Casa abierta al tiempo

UNIVERSIDAD AUTÓNOMA METROPOLITANA

Maestría en Ciencias y Tecnologías de la Información

Diseño e Implementación de una Arquitectura de Chatbot

Idónea Comunicación de Resultados
Para obtener el grado de

MAESTRA EN CIENCIAS
EN TECNOLOGÍAS DE LA INFORMACIÓN

Presentada por

Ing. Areli Anzures Villarreal

2202800380

areli.anzu@gmail.com

Asesores:

Dr. Enrique Rodríguez de la Colina

Dr. Eric Alfredo Rincón García

Fecha: 29/09/22

Resumen

En la actualidad los chatbots permiten expandir los canales de comunicación entre los usuarios y las instituciones, lo que los hace una herramienta muy importante. La creación de los chatbots ha aumentado de manera significativa en los últimos años, pero la creación de estos puede presentar varios inconvenientes, por ejemplo, cada usuario puede hacer una misma pregunta de diferentes maneras, obligando al creador a identificar cada una de ellas correctamente y poder generar una respuesta adecuada. Con el avance de la tecnología se han desarrollado múltiples técnicas para poder entender, procesar y generar el lenguaje natural, tales como el Procesamiento de Lenguaje Natural (PLN), Comprensión del Lenguaje Natural (CLN) y Generación del Lenguaje Natural (GLN). Todas las técnicas antes mencionadas han ayudado a los desarrolladores a poder entender las peticiones de los usuarios sin importar de qué manera las digan y posteriormente poder generar la respuesta adecuada.

La investigación presentada se enfoca en identificar los componentes más utilizados en el desarrollo de chatbots encontrados en la literatura y proponer una arquitectura que sea capaz de satisfacer las necesidades del usuario proporcionando respuestas adecuadas a cada pregunta. La arquitectura está basada en múltiples componentes recopiladas de la literatura, para ayudar a los alumnos de la Universidad Autónoma Metropolitana Unidad Iztapalapa (UAM-I) a responder sus dudas de una manera más amigable sin necesidad de buscarlo manualmente desplazándose por las páginas de la unidad. El chatbot incorpora las técnicas de extracción de información para alimentar la base de datos interna, lo que permitirá poder responder cualquier pregunta relacionada con la información que se encuentre en las páginas de la unidad, teniendo como resultado un chatbot con información actualizada sin implicar un gran esfuerzo para la creación de las bases de datos.

Agradecimientos

A la Universidad Autónoma Metropolitana Unidad Iztapalapa, por darme la oportunidad de realizar mis estudios de posgrado.

Al Consejo Nacional de Ciencia y Tecnología (CONACyT), por otorgarme el financiamiento para realizar mis estudios.

A mis asesores el Dr. Enrique Rodríguez de la Colina y el Dr. Eric Alfredo Rincón García por su apoyo y paciencia que me brindaron durante la realización del proyecto.

A mis padres Mateo Luis y Luz María, quienes son mi inspiración, por su cariño y apoyo incondicional que me han brindado durante toda la vida.

A mi mejor amigo Luis Gil que me ha brindado su amistad y me ha apoyado en mis buenos y malos momentos. Por último, pero no menos importante a mi novio César Morales por apoyarme durante todo este tiempo, dándome todo su amor, cariño y comprensión.

índice

Lista de Figuras

Lista de Tablas

Capítulo 1

1. Introducción y Fundamentos Teóricos

La inteligencia artificial ha tenido un gran impacto desarrollando múltiples aplicaciones o dispositivos para facilitar las actividades diarias. Una de estas aplicaciones puede ser un agente inteligente, el cual sea capaz de mantener una conversación con los usuarios, este agente es mejor conocido como chatbot. Un chatbot es un programa informático que simula conversaciones humanas y proporciona respuestas en un corto periodo de tiempo [1].

Al inicio, los chatbots se crearon únicamente para el idioma inglés, pero al aumentar su popularidad, se han ido integrando nuevos idiomas. El crecimiento de la Inteligencia Artificial (IA) y el uso del Procesamiento de Lenguaje Natural (PLN) permiten que los chatbots sean capaces de entender el lenguaje y sean capaces de interactuar de una forma más amigable y humana. Además de ser capaces de imitar la interacción humana, la evolución del aprendizaje maquinal (Machine Learning) y el análisis de sentimientos, los chatbots pueden contar con la capacidad de responder de una manera más “emocional” a los usuarios, lo que puede hacer parecer que no se está hablando con un robot, si no, con un ser humano.

El desarrollo de la Inteligencia Artificial y la creciente demanda de servicios ha dado paso al desarrollo de chatbots en múltiples sectores, incluyendo el bancario, negocios, salud, educación, informativos, entre otros. En la Figura 1 se puede observar que a inicios de los años 90’s se comenzaron a realizar los primeros chatbots, pero fue a partir del año 2016 que se ha observado un incremento en su uso. De acuerdo con la Figura 2 los países que han mostrado más interés en la investigación han sido Estados Unidos de América, Reino Unido y Japón.

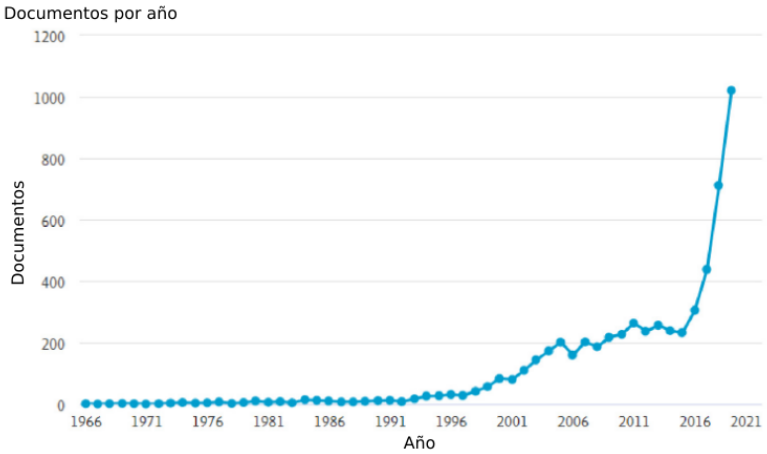

Figura 1 Documentos sobre chatbots publicados por año. Traducido de [2].

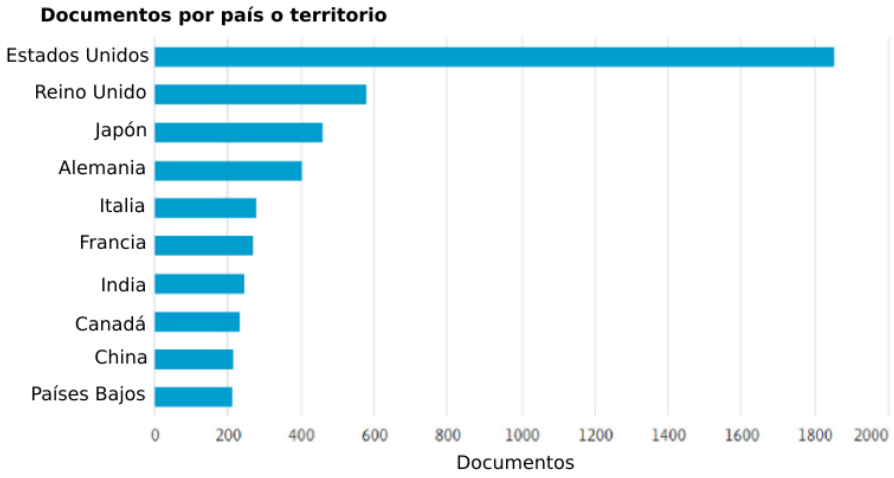

Figura 2 Número de documentos publicados por país o territorio. Traducido de [2].

Ahora bien, el diseño de chatbots no es algo nuevo, en 1966 se creó el primer chatbot llamado ELIZA, el cual simulaba a un psicoterapeuta, devolviendo las oraciones del usuario en forma interrogativa [3]. La capacidad comunicativa era limitada, pero sirvió como inspiración para múltiples chatbots. Posteriormente, en 1972, fue creado PARRY, el cual actuaba como un paciente con esquizofrenia. PARRY fue considerado más avanzando que ELIZA porque contaba con una "personalidad". Este chatbot fue comparado contra un paciente real con este padecimiento y fueron diagnosticados por psiquiatras donde 4 de 10 de ellos si confundieron el chatbot, con un paciente real [4]. Con el inicio de la Inteligencia Artificial, en 1988 se utilizó para la creación de chatbots con el desarrollo de Jabberwacky [5]. Jabberwacky utilizaba la coincidencia de patrones, dicho de otra manera, una respuesta que coincida con lo que el usuario solicita.

Uno de los chatbots inspirados en ELIZA fue ALICE [6], el cual estaba basado en la coincidencia de patrones. ALICE fue desarrollado con un nuevo lenguaje creado para este propósito, Artificial Intelligence Markup Language (AIML). El leguaje AIML trataba de hacer coincidir palabra por palabra entre el mensaje del usuario y las respuestas predefinidas para obtener la coincidencia más larga y encontrar cual era la mejor respuesta, lo que podía permitir al chatbot ser más flexible e interactivo para diferentes campos [7].

