natural, sin embargo, las limitaciones es que no pueden generar nuevas respuestas, esto conlleva a no tener un sistema de aprendizaje como tal, ni ofrecer nuevos servicios que no hayan sido programados previamente por un humano.

Este tipo de agente nos ayudará en poder entablar una conversación en lenguaje natural, así pues, mediante la conversación poder indicar ciertas recetas con respecto a lo requerido por el usuario ya sea que quiera una receta sencilla o complicada, en otro caso arrojará una respuesta que esté dentro del tema de conversación o de lo contrario indicar que no puede resolver tal tarea.

Al ser un agente inteligente sus **REAS** son las siguientes:

- **Rendimiento:** La coherencia con la que respondería a las preguntas en un lenguaje natural

Ya que es un agente con respuestas establecidas según a los requerimientos del usuario tendrá su respuesta concisa, ahora bien, en el caso en el que el requerimiento esté fuera del alcance del agente, éste responderá con un menú de opciones o una respuesta en la que indique que no puede actuar ante tal requerimiento.

- **Entorno:** El chatbot se encontrará en la plataforma DISCORD y mediante una conversación en lenguaje natural.
- **Actuadores:** Sistema de aprendizaje y sistema de recomendación. Mediante la conversación y según a las peticiones del usuario, el sistema de recomendación<sup>1</sup> tratará la petición, procesará el lenguaje y así poder dar una respuesta coherente a la petición
- **Sensores:** Modo de chat y poder conversar con el usuario. Corriendo el chatbot en la plataforma DISCORD mediante la conversación, el usuario podrá entablar una conversación, así podemos decir que el sensor es en parte la plataforma ya que por ése medio podrá obtener el requerimiento del usuario mediante textos.

Y sus propiedades del entorno son:

- **Parcialmente Observable.** Los sensores son poco exactos o no reciben la información del sistema, ya que el usuario puede dar sólo ciertas instrucciones o información parcial.
- **Estocástico.** No puede determinar su siguiente estado dado, que no se sabe que es lo que puede pedir el usuario.
- **Episódico.** Cada requerimiento del usuario lo podemos ver como un episodio y el agente realizará una acción, así pues, esta acción no afecta alguna otra acción del agente.
- **Semi-dinámico.** El entorno del agente no cambia con el tiempo, sin embargo, el rendimiento de éste, puede cambiar en el dado caso que se le cuestione con temas fuera de alcance.
- **Discreto.** El entorno tiene un número finito de estados, ya que se tiene preestablecidas ciertas preguntas y ciertas respuestas. En el caso en el que se quiera agregar algo nuevo al agente, la nueva tarea debe ser programada previamente.
- **Individual.** Un sólo agente esta respondiendo al requerimiento del usuario. No hay n agentes dando peticiones a otros.

## Resultados

### Sistema Recomendador

Sabemos que obtendremos una lista distinta con respecto a las calificaciones que damos, así podemos ver que el filtrado es distinto para cada usuario, cabe aclarar que puede que en un listado de recomendación de un usuario una película se puede repetir en algún otro listado. Así al hacer uso del sistema de recomendación sólo dependerá del gusto de cada usuario, además

---

<sup>1</sup>En este caso es algo mucho más simple que el anterior, en este caso hará match con las palabras propuestas.



de está ventaja, podemos tener que el sistema recomendará juegos que el usuario aún no ha clasificado.<sup>2</sup>

Sin embargo, el sistema de recomendación se limita, esto debido a que necesitamos que realice una clasificación de juegos para poder dar una recomendación, y si no tenemos esta información, entonces no se podría dar una buena recomendación.

Ahora bien, probando el sistema recomendador, tenemos la siguiente función, la cual de una lista de juegos, selecciona 5 y nos pregunta que calificación le daríamos:

```
gamesSurvey
✓ 0.0s

['Dead Island',
 'Call of Duty IV: Modern Warfare',
 'Postal',
 'Battlefield III',
 'Hearts of Iron']
```

Una vez que se nos pregunte, el sistema hara todo el proceso antes mencionado, sin que el usuario tenga que interferir más allá de las 5 valoraciones, al final el sistema retornara el siguiente resultado:

### Se busca recomendaciones para el usuario segun sus calificaciones

Correcto

```
es.loc[movies['movieId'].isin(recommendation_df.head(10)['movieId'].tolist())].drop('movieId', 1)
```

✓ 0.0s

```
/var/folders/pl/m9g73fks67x0r_5pptyg_4f00000gn/T/ipykernel_94038/1483887295.py:1: FutureWarning: In a future vers
movies.loc[movies['movieId'].isin(recommendation_df.head(10)['movieId'].tolist())].drop('movieId', 1)
```

	title	year
0	50 Cent: Bulletproof	2010
8	Bully	2011
9	Call of Duty IV: Modern Warfare	2010
14	Command & Conquer Generals	1999
20	Dark Sector	2006
22	Dead Island	2010
23	Dead Island: Riptide	2012
32	EA Sports MMA	1999
35	Football Manager	2005
51	L.A. Noire	2006

Como se puede apreciar, nos da la lista de recomendaciones de juegos que nos pueden llegar a agradar, esta variara con las respuestas proporcionadas por el usuario.

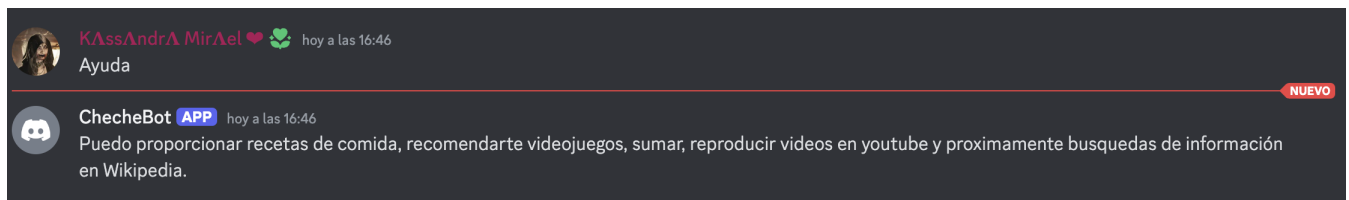
<sup>2</sup>Cabe aclarar que es un usuario promedio, es decir, juega lo recomendable al día y no posee tantos juegos (menor a 100-150 aproximadamente), en caso de que fuera un usuario que supera dichos números, el set de datos proporcionado, llegara a ser ineficiente.

## Chatbot

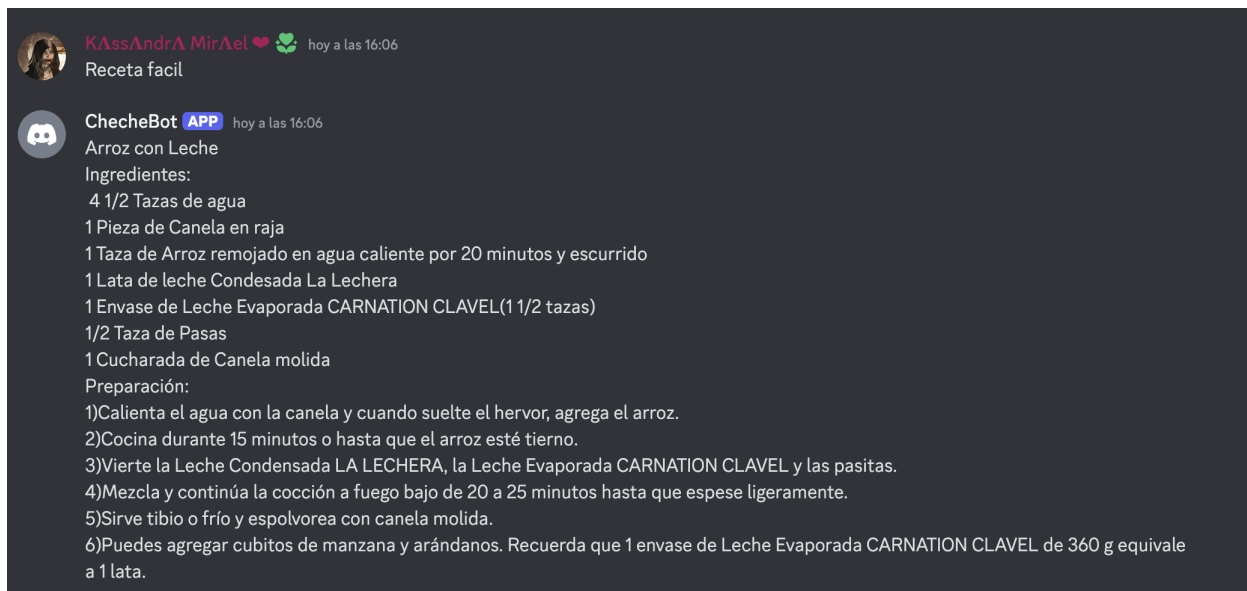
Los resultados del chatbot, en su mayoría fueron bastante acertados (hasta donde se limitó que respondería), por otra parte, si se le preguntara cualquier otra cosa que no va de acuerdo con lo que se le enseñó, este contestaría lo que crea más adecuado.