Años después, en el 2001, se creó SmarterChild el cual estaba disponible en America Online (AOL) y Microsoft Messenger (MSN) [8]. Este chatbot fue el primero en ayudar a las personas en tareas diarias, ya que era capaz de recuperar la información de múltiples bases de datos de películas, resultados deportivos, noticias y clima. Finalmente, con el desarrollo de la inteligencia artificial, los chatbots fueron un paso más adelante con la creación de los asistentes personales, los cuales cuentan con voces artificiales y son capaces de hacer múltiples tareas como el manejo de calendarios, correos y hasta manejar dispositivos automatizados capaces de ser conectados al Internet. Siri de Apple, Alexa de Amazon, Cortana de Microsoft y Watson de IBM son algunos ejemplos de estos dispositivos [2].

Como se mencionó anteriormente para la creación de los chatbots se han desarrollado múltiples técnicas, desde chatbots que regresaran la entrada del usuario en forma de respuesta, hasta ser capaces con ayuda de la IA de entender y generar textos en cualquier idioma, además de poder hacer tareas como dar reportes del clima, agendar citas, dar información sobre algún tema, etc.

1.1 Inteligencia Artificial

Se pueden encontrar múltiples definiciones distintas para Inteligencia Artificial (IA), algunas de ellas son:

"La IA es la capacidad de un sistema para poder interpretar datos externos, aprender de dichos datos y utilizar esos aprendizajes para lograr objetivos específicos y tareas a través de la adaptación." [9]

"IA es la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano." [10]

"En términos simples, la IA se refiere a sistemas o máquinas que imitan la inteligencia humana para realizar tareas y pueden mejorar iterativamente a partir de la información que recopilan." [11]

Dentro de la inteligencia artificial hay varios subcampos, uno resulta de interés para este proyecto, es el procesamiento de lenguaje natural, el cual se describe a continuación.

1.1.1 PNL

El PLN es un subcampo de la Inteligencia Artificial que explora cómo se puede utilizar las computadoras para que puedan comprender y manipular el texto y/o habla del lenguaje natural [12]. El PLN se puede dividir en dos partes fundamentales:

- CLN: En el núcleo de cualquier tarea del PLN está la comprensión del lenguaje natural, el cual se dedica a interpretar un mensaje y para ser capaz de entender el significado e intención. Para esto es necesario entender las reglas gramaticales, el orden de las palabras y su significado
- GLN: Una vez lograda la habilidad del entendimiento del lenguaje se debe de tener la capacidad de crear un nuevo mensaje. La generación del lenguaje natural se encarga de elegir la información necesaria para poder producir un mensaje, para esto hay que decidir cómo organizarla de acuerdo con los recursos gramaticales, morfología, etc.

Las aplicaciones del PLN pueden ser variadas, algunas incluyen:

- Traducción de idiomas
- Análisis de sentimientos
- Texto predictivo
- Asistentes inteligentes o Chatbots
- Análisis de texto

En la siguiente sección se presenta una descripción más detallada de lo que es un chatbot.

1.1.2 Chatbot

Se pueden encontrar múltiples definiciones de un chatbot, algunas de ellas son:

“Un chatbot es un software capaz de responder una serie de preguntas de los usuarios proporcionando respuestas correctas.”[13]

“Un chatbot es un programa de computadora que fue creado para imitar a los humanos en una conversación.”[14]

“Un chatbot es una herramienta de software que interactúa con los usuarios sobre un tema determinado utilizando texto o voz.”[15]

El uso de los chatbots y sus aplicaciones pueden ser variadas, estos son creados con el propósito de mejorar la experiencia de los usuarios por parte de las empresas de una manera eficiente sin la necesidad de aumentar su personal. A continuación, se describen algunas de las aplicaciones más relevantes.

Los chatbots como un apoyo educativo pueden repetir lecciones anteriores cuando los estudiantes no han asistido a clases o no han entendido por completo tema. Se puede mejorar el proceso de aprendizaje y enseñanza ya que puede interactuar con los alumnos en cualquier momento así que es capaz de ajustarse a las necesidades de cada uno de ellos. Otra aplicación es en el comercio. Con el desarrollo del comercio electrónico, múltiples empresas optaron por vender sus productos en línea teniendo como inconveniente el responder dudas sobre sus productos, por lo que el uso de chatbots han facilitado este proceso en comparación con atender a sus clientes por medio telefónico o chats en vivo, lo que puede tener un gran tiempo de espera si se tienen muchos usuarios a la vez. Los chatbots pueden operar las 24 hrs del día respondiendo cualquier pregunta, lo que da como resultado aumentar la satisfacción del usuario y las ventas de la empresa. Respecto a aplicaciones médicas los chatbots han sido diseñados para proporcionar a los pacientes información sobre salud, productos relacionados a su enfermedad o hasta ofrecer diagnósticos y sugerir tratamientos basados en los síntomas que el usuario le proporcione al bot. Finalmente, hay chatbots los cuales tienen el objetivo de entretener a la personas con una simple conversación, presentando los resultados de partidos de futbol o ser guía en algún lugar turístico. Aunque en esta sección se mencionan algunas aplicaciones, los chatbots pueden tener múltiples aplicaciones más.

Una manera de clasificar a los chatbots es por su propósito como se mencionó anteriormente, pero los chatbots se pueden clasificar por múltiples maneras (Tabla 1), ya sea por el dominio del conocimiento, el cual sea capaz de dominar uno o múltiples temas o ser un chatbot genérico. Se puede clasificar en el tipo de servicio que proporciona, el ser capaz de hablar con una o múltiples personas a la vez. El canal de comunicación, si el chatbot será capaz de comunicarse por medio de la voz o solo por texto o imagen.

Tabla 1. Clasificación de los chatbots. Traducida de [2].

Categorías	Dominio del Conocimiento	Genérico
		Dominio abierto
		Dominio cerrado
	Servicio	Interpersonal
		Intrapersonal
		Inter-Agente
	Propósito	Informativo
		Conversacional
		Basado en tareas
	Método de Generación de Respuesta	Basado en reglas
		Basado en recuperación
		Generativo
	Permisos	Código abierto
		Comercial
	Canal de Comunicación	Texto
		Voz
		Imagen

Una de las categorías más importantes para la creación de un chatbot es el método de generación de respuesta. Existen cuatro métodos de generación de respuesta:

Basado en reglas: Selecciona las respuestas de un conjunto de posibles respuestas predefinidas (reglas), utilizando la coincidencia de patrones. El funcionamiento básico es el siguiente: Se busca una coincidencia entre el mensaje del usuario y las posibles respuestas, aquella que sea la más aproximada será presentada al usuario Figura 3.

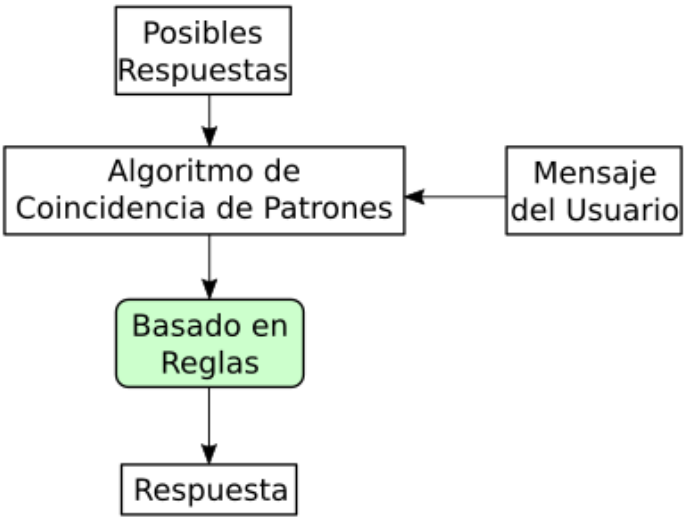

Figura 3. Modelo basado en reglas.

Basado en recuperación: Selecciona la respuesta más adecuada con el análisis de recursos disponibles. Su funcionamiento es muy similar al modelo anterior, a diferencia de que este modelo también toma en cuenta el contexto de la conversación, de esta manera dará respuestas más personalizadas de acuerdo con el usuario que esté hablando. Figura 4.

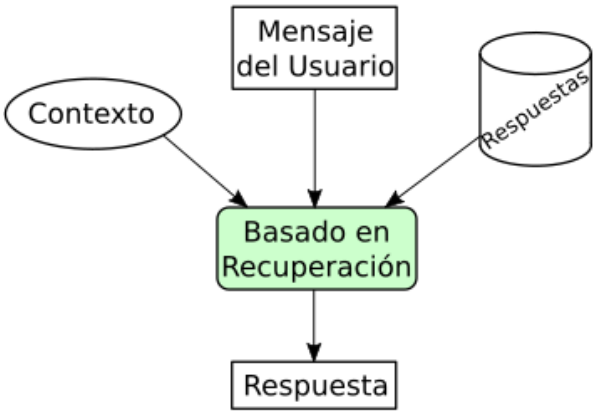

Figura 4. Modelo basado en recuperación.

Generativo: Utiliza el lenguaje natural para responder de una manera más humana. A diferencia de los modelos anteriores, el modelo generativo toma en cuenta el mensaje actual del usuario y los mensajes previos (del chatbot y usuario) e identifica las intenciones y entidades para poder generar una respuesta, esto lo hace con técnicas de Machine Learning. Figura 5.

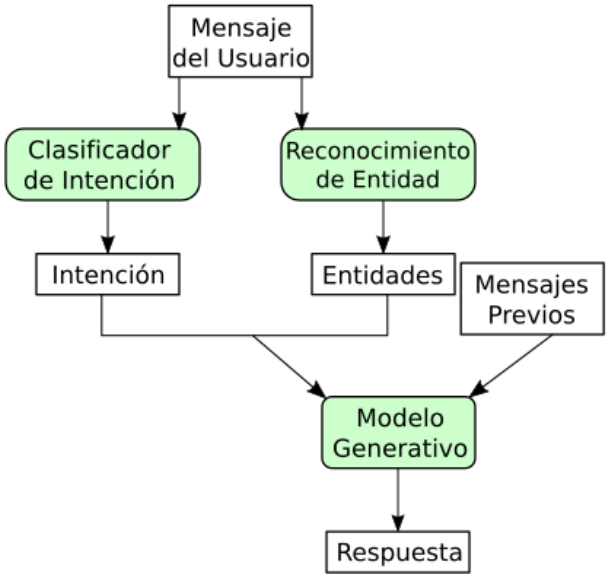

Figura 5. Modelo generativo.

Las entidades pueden ser palabras o datos claves que describen cualquier cosa. Por ejemplo, una hora, un lugar, etc [16]. La intención se refiere al objetivo que el usuario

tiene cuando realiza una pregunta o algún comentario [16], por ejemplo, hacer una petición, un pedido, solicitar información de un producto, etc.

Híbrido: Combina dos o más de los modelos disponibles antes descritos. Usando la combinación de todos los modelos se genera una serie de posibles respuestas y de acuerdo con el contexto y un selector de respuesta puede presentar la respuesta más indicada para el usuario. Figura 6.

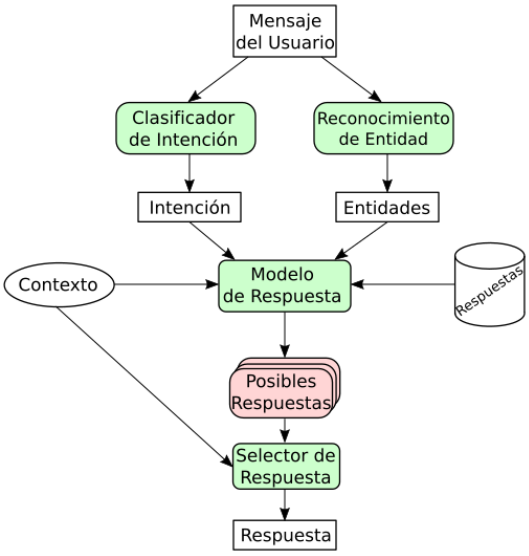

Figura 6. Modelo híbrido.

Otra de las categorías de los chatbots que es importante para su creación es el tipo de permisos que tienen las herramientas. En la actualidad existen diversas herramientas para el desarrollo de Chatbots sin necesidad de un conocimiento previo de programación, esto puede ser una gran ventaja, pero se tiene el inconveniente de no poder contar con un chatbot al gusto del desarrollador o estar limitado a las características o herramientas predefinidas, sin poder adecuarlas, modificarlas o mejorarlas de forma personal. Algunas de estas herramientas son:

- Facebook Messenger
- Bots de Telegram
- Rebot.me
- Botsify
- SnatchBot

Así como existen herramientas que no necesitan tener un conocimiento previo, también existen herramientas que permiten la creación de chatbots y la mejora de estos mismos con el fin de adaptarlo a las necesidades de los usuarios, tal como es el caso de RASA. A continuación, se hablará a detalle de esta herramienta, la cual fue utilizada para el presente proyecto ya que cuenta con una extensa documentación, posee las herramientas necesarias para crear chatbots, es fácil de utilizar, disponible en múltiples idiomas y permite la conexión con APIs.

1.1.3 RASA

RASA es un framework de aprendizaje automático de código abierto para conversaciones automatizadas de texto y voz [17]. El mensaje del usuario se maneja en dos módulos diferentes:

El módulo NLU proporciona clasificación de intenciones y extracción de entidades, mientras que el módulo Core es el módulo de gestión de diálogo basado en aprendizaje automático. Es el componente principal, cuya función es recibir y responder las solicitudes. En términos simples, RASA Core maneja el flujo de la conversación.

En la Figura 7 se muestra un ejemplo más a detalle del funcionamiento de RASA. Cuando el usuario hace una petición, por ejemplo "*Dame información sobre las becas*", el módulo NLU se encargará de extraer las intenciones y entidades, "*Solicitar información*" y "*Becas*" respectivamente. Una vez obtenida esa información, el módulo Core se encargará de generar una respuesta, esto lo hace mediante el uso de la intención, la entidad y la base de datos disponible. Este módulo puede generar múltiples posibles respuestas, pero solo se le mostrará al usuario la más indicada de acuerdo con la conversación.

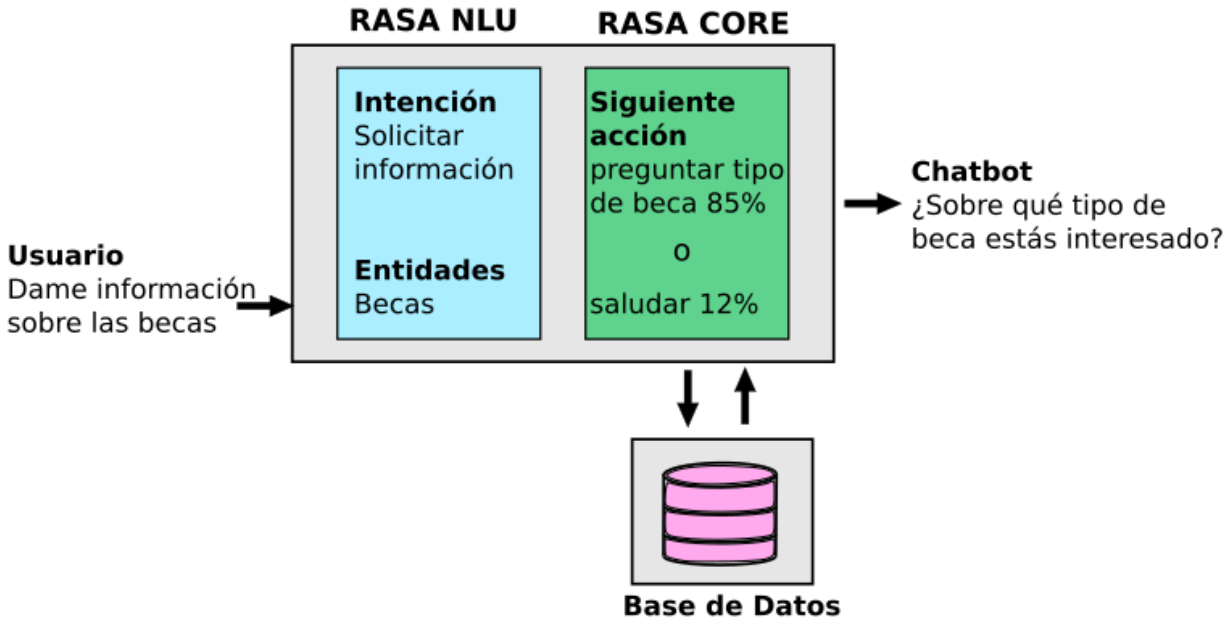

Figura 7. Funcionamiento de RASA.

Además de lo antes mencionado, RASA cuenta con algunos componentes que vale la pena mencionar:

Los datos de entrenamiento se dividen en distintos archivos, nlu, stories y rules en donde se encuentran los datos de ejemplos para entrenar el chatbot.

Los datos nlu son ejemplos de posibles expresiones empleadas por el usuario al interactuar con el bot, estos ejemplos están categorizados por intención y pueden contar con entidades. En la Figura 8 se muestran un ejemplo de estos datos, donde se indica la

intención dar_info_general y se muestran los ejemplos que el usuario podría ingresar “necesito información sobre los [seminarios](info)”. Se puede observar que seminarios está entre corchetes, esto se hace para indicar una entidad, en seguida se indica de que tipo es la entidad, en este caso es una entidad de tipo info, debe ser indicada entre paréntesis.

```
nlu:
- intent: dar_info_general
  examples: |
    - necesito informacion de los [seminarios](info)
    - información sobre [Plan de estudios](info)
    - información sobre [biblioteca](info)
    - quiero saber sobre la división [cbi](info)
```

Figura 8. Datos de entrenamiento nlu.

Las historias (stories) son ejemplos de conversaciones que podrían darse entre el usuario y el bot. Son utilizadas para entrenar el modelo y así poder identificar patrones de conversaciones. En la Figura 9 se pueden ver dos ejemplos de historias, éstas están conformadas por lo siguiente:

Story, indica el nombre de la historia “hola_adios”. Steps, indica los pasos a seguir en la conversación, por ejemplo: Si el chatbot recibe un mensaje “hola”, este debe de clasificar la intención como saludo, por lo que debe de tomar la acción de regresar ese saludo “utter_saludo”, el cual es un mensaje predefinido (se explica más adelante). De la misma manera debe de proceder cuando el usuario envía “adios”, detecta la intención de despedida y responde con un mensaje predefinido ”utter_despedida”.

```
- story: hola_adios
  steps:
  - intent: saludo
  - action: utter_saludo
  - intent: despedida
  - action: utter_despedida
```

Figura 9. Datos de entrenamiento historias.

Finalmente, las reglas (rules) describen pequeños fragmentos de conversación que siempre deben seguir el mismo camino, sin importar el contexto ni la secuencia de la conversación. En la Figura 10 se muestra un ejemplo de una regla, es muy similar a una historia, la diferencia es que las reglas se tienen que cumplir obligatoriamente, por ejemplo, sin importar en que etapa de la conversación se encuentre, si se recibe un mensaje de despedida, el chatbot siempre debe de despedirse. A diferencia de las historias, este flujo de conversación puede tener cierta flexibilidad y no fluir como se indica en los ejemplos.

```
rules:
- rule: decir adios cuando el usuario diga adios
  steps:
  - intent: despedida
  - action: utter_despedida
```

Figura 10. Datos de entrenamiento reglas.