Además se realizaron pruebas con distintos usuarios explicando el uso del chatbot de forma básica, los resultados fueron mixtos, puesto que unos usuarios hicieron búsquedas muy complejas, es decir, pidiendo recetas con una lista de ingredientes, cosa que a veces al pedir un ingrediente bien específico podría dar cosas diferentes, por lo cual las respuestas en su totalidad eran incorrectas o parcialmente correctas, por otra parte el tipo de búsqueda que hicieron fue pedir simplemente recetas del estilo “Me podrías dar una receta”, “Me podrías brindar una receta.”<sup>o</sup> simplemente “Dame una receta”, aquí ya se respondía de manera más exacta a lo que pedían.<sup>3</sup>

Se dejan capturas de cómo se muestra el chatbot en discord:



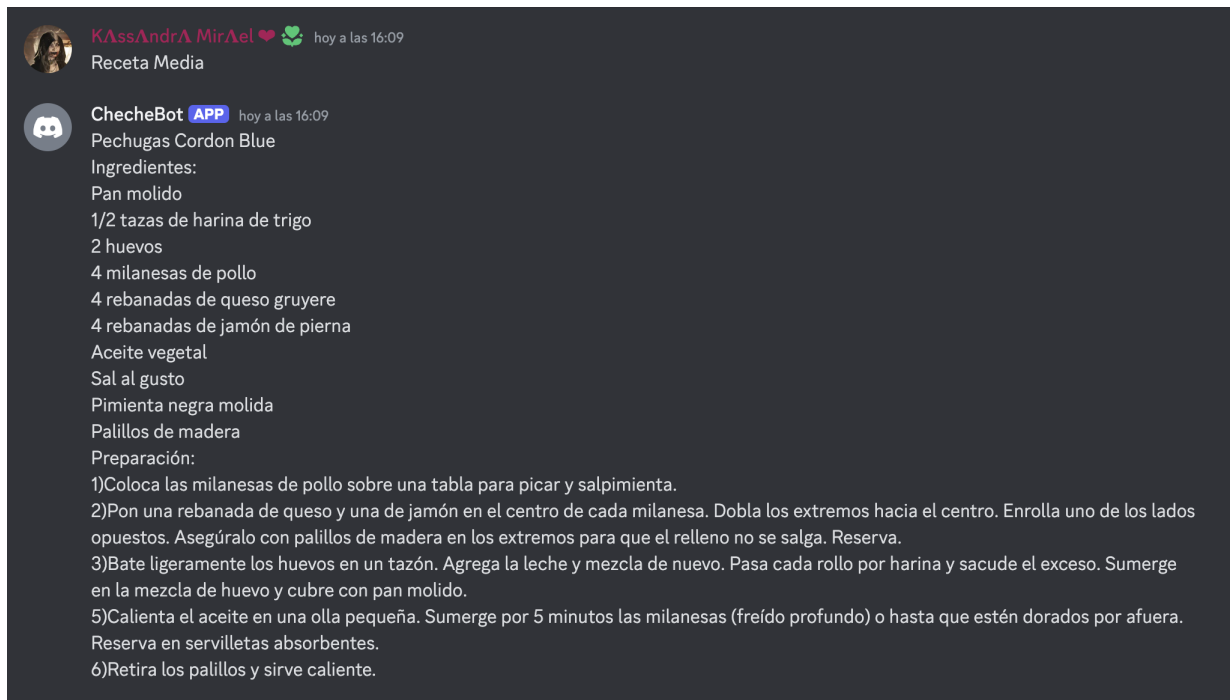
Pidiendo ayuda al Chatbot, para ver que tareas sabe realizar.