El dominio (domain) es un archivo que contiene la lista de intenciones, entidades, slots, respuestas y acciones que el bot debe de conocer para su correcto funcionamiento. En Figura 11 se muestra la lista de intenciones y entidades utilizadas en el chatbot. Los slots son parecidos a las entidades, pero en este caso, un slot puede permanecer más tiempo en la memoria, mientras que las entidades no. Las entidades se utilizan únicamente en el momento en el que el usuario las menciona y el chatbot las emplea para generar la respuesta, mientras que los slots pueden ser utilizados en mensajes posteriores dentro de la misma conversación. Para la implementación no se hizo uso de los slots.

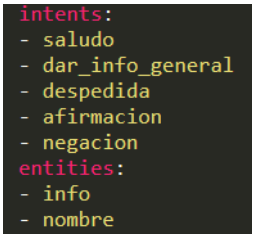

Figura 11. Ejemplo de entidades e intenciones.

Las respuestas (responses) son acciones que envían un mensaje a un usuario sin la necesidad de utilizar un código. Son respuestas predefinidas que se pueden incluir en el archivo como se muestra en la Figura 12. Finalmente se enlistan las acciones (actions) las cuales son lo que el chatbot puede hacer, por ejemplo, responder a un usuario, hacer una llamada API externa o consultar una base de datos (Figura 13).

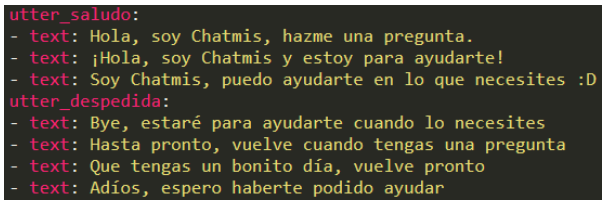

Figura 12. Mensajes predefinidos.

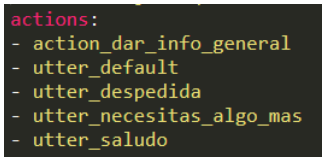

Figura 13. Lista de acciones.

El archivo de configuración es en donde se definen los componentes que el bot usará para hacer predicciones basadas en la entrada de los usuarios. Se cuentan con dos configuraciones distintas: el pipeline sirve para procesar la entrada del usuario. Hay componentes para extracción de entidades, clasificación de intenciones. Si no se indica en el archivo se utiliza DIETClasisifier por default. Finalmente, las policies se utilizan para decidir qué acción tomar en cada paso de una conversación. Hay basadas en reglas y en aprendizaje automático, todo esto se explica más detalladamente en el Capítulo 3.

Como se mencionó anteriormente, RASA cuenta con acciones que pueden ser utilizadas para leer una base de datos, en el caso del chatbot desarrollado se creó un acción la cual lee una base de datos para poder realizar una búsqueda y generar la respuesta a lo que el usuario está buscando, pero antes de realizar la búsqueda es necesario crear la base de datos y alimentarla con los datos extraídos de la web. A continuación, se explicará todo lo relacionado al tema.

1.2 Extracción de información de páginas web

Al inicio de la web no existían motores de búsqueda, por lo que si se deseaba encontrar archivos se necesitaba navegar en la web para poder encontrarlos, así que con el paso del tiempo se crearon distintos programas para encontrar y organizar los contenidos de la web. En 1993, Matthew Gray creó un programa llamado Word Wide Web Wanderer para poder recorrer la web y poder recopilar los sitios, este programa es considerado la primera spider o araña para poder buscar todas la páginas y poder copiar todo ese contenido para su indexación [18]. Aunque el programa no logró recorrer todos los sitios pudo realizar una ejecución que arrojó datos coherentes para poder aportar en el crecimiento de la web [19]. Una araña es un programa que es capaz de rastrear todos los links desde un sitio web de inicio con el fin de extraer los datos.

Posteriormente en ese mismo año se creó el primer motor de búsqueda llamado JumpStation basado en web crawlers, este buscador únicamente podía extraer los títulos y encabezados de los sitios web, lo cual era simple pero de gran utilidad [20]. En la actualidad se encuentran múltiples APIs que permiten acceder y descargar datos de la web que están disponibles para todo el público, aunque aún no es posible encontrar toda la información que pueden proporcionar los sitios web, por lo que aún hay interés en este tema.

Existen dos técnicas distintas para poder realizar la extracción de información, Web Crawling y Web Scraping. Web Crawler es un software que navega automáticamente por la web y recupera cada una de las páginas web [21]. Mientras que el Web Scraping se define como el proceso de extracción y combinación de contenido de la web de forma sistemática [22].

En la actualidad hay múltiples razones para extraer información de la web: Como se mencionó anteriormente se hace el uso del web crawling con el fin de realizar la indexación de sitios web y poder crear un motor de búsqueda. El web Crawler visita diferentes sitios web con el fin de presentar el contenido disponible en la web de acuerdo con la búsqueda del usuario. También puede ser de gran utilidad para construir un conjunto de datos, a menudo se necesitan miles para construir, probar y entrenar modelos de aprendizaje automático. En algunos casos los datos ya se encuentran empaquetados para ser utilizados pero la mayor parte del tiempo es necesario crear los propios conjuntos de diversas fuentes, por lo que se puede recurrir a la extracción de información web para poder obtenerlos fácilmente. Otra aplicación puede ser el crear un comparador de precios y poder buscar productos o servicios vendidos a través de

distintos sitios web. Con estas técnicas se pueden obtener información valiosa para saber quién vende el producto, quien tiene los mejores precios o la disponibilidad. Se puede hacer esto periódicamente para monitorear los productos y servicios y poder ser comparados fácilmente [23].

Uno de los desafíos de la extracción de información es la diversidad de los contenidos de las páginas web, por lo que el uso de una biblioteca puede ser insuficiente, es por eso por lo que el uso de bibliotecas es uno de los enfoques más utilizados para web scraping, es posible el uso de múltiples lenguajes de programación con el que se esté más familiarizado.

En PHP podemos encontrar a Goutte la cual proporciona una API para realizar web crawling y web scraping de archivos HTML y XML[1] [24] y que Guzzle es una herramienta tipo cliente PHP la cual puede enviar solicitudes HTTP de una manera simple, cuenta con una interfaz para poder obtener gran cantidad de datos HTTP [25].

Respecto a Python existen dos bibliotecas principales: Beautiful Soup sirve para extraer datos de archivos HTML y XML. Proporciona formas para navegar, buscar y modificar el árbol de análisis [26] y Scrapy es un framework de alto nivel utilizado para web crawling y web scraping. Puede ser utilizado para múltiples propósitos desde minería de datos hasta monitoreo y pruebas automatizadas [27].

Adicionalmente está Selenium el cual es un framework de optimización de pruebas, el cual proporciona extensiones para emular la interacción del usuario con los navegadores. Utiliza el protocolo WebDriver para controlar un navegador web [28]. Selenium es útil para que el HTML de una página web dinámica se ejecute en un navegador, ver los valores correctos y posteriormente poder capturar esos valores mediante web scraping. Se hace uso de esta biblioteca ya que se encontraron algunas páginas a las que es más complicado entrar debido a que cuentan con un código externo, por ejemplo, JavaScript y es necesario ejecutarlo para poder visualizar correctamente la página, en caso contrario no se podrán extraer los links que se encuentren en esas páginas. Selenium puede simular esa ejecución del código para posteriormente poder extraer los datos necesarios.

Para el proyecto se eligió utilizar Beautiful Soup, Scrapy y Selenium ya que como se mencionó anteriormente, RASA cuenta con acciones que permiten hacer múltiples funciones, en este caso es necesario utilizar Python ya que las acciones deben de ser programadas en ese lenguaje por lo que sería más sencillo incorporar a RASA, además de que cuentan con una gran documentación. Beautiful Soup y Scrapy son dos bibliotecas que son utilizadas para lo mismo, pero se consideró el uso de ambas para hacer distintas pruebas y hacer la elección de la más adecuada para el proyecto.

[1] HTML y XML son lenguajes de diseño de documentos y especificación de hipervínculos. Definen la sintaxis y la ubicación de direcciones incrustadas especiales que no se muestran en el navegador, pero le indican cómo mostrar el contenido del documento, incluido el texto, las imágenes y otros medios de soporte [39].

En la propuesta de arquitectura de este proyecto se hace uso de distintas componentes extraídos de la literatura para poder obtener mejores interacciones del usuario con el bot, además de contar con respuestas más efectivas.

1.3 Objetivo general

Diseñar e implementar una arquitectura de chatbot con elementos de búsqueda externa.

1.4 Objetivos específicos

- Incorporar en la arquitectura del chatbot elementos de búsqueda externa para ampliar la cantidad de respuestas satisfactorias para el usuario y evitar capturar una gran cantidad de información en la base de datos
- Concentrar en el diseño los componentes más utilizados en las arquitecturas propuestas en la literatura

1.5 Estructura

A continuación, se describe cómo está estructurado este trabajo de investigación: en el Capítulo 2, se presentan los resúmenes de antecedentes más relevantes para poder realizar la investigación, los cuales permitirán profundizar a la arquitectura. En el Capítulo 3, se describe la arquitectura ideal propuesta y la arquitectura para el chatbot realizado. En el Capítulo 4, se analizan los resultados obtenidos y por último en el Capítulo 5 se muestran las conclusiones y el trabajo futuro.

Capítulo 2

2. Estado del Arte

En la actualidad hay múltiples trabajos relacionados con los chatbots, de los cuales se pueden mencionar algunos de ellos que fueron relevantes para esta investigación.

En [2] los autores presentan un diseño de una arquitectura general, la cual incluye todas las componentes que los autores consideran cruciales e importantes para el desarrollo de un chatbot (Figura 14): Interfaz de usuario, servicios cognitivos, clasificador de intención, identificador de entidades, componente de manejo de dialogo, base de datos y generador de respuesta. Algo muy similar ocurre en [29] donde se presenta una arquitectura simple con el uso de técnicas de PLN, CLN y GLN (Figura 15). Las componentes de esta arquitectura son: capa de presentación o interfaz de usuario, capa de predicción o extractor de entidades e intenciones, base de datos y GLN o generador de respuesta. Las arquitecturas antes mencionadas son chatbots sin ningún propósito en específico, pero a continuación se hablará de las arquitecturas que fueron creadas para un propósito.

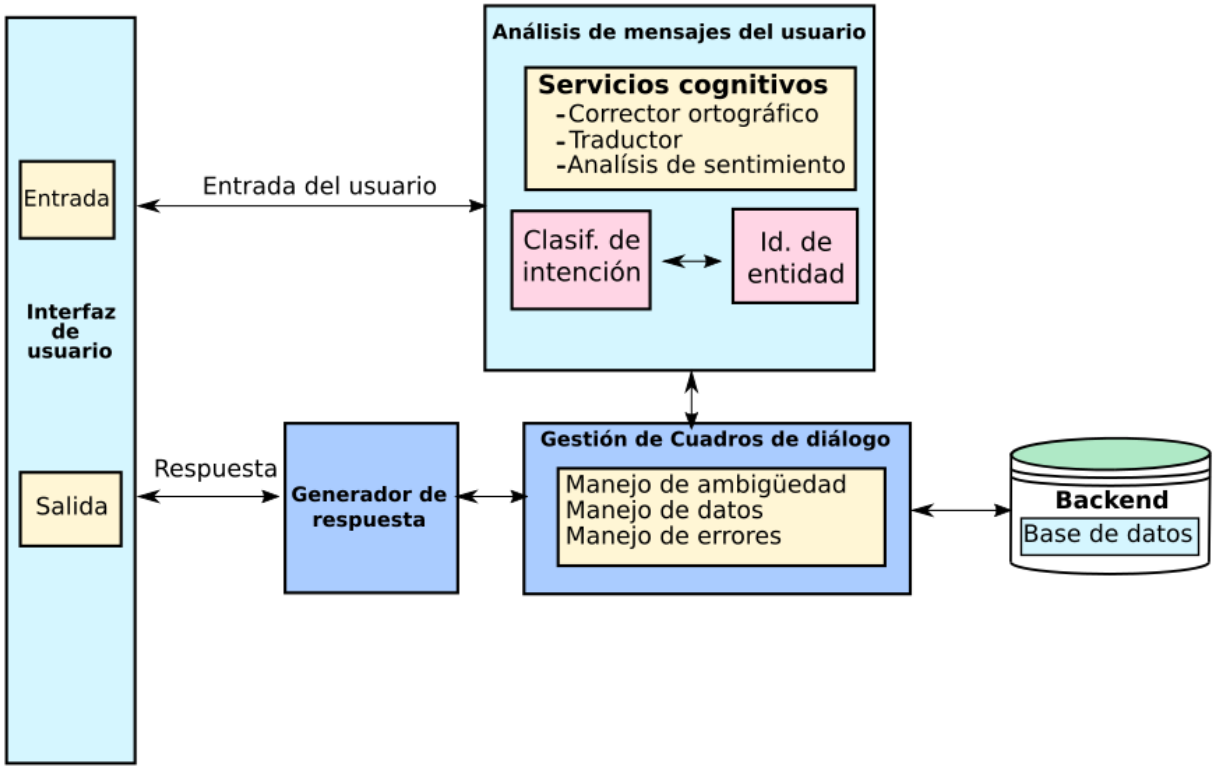

Figura 14. Arquitectura traducida de [2].

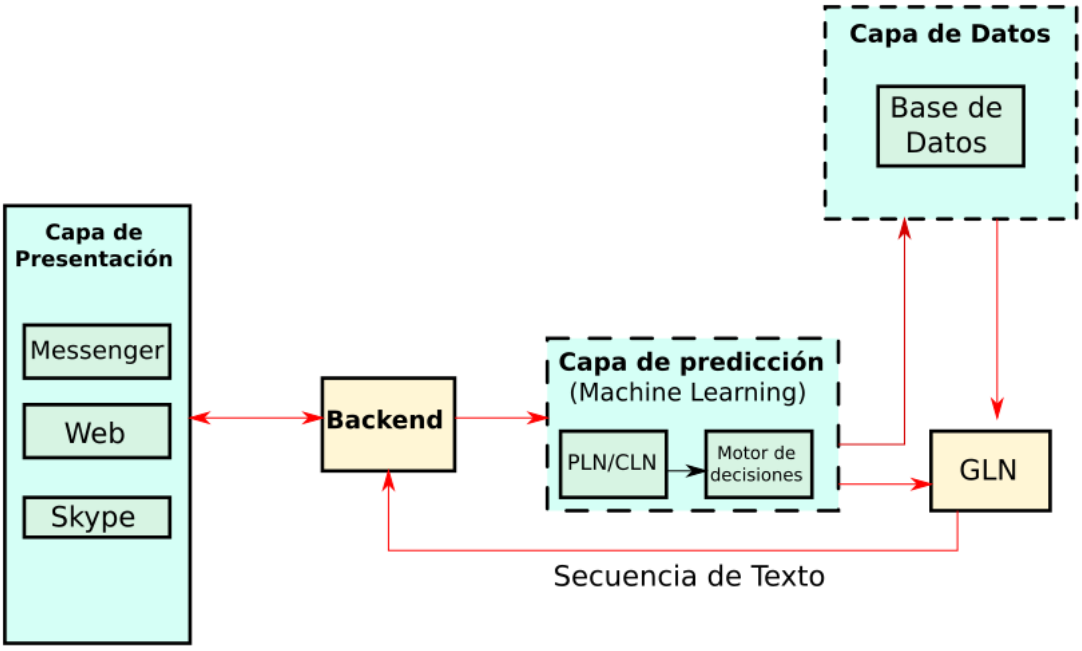

Figura 15. Arquitectura traducida de [29].

En [30] se describe la arquitectura de un chatbot que enseña a los usuarios sobre la Diabetes y en caso de no ser encontrada la información solicitada por el usuario en la base de datos, se procede a realizar una extracción de información en páginas web de confianza, la arquitectura consta con los siguientes elementos (Figura 16): interfaz de usuario, base de datos local, base de datos externa, generador de respuesta por coincidencia de patrones y base de datos de información por cada usuario nuevo, existente o registro. Otro de los chatbots con fines médicos es el presentado en [31], el cual proporciona información sobre el cáncer. Este chatbot utiliza Beautiful Soup para extraer la información de la web para ser almacenados en una base de datos local y posteriormente ser proporcionado al usuario. Los componentes de esta arquitectura son: interfaz de usuario vía Facebook, identificador de entidades e intenciones, generador de respuesta y base de datos extraída de la web (Figura 17). Otro chatbot que utiliza la extracción de información es el descrito en [32], el cual es similar a ELIZA, toma la entrada del usuario para reformular dicha expresión en forma de pregunta. Adicionalmente se utiliza la entrada del usuario para buscar la información recopilada en la web y encontrar alguna información que pueda ser relevante para el usuario. Esta arquitectura consta de: interfaz de usuario, base de datos interna y externa y generador de respuesta.

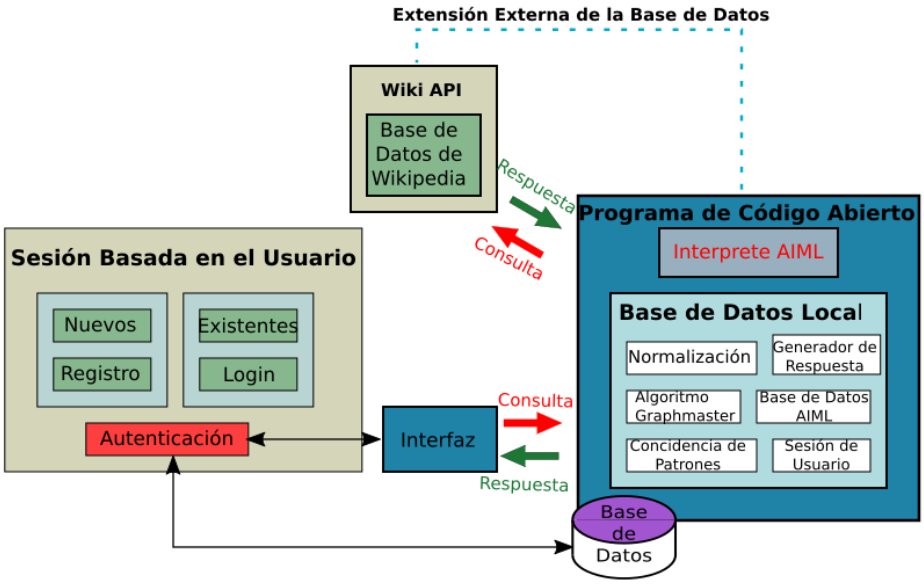

Figura 16. Arquitectura traducida de [30].