Pidiendo una receta de dificultad fácil.

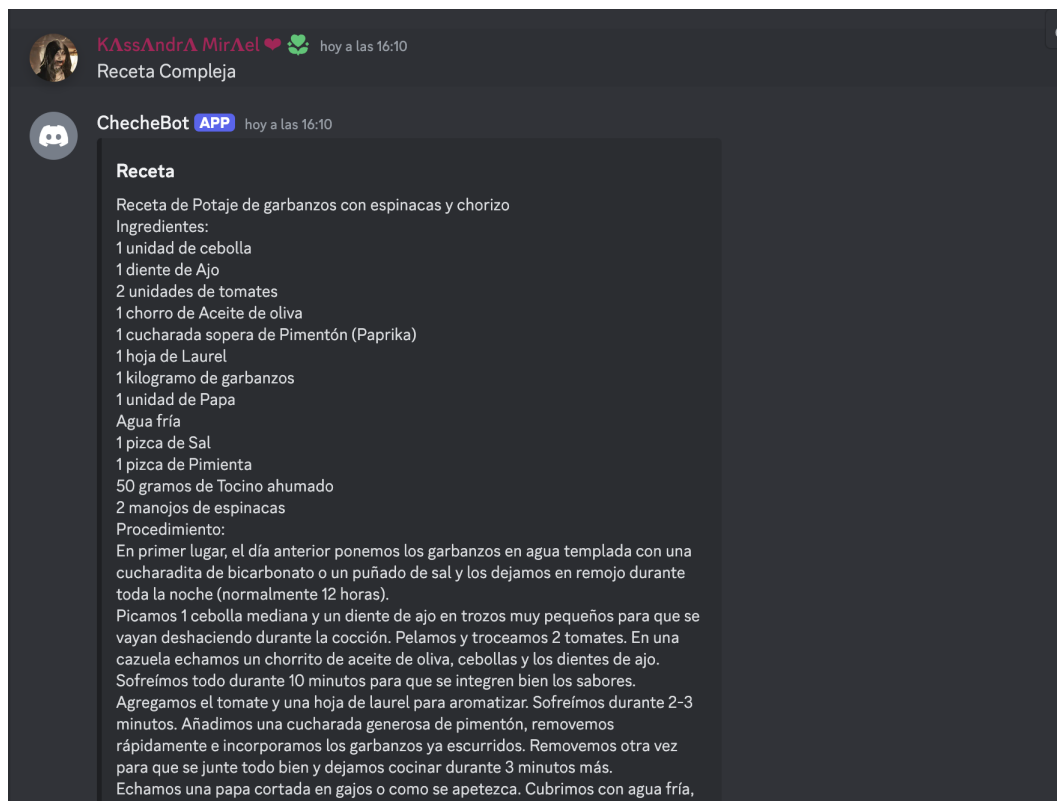
---

<sup>3</sup>Al ser más general, pues el resultado casi siempre es el esperado.



Pidiendo una receta de dificultad Media.

Como se puede apreciar, en los casos anteriores, son recetas que son realizables en pocos pasos, y que poseen una cantidad menor a 2000 caracteres, en el caso de que requiramos una receta un poco más compleja, ya sea como con más pasos, tendremos el siguiente caso:



Se utiliza un “embed” para almacenar el mensaje, estos solo se harán cuando los caracteres sean mayores o iguales a 2000 pero menores a 6000.