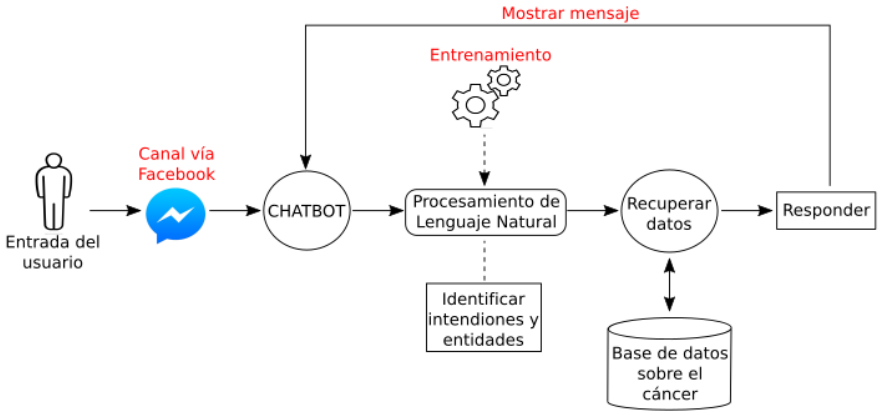

Figura 17. Arquitectura traducida de [31].

De las aplicaciones más comunes para los chatbots es en la educación, en [33] se presenta la arquitectura de un chatbot como una herramienta de aprendizaje. El chatbot se encarga de realizar preguntas al usuario para evaluar el aprendizaje sobre un tema, y determinar si el usuario ha aprendido para poder pasar a la siguiente pregunta, o en caso contrario le presentará información para reforzar el tema. La arquitectura cuenta con los siguientes elementos: interfaz, base de datos de respuestas, base de datos de preguntas, control de flujo de conversación, generador de preguntas y generador de respuestas. Otro tipo de chatbots educativos tienen un enfoque de apoyo a los estudiantes [34]. En este caso se presenta una arquitectura realizada con PLN y la representación ontológica del conocimiento para proporcionar a partir de las peticiones de los usuarios, lo que realmente requieren, aunque no se indique de forma clara o precisa, el chatbot puede ser capaz de inducirlo. La arquitectura (Figura 18) consta de las siguientes componentes: una base de datos para la ontología, los datos de los

usuarios y de preguntas frecuentes, interfaz de usuario servicios externos web, una máquina de inferencia, manejador de contexto y verificador de calidad de interacción.

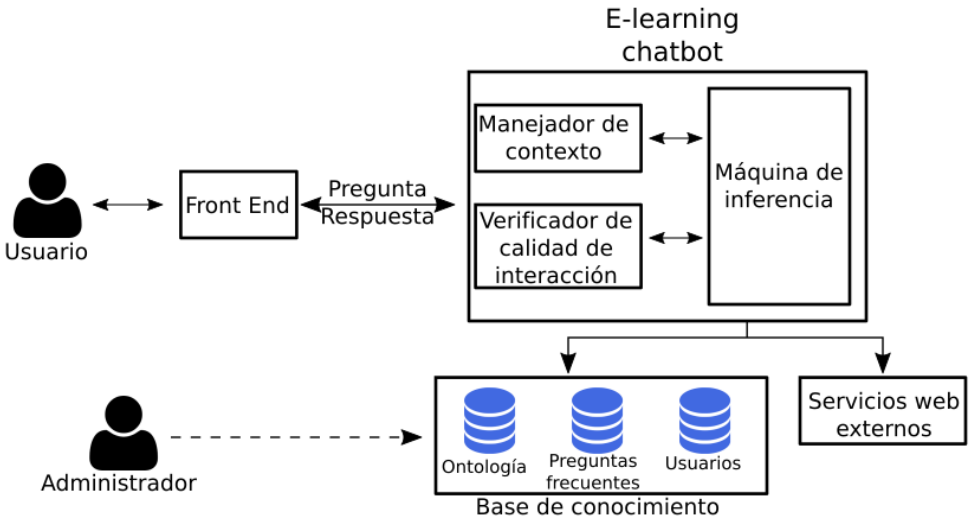

Figura 18. Arquitectura traducida de [34].

Otro de los propósitos que se le puede dar a un chatbot es el uso recreativo o entretenimiento. En [35] se propone la arquitectura para un chatbot que tiene como objetivo dar información sobre los partidos relacionados con la Liga Española. Esta arquitectura hizo uso de RASA y la extracción de información de la web para las respuestas. La arquitectura (Figura 19) consta de los siguientes elementos: interfaz de usuario usada de un servicio externo slack, extractor de entidades y clasificador de intenciones, gestor de diálogo, generador de respuesta basado en recuperación base de datos interna y base de datos externa (datos extraídos de wikidata). Finalmente, en [13] se muestra un chatbot el cual está integrado a un dispositivo que se les proporciona a los usuarios durante la visita a un parque arqueológico en Italia. Este chatbot contiene información sobre la zona arqueológica y es capaz de responder a las preguntas sobre el parque usando técnicas de PLN. La arquitectura (Figura 20) consta de: interfaz de usuario, analizador de semántica o analizador de mensajes de usuario, administrador de flujo de trabajo o administrador de flujo de conversación, módulo de contexto, base de datos interna y servicios externos.

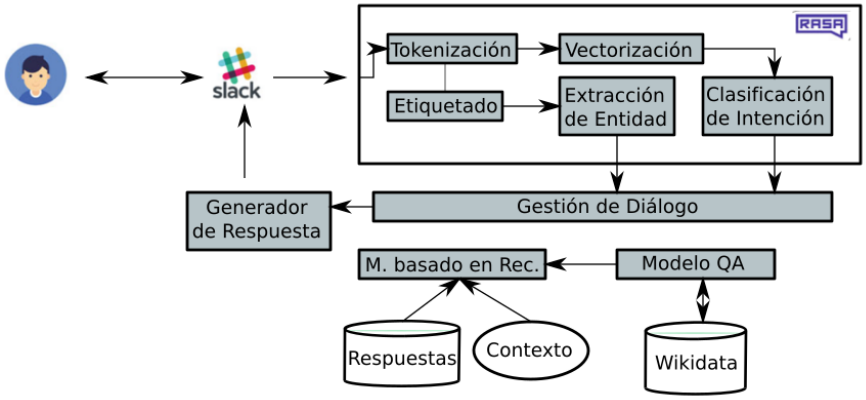

Figura 19. Arquitectura traducida de [35].

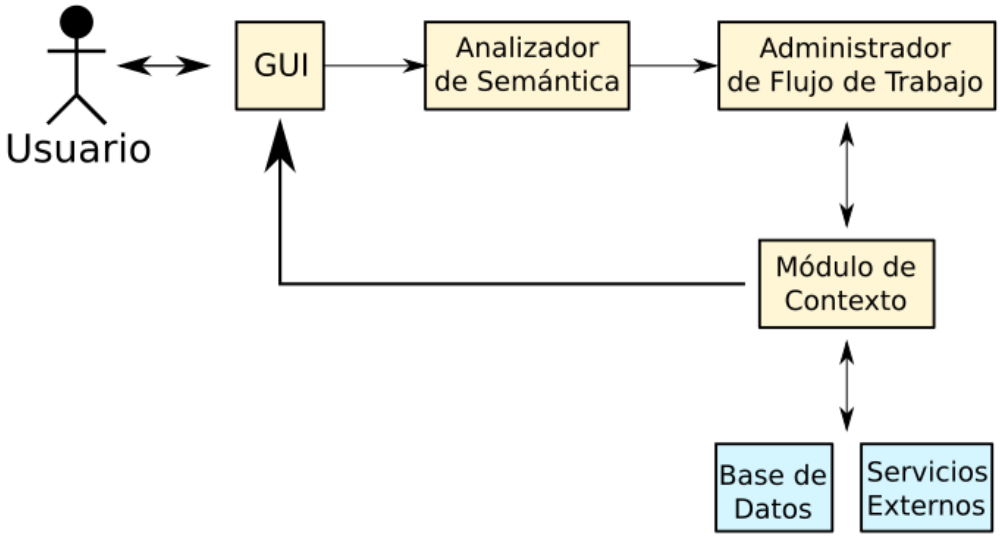

Figura 20. Arquitectura traducida de [13].

El resumen de las arquitecturas antes descritas y la comparación de los componentes entre cada una de ellas se muestra en la Tabla 2. A continuación, se describen más a detalle cada una de las componentes antes mencionadas en las arquitecturas.

- Interfaz de usuario – chatbot: Medio por el cual el usuario se comunica con el chatbot. Entrada y salida de mensajes
- Análisis de mensajes del usuario: Es utilizado para encontrar la intención y extraer las entidades de los mensajes del usuario.
- Servicios cognitivos: Está compuesto por
    - Gestor ortográfico: Se procesa el mensaje del usuario quitando signos de puntuación, eliminando espacios extras y convirtiendo cada letra en minúsculas
    - Análisis de sentimiento: Detecta la opinión del usuario, ya sea positiva o negativa

- Gestor de dialogo: Este módulo controla y actualiza constantemente el contexto de la conversación. En caso de que el chatbot no sea capaz de determinar la información necesaria para dar una respuesta, solicita información adicional al usuario. Se tienen dos módulos:
    - Manejo de ambigüedad: Se encarga de preguntar al usuario por más información, iniciar una conversación nueva o dar una respuesta simple y tratar que el usuario este satisfecho con la respuesta
    - Manejo de errores: Hace frente a situaciones inesperadas, trata de corregir errores para garantizar el correcto funcionamiento del chatbot
- Base de datos interna: Conjunto de información para ser utilizada por el chatbot y poder generar las respuestas. Está contenida en el chatbot por default y es creada para contestar preguntas básicas
- Generador de respuesta: Produce las respuestas usando uno de los modelos disponibles:
    - Basado en reglas
    - Basado en recuperación
    - Generativo
    - Híbrido
- Servicios externos: está conformado por:
    - Servicios de mensajería como WhatsApp, Facebook, Line, etc. Esto sustituye a la interfaz de usuario
    - Base de datos externa: Es creada por medio de la extracción de información de la web. En caso de que la respuesta no se encuentre en la base de datos interna, se realizará la búsqueda en la base de datos externa.

Tabla 2. Resumen de arquitecturas.

	Componentes							
Autores	Interfaz	Base de datos	Servicios cognitivos	Gestor de dialogo	Generador de respuesta	Detección de intención	Identificador de entidades	Base de datos externa
E. Adamopoulou and L. Moussiades [2]	Si	Si	Si	Si	Si	Si	Si	
S. Abhishek, K. Ramasubramanian, and S. Shivam [29]	Si	Si		Si	Si			
S. Hussain and A. Ginige [30]	Si	Si			Si			Si
M. Manilal, A. Mathew, and B. Babu [31]	Si	Si		Si	Si	Si	Si	Si
H. Collins and S. Alam [32]	Si	Si			Si			Si
R. Bathija, P. Agarwal, and R. Somanna [33]	Si	Si		Si	Si		Si	
C. Fabio, M. Lombardi, F. Colace, and F. Pascale [34]	Si	Si		Si	Si	Si		Si
C. Segura, Á. Palau, J. Luque, M. Costa-Jussá, and R. E. Banchs [35]	Si	Si			Si			Si
M. Lombardi, F. Pascale, and D. Santaniello [13]	Si	Si		Si	Si	Si	Si	

Otro de los temas de interés para esta investigación es la extracción de la información más relevante en el menor tiempo posible. Algunos de los algoritmos que son utilizados para extraer, ordenar y clasificar los resultados de información son: búsqueda primero en amplitud, A*, A* adaptativo, Best First Search, Fish search, entre otros [36]. Los tres primeros algoritmos antes mencionados son de los más utilizados en el web crawling para ser implementados en los motores de búsqueda, los cuales se explican a continuación.

El algoritmo de búsqueda de primero en amplitud es el más simple de todos los algoritmos de web crawling. Comienza con un enlace y sigue atravesando los enlaces sin tener en cuenta ningún conocimiento del tema que se ha buscado. Este algoritmo es conocido como un algoritmo de búsqueda ciega [37].

El algoritmo de A* combina las característica de la búsqueda de costo uniforme y la búsqueda heurística para calcular de manera eficiente las búsquedas relacionadas. El costo asociado con un nodo es: f(n) = g(n) + h(n), donde g(n) es el costo del camino desde el estado inicial al nodo n y h(n) es la estimación heurística del costo del camino desde el nodo n hasta el nodo objetivo. f(n) estima el costo más bajo de cualquier ruta. Por otro lado, el algoritmo de A* adaptativo funciona con una heurística informada. Con cada iteración actualiza la relevancia de la página y lo utiliza para el siguiente recorrido [36].

En la Figura 21 se muestra una comparación de los tiempos de ejecución de distintos algoritmos de web crawling, en términos de tiempo de búsqueda, el algoritmo de búsqueda en amplitud toma valores más elevados, mientras que A* y Best First Search dan mejores resultados ya que requieren menos tiempo de ejecución, además de ser constantes esos tiempos. Se puede observar que existen diferentes algoritmos para la búsqueda de web crawling, algunos algoritmos pueden ser fácil de implementar, pero resultan no ser eficientes.

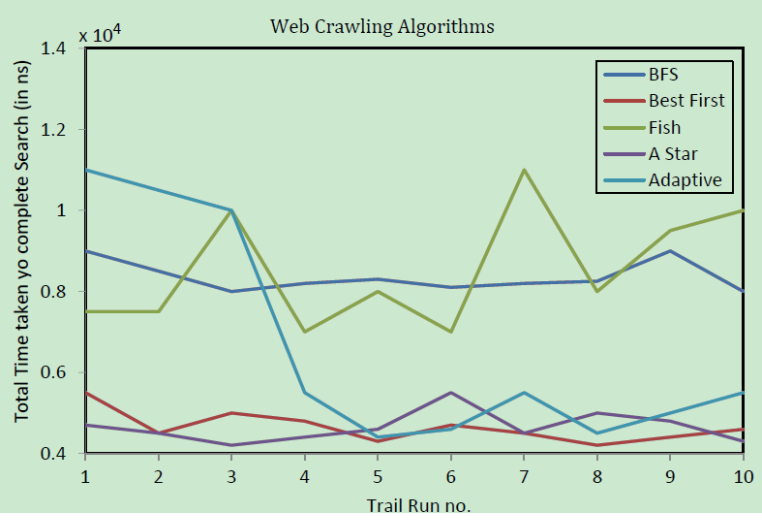

Figura 21 Tiempo de búsqueda de distintos algoritmos de web crawling. Extraída de [36].

Capítulo 3

3. Arquitectura

En este capítulo se detalla la arquitectura propuesta para el proyecto.

3.1 Descripción de la arquitectura

Tomando en consideración los componentes encontrados en la revisión de la literatura se concluyó que las características para una arquitectura ideal son las mostradas en la Figura 22. La arquitectura se dividió en dos tipos de elementos, los fundamentales y los deseables ya que algunos de los elementos son considerados de mejora, pero antes de eso es necesario tener un chatbot base, es por eso que se definen los elementos fundamentales.

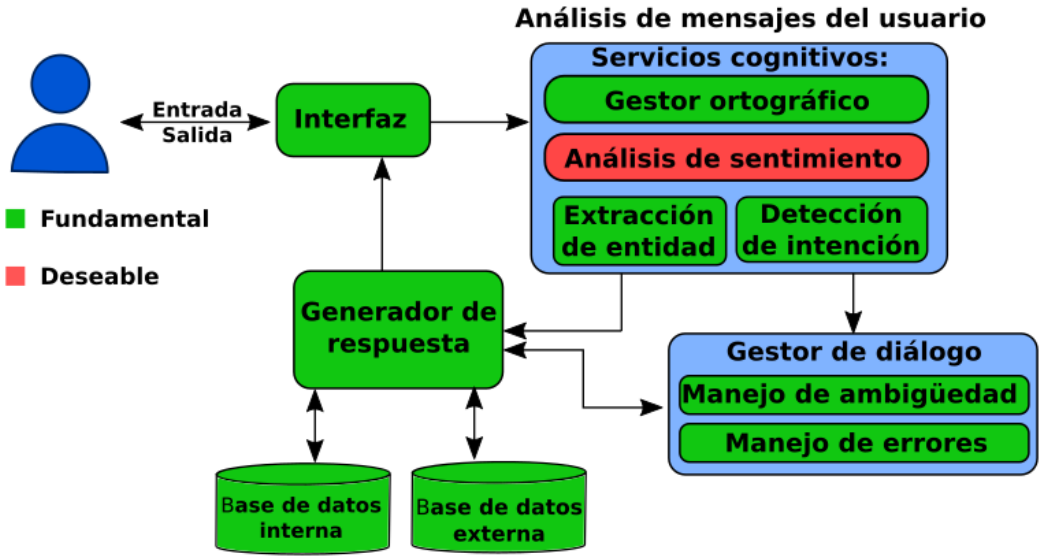

Figura 22. Arquitectura propuesta.

Los elementos deseables incluyen el análisis de sentimientos, el cual no se implementará ya que para eso se necesita un chatbot base que ya cuente con los elementos fundamentales y se considera un elemento de mejora continua. El análisis de sentimientos puede tener múltiples usos, por ejemplo, sirve para determinar el estado de ánimo de una persona. En este caso se desea utilizar el análisis de sentimiento de la siguiente manera: se le preguntará al usuario si la respuesta dada previamente cumple con sus expectativas, en caso de que la respuesta del usuario sea negativa se tomará como referencia para próximas mejoras del chatbot.

Los elementos fundamentales son todos los elementos que se consideran importantes para el desarrollo de un chatbot inicial.

En la Figura 23 se muestran los componentes de la arquitectura, cuyo funcionamiento se describe a continuación.

Se recibe el mensaje por medio de la interfaz para ser procesado en el módulo de análisis de mensajes, en donde se quitarán acentos, signos de puntuación, etc. Se procede a extraer la intención y entidades. Si el chatbot no fue capaz de comprender el mensaje por completo, el módulo de manejo de ambigüedad y el módulo generador de respuesta se encargarán de generar una pregunta al usuario para poder extraer más información. En caso contrario el módulo generador de respuesta con ayuda de la base de datos interna procederá a generar la respuesta o si la respuesta no se encuentra en la base de datos interna, se realizará la búsqueda de información en la base de datos externa.

Para mejorar el desempeño del chatbot se usará una base de datos interna que se actualizará empleando técnicas de Web Scraping. Con el uso de la extracción de la información de páginas web se cree que se podrán obtener mejores resultados en las respuestas y será capaz de contestar más preguntas. En el caso de esta arquitectura no se hará uso de una base de datos externa, si no que los datos extraídos de la web se utilizarán para alimentación de la base de datos interna. El chatbot usará un modelo de generación de respuestas híbrido en donde se combinarán los modelos basado en reglas y en recuperación. Se hará uso de RASA para la implementación del chatbot. El modelo basado en reglas será utilizado para contestar algunos mensajes simples como los saludos y despedidas, mientras que el modelo basado en recuperación se utilizará para cuando el usuario solicite alguna información específica, la respuesta se generará en base con lo solicitado.