# Evaluación

Aunque como tal, no se puede evaluar tan directamente el sistema de recomendación (Se debe de graficar cada salida lo cual lleva tiempo), el chatbot es mas fácil evaluarlo, pues como se generan probabilidades dada la entrada, entonces es fácil ver a que tag es el que trata de hacer énfasis, por ejemplo, si usamos la palabra “Hola” y tenemos los tags [“saludo”, “despedida”, “receta”], entonces es fácil ver que Hola corresponde a un saludo, por lo que su probabilidad sera alta que corresponda esa tag.

```

Hora de Testear
Tu: Hola
[ 'AI': 2.779862e-12', 'Asassins': 4.779578e-18', 'Ponche crema de Oreo': 5.136879e-19', 'Ponche crema de café casero': 4.4264788e-18', 'Receta de Adobo para carne de cerdo a la parrilla': 0.0', 'Receta de Alioli con túrmix': 0.0', 'Receta de Alitas de pollo en salsa de vino blanco': 1.4240344e-35', 'Receta de Aporreadillo rojo': 0.0', 'Receta de Arroz con bacalao y verduras': 3.2815915e-36', 'Receta de Arroz con quinoa y verduras': 0.0', 'Receta de Bechamel de calabacín light': 7.8680016e-32', 'Receta de Bizcocho de yogur griego': 0.0', 'Receta de Bizcocho húmedo de calabaza': 5.5810067e-35', 'Receta de Bolitas energéticas de avena y frutos secos': 1.4488463e-26', 'Receta de Burritos veganos (vegetarianos) con frijoles': 0.0', 'Receta de Calamares encebollados': 9.5999015e-32', 'Receta de Calamares marinados a la plancha': 0.0', 'Receta de Champiñones rellenos de queso': 9.215837e-37', 'Receta de Chuletas de cerdo en salsa fácil': 0.0', 'Receta de Cous cous vegetariano': 0.0', 'Receta de Croquetas de patata y queso': 0.0', 'Receta de Croquetas de verduras veganas': 0.0', 'Receta de Empanada de bacalao con masa quebrada': 4.610206e-11', 'Receta de Empanada de pollo con masa quebrada': 1.5802497e-22', 'Receta de Empanadas de jamón, queso y choclo': 1.307549e-22', 'Receta de Empanadillas de pisto': 2.553535e-22', 'Receta de Ensalada de quinoa vegana': 0.0', 'Receta de Fajitas de verduras y tofu': 0.0', 'Receta de Fajitas veganas': 0.0', 'Receta de Galletas de alubias': 0.0', 'Receta de Galletas de crema de cacahuate veganas': 0.0', 'Receta de Gambones al horno con limón': 0.0', 'Receta de Queso provolone al horno con tomate': 1.8995564e-29', 'Receta de Quiche con base de boniato': 0.0', 'Receta de Quiche de bacon y queso de cabra': 2.0596317e-21', 'Receta de Queso provolone al horno con tomate': 1.8995564e-29', 'Receta de Quiche con base de boniato': 0.0', 'Receta de Quiche de bacon y queso de cabra': 2.0596317e-21', 'Receta de Quinoa con verduras y pollo': 0.0', 'Receta de Risotto de gambas y parmesano': 2.82142e-35', 'Receta de Risotto de setas y queso parmesano': 1.25675e-26', 'Receta de Rollitos de tortillas de trigo': 0.0', 'Receta de Rollos de canela sin huevo': 0.0', 'Receta de Salsa Alfredo sin nata ni leche': 0.0', 'Receta de Salsa de camarones para pasta': 0.0', 'Receta de Salsa de zanahoria y cebolla para carne o verdura': 0.0', 'Receta de Setas con jamón': 2.5755902e-08', 'Receta de Setas rebozadas': 0.0', 'Receta de Shiro etíope': 5.9055025e-11', 'Receta de Tacos veganos de coliflor': 4.5447705e-37', 'Receta de Tempeh rebozado': 1.5773402e-26', 'Receta de Torrijas veganas al horno': 0.0', 'Receta de Torta de coco húmeda': 0.0', 'Receta de Veganas de aguacate': 0.0', 'Receta de Veganas de zanahoria': 0.0', 'Respuestas_x': 7.181385e-08', 'broma': 1.0056061e-11', 'contestacion_saludo': 3.715115e-20', 'desayuno': 2.8901963e-16', 'desayuno_respuesta': 3.0880256e-09', 'despedida': 2.5829302e-31', 'pregunta_receta': 4.5270943e-26', 'preguntas_saludo': 7.223402e-21', 'recetas_de_Cocina_Complejas': 3.802296e-08', 'recetas_de_Cocina_Medias': 1.295999e-15', 'recetas_de_Cocina_Simples': 7.874146e-12', 'saludo': 0.99999976', 'uso': 3.835143e-32']
ChecheBot: ¿Como te va?
Tu: 
```

Así mismo, si utilizamos otras expresiones, nos calculara la probabilidad de que haga referencia dicha expresión, en este caso solicitamos una receta

```

Tu: Receta facil
[ 'AI': 1.5230417e-25', 'Asassins': 0.0', 'Ponche crema de Oreo': 6.9573406e-34', 'Ponche crema de café casero': 2.0697228e-33', 'Receta de Adobo para carne de cerdo a la parrilla': 0.0', 'Receta de Alioli con túrmix': 0.0', 'Receta de Alitas de pollo en salsa de vino blanco': 0.0', 'Receta de Aporreadillo rojo': 0.0', 'Receta de Arroz con bacalao y verduras': 0.0', 'Receta de Arroz con quinoa y verduras': 0.0', 'Receta de Bechamel de calabacín light': 0.0', 'Receta de Bizcocho de yogur griego': 0.0', 'Receta de Bizcocho húmedo de calabaza': 0.0', 'Receta de Bolitas energéticas de avena y frutos secos': 0.0', 'Receta de Burritos veganos (vegetarianos) con frijoles': 0.0', 'Receta de Calamares encebollados': 0.0', 'Receta de Calamares marinados a la plancha': 0.0', 'Receta de Champiñones rellenos de queso': 3.967523e-31', 'Receta de Chuletas de cerdo en salsa fácil': 0.0', 'Receta de Cous cous vegetariano': 0.0', 'Receta de Croquetas de patata y queso': 0.0', 'Receta de Croquetas de verduras veganas': 0.0', 'Receta de Empanada de bacalao con masa quebrada': 6.8537003e-22', 'Receta de Empanada de pollo con masa quebrada': 1.2317435e-14', 'Receta de Empanadas de jamón, queso y choclo': 2.1966999e-12', 'Receta de Empanadillas de pisto': 1.2620115e-14', 'Receta de Ensalada de quinoa vegana': 0.0', 'Receta de Fajitas de verduras y tofu': 0.0', 'Receta de Fajitas veganas': 0.0', 'Receta de Galletas de alubias': 0.0', 'Receta de Galletas de crema de cacahuate veganas': 0.0', 'Receta de Gambones al horno con limón': 4.6662406e-32', 'Receta de Guiso de frijoles con carne': 0.0', 'Receta de Guiso de frijoles negros con carne': 0.0', 'Receta de Guiso de ternera con verduras': 0.0', 'Receta de Gyoza de verduras': 2.2829055e-32', 'Receta de Hamburguesas vegetarianas de quinoa, miho y vegetales': 0.0', 'Receta de Higaditos de pollo en salsa con vino blanco': 0.0', 'Receta de Hojaldre de espárragos trigueros': 9.027202e-18', 'Receta de Hojaldre relleno de queso de cabra': 9.633175e-36', 'Receta de Koba akondro': 0.0', 'Receta de Paella de verduras vegana': 0.0', 'Receta de Paella vegana de quinoa': 0.0', 'Receta de Pastel de champiñones al microondas': 0.0', 'Receta de Pastel de remolacha salado': 7.405292e-33', 'Receta de Patatas a la riojana fácil': 0.0', 'Receta de Paté de atún con queso crema': 5.6924047e-19', 'Receta de Paté de mejillones': 2.7874286e-21', 'Receta de Pesto de espinacas vegano': 0.0', 'Receta de Pollo enchilado al horno para Navidad': 0.0', 'Receta de Porridge inglés': 0.0', 'Receta de Potaje de garbanzos con espinacas y chorizo': 0.0', 'Receta de Queso provolone al horno con tomate': 5.7845407e-34', 'Receta de Quiche con base de boniato': 7.2670556e-31', 'Receta de Quiche de bacon y queso de cabra': 8.2875347e-38', 'Receta de Quinoa con verduras y pollo': 0.0', 'Receta de Risotto de gambas y parmesano': 0.0', 'Receta de Risotto de setas y queso parmesano': 0.0', 'Receta de Rollitos de tortillas de trigo': 0.0', 'Receta de Rollos de canela sin huevo': 0.0', 'Receta de Salsa Alfredo sin nata ni leche': 0.0', 'Receta de Salsa de camarones para pasta': 0.0', 'Receta de Salsa de zanahoria y cebolla para carne o verdura': 0.0', 'Receta de Setas con jamón': 0.0', 'Receta de Setas rebozadas': 0.0', 'Receta de Shiro etíope': 2.884489e-31', 'Receta de Tacos veganos de coliflor': 8.024775e-38', 'Receta de Tempeh rebozado': 0.0', 'Receta de Torrijas veganas al horno': 0.0', 'Receta de Torta de coco húmeda': 0.0', 'Receta de Veganas de aguacate': 0.0', 'Receta de Veganas de zanahoria': 0.0', 'Respuestas_x': 8.0713505e-07', 'broma': 0.0', 'contestacion_saludo': 0.0', 'desayuno': 5.3650532e-33', 'desayuno_respuesta': 2.8000994e-21', 'despedida': 0.0', 'pregunta_receta': 3.8891867e-36', 'preguntas_saludo': 9.996952e-33', 'recetas_de_Cocina_Complejas': 3.901031e-12', 'recetas_de_Cocina_Medias': 4.0767645e-06', 'recetas_de_Cocina_Simples': 0.99999951', 'saludo': 9.533909e-23', 'uso': 0.0']
```

# Conclusiones

Podemos ver que el alcance de este proyecto nos para explorar distintas técnicas para procesar set de datos, siendo uno un conjunto de archivos csv y el otro un archivo json.

Aunque si bien los set de datos se encuentran limitados por la cantidad inmensa de juegos así como de recetas, se deja la base para poder conectar a otros set de datos que si bien, son mas grandes, estos pueden venir menos limpios, por lo que se deberá usar funciones adicionales para limpieza de estos, asi mismo, buscar la manera para poder procesar todos los datos en

tiempo razonable para que el usuario no debe esperar demasiado, pues en estos casos, la espera hace que pierda interés.

## Referencias

- [1] Recommender System Python
- [2] Build recommendation engine collaborative filtering
- [3] Collaborative Filtering Using n Nearest Neighbors
- [4] Sistemas
- [5] Galán S., Filtrado Colaborativo y Sistemas de Recomendación, Universidad Carlos III de Madrid, Inteligencia en Redes de Comunicación
- [6] eris, R. (2018). Chatbot: ¿Qué es, para qué sirve y cómo funcionan?. enero 19, 2021, de Bloo Media Sitio web: <https://bloo.media/blog/por-que-implementar-chatbot-en-tu-estrategia-de-marketing/>
- [7] edreño, A. (2019). La revolución de los chatbots y la inteligencia artificial. enero 19,2021, de LLYC Sitio web: <https://www.revista-uno.com/33-revista-uno-tecnologias-exponenciales/la-revolucion-de-los-chatbots-y-la-inteligencia-artificial/>
- [8] LYC. (2019). CHATBOTS Y AGENTES INTELIGENTES. EL DESAFÍO DE LA CONVERSACIÓN ARTIFICIAL. enero 19,2021, de LLYC Sitio web: <https://ideas.llorenteycuenca.com/2019/12/chatbots-y-agentes-inteligentes-el-desafio-de-la-conversacion-artificial/>
- [9] a Silva, D. (2020). ¿Qué tipos de chatbots existen, cómo podemos categorizarlos y cuál puede servirte de acuerdo a las necesidades de tu empresa?. enero 19,2021, de Zendesk Sitio web: <https://www.zendesk.com.mx/blog/tipos-de-chatbot/>
- [10] nboundCycle. (2020). Chatbot: la automatizacion de la atención al cliente. ENERO 19,2021, de InboundCycle Sitio web: <https://www.inboundcycle.com/diccionario-marketing-online/chatbot>
- [11] 0defiebre. (2020). ¿Qué es un chatbot?. enero 19,2021, de 40defiebre Sitio web: <https://www.40defiebre.com/que-es/chatbot>
- [12] utimbo, L. (2018). IMPLEMENTAR CHATBOT BASADO EN INTELIGENCIA ARTIFICIAL PARA LA GESTIÓN DE REQUERIMIENTOS E INCIDENTES EN UNA EMPRESA DE SEGUROS. Lima.Perú: UNIVERSIDAD SAN IGNACIO DE LOYOLA FACULTAD DE INGENIERIA .
- [13] tolomeo. (2003). Inteligencia Artificial. SGMA: CDMX: UNAM.