En la Figura 23 se puede observar que se eliminó el componente de manejo de errores, en el caso de RASA no maneja el termino de “errores” y es considerado como ambigüedad, así que se eliminó ese elemento, pero es incorporado en el manejo de ambigüedad. Si el chatbot no ha entendido alguna solicitud del usuario se puede considerar que hubo un error y no será capaz de responder correctamente, es por eso que el chatbot le indicará al usuario que no fue capaz de entender la pregunta y le solicitará que la vuelva a hacer con el fin de darle una respuesta apropiada al usuario. La manera en que el chatbot enfrentará ese tipo de inconvenientes se considera manejo de ambigüedad.

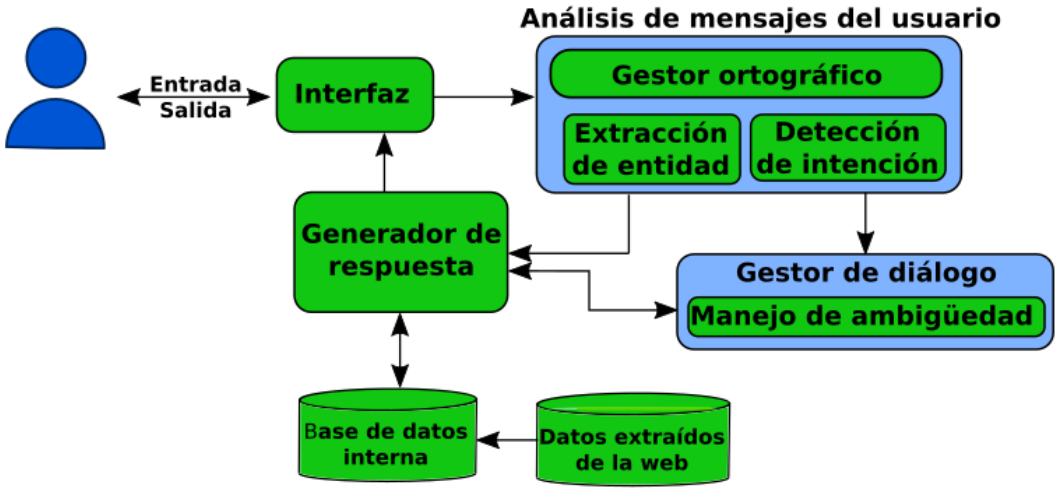

Figura 23. Arquitectura final del chatbot.

3.2 Implementación

En esta sección se describirá la implementación de cada elemento de la arquitectura mostrada en la Figura 24.

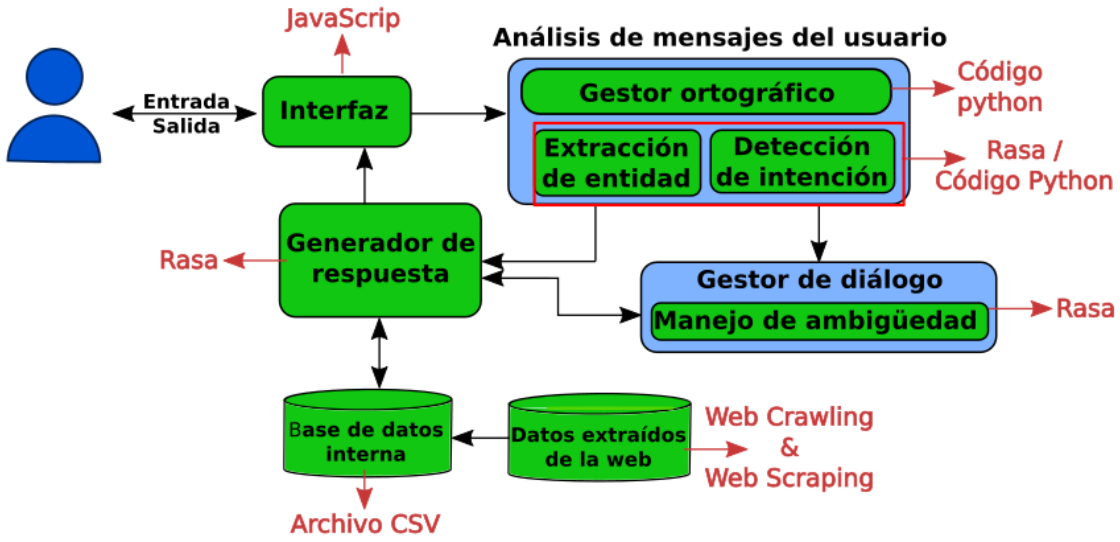

Figura 24. Implementación de la arquitectura.

Para la interfaz se hizo uso del código en html, css y JavaScript extraído de [38] con algunas modificaciones para ser adaptado al proyecto. Mientras que el gestor ortográfico fue realizado con funciones propias de Python.

Para la extracción de entidades y detección de intenciones RASA cuenta con diferentes tipos de clasificadores para este propósito, en este caso es utilizado el

clasificador DIETClassifier (Dual Intent Entity Transformer). La arquitectura está basada en un transformador que es compartido para ambas tareas.

Adicionalmente se creó un extractor de entidades con el código de Python, el cual elimina todas las palabras que estén de más como preposiciones o palabras que no le aporte nada al mensaje del usuario. En la Figura 25 se muestra un ejemplo del proceso que se realiza para la extracción de entidades por medio del código de Python. Se hace uso de una biblioteca de stopwords de nltk para eliminar las palabras que no aportan nada al mensaje y también se hace uso de una lista donde se incluyen las palabras que no son necesarias como: necesito, información, universidad. El pseudocódigo del extractor se muestra en el Anexo 1.

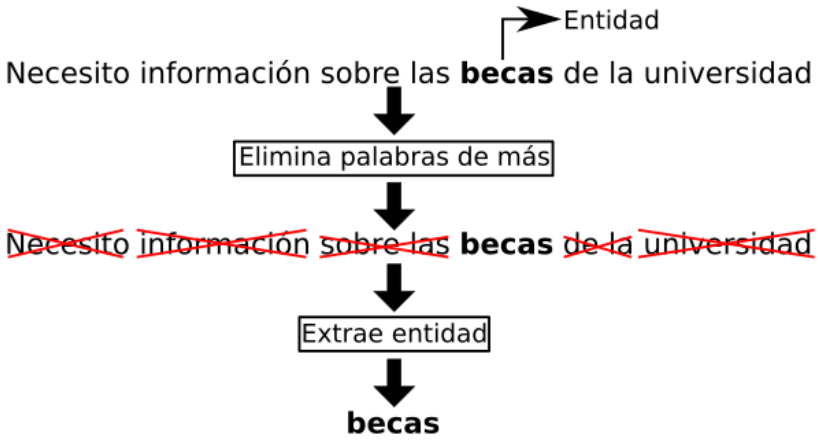

Figura 25. Extracción de entidades con código Python.

De igual forma con RASA se puede hacer la implementación del manejo de ambigüedad. Se puede hacer de dos maneras distintas, puede ser por acciones o bien con un mensaje predefinido. En este caso se hizo uso de un mensaje predefinido y cuando no se detecte correctamente la intención del mensaje le indicará al usuario que no entendió el mensaje y que trate de reformular su pregunta.

La base de datos interna es un archivo CSV el cual es consultado por el chatbot cada vez que el usuario realiza una pregunta. El archivo CSV no cuenta con ninguna respuesta predefinida, solo con los datos extraídos de la web, para lo cual se emplean dos técnicas, Web Scraping y Web Crawling.

Finalmente, el generador de respuesta se realiza con RASA el cual cuenta con diferentes maneras de generar la respuesta, en este caso se hace uso de TEDPolicy (Transformer Embedding Dialogue Policy) la cual toma en cuenta las intenciones y entidades actuales y pasadas tanto del usuario como del chatbot, para elegir la acción correcta a realizar en el paso siguiente. Adicionalmente cuenta con algunos mensajes predefinidos que son utilizados como un modelo basado en reglas. Esto da como resultado un generador de respuesta híbrido (generativo y basado en reglas).

Capítulo 4

4. Pruebas y Resultados

En este capítulo se muestran las pruebas realizadas y finalmente se comentan los resultados.

4.1 Pruebas extracción web

Las primeras pruebas se dividen en dos casos con el fin de determinar las bibliotecas adecuadas para la extracción. Verificar cuáles eran las más rápidas y lograban una mayor extracción de los enlaces y una búsqueda más rápida. Estas pruebas fueron realizadas en la página del posgrado pcyti.izt.uam.mx buscando “Areli”.

Para estas pruebas se utilizaron dos bibliotecas de cuatro maneras distintas: Scrapy (S), Beautiful Soup (BS), Scrapy & Beautiful Soup (S & BS) y Beautiful Soup con hilos (BS con hilos).

a) Caso I: Extraer datos, realizar la búsqueda y presentar respuesta. Este proceso se repetía cada que el usuario realizaba una petición. Los resultados de esta prueba se muestran en la Figura 26, donde se puede observar que Beautiful Soup tiene un sesgo hacia arriba del segundo cuartil, además de que se tiene un mayor rango de tiempo mientras que las demás distribuciones el rango es menor y son asimétricas. La biblioteca más rápida fue Scrapy con un tiempo de 21.57 seg.

Se realizaron un total de 30 pruebas en donde siempre se buscó la misma palabra para asegurarnos de que los tiempos fueran similares. El algoritmo se detenía una vez que la palabra era encontrada. Los resultados de los tiempos de búsqueda de este algoritmo se muestran en la Tabla 3 del Anexo 2.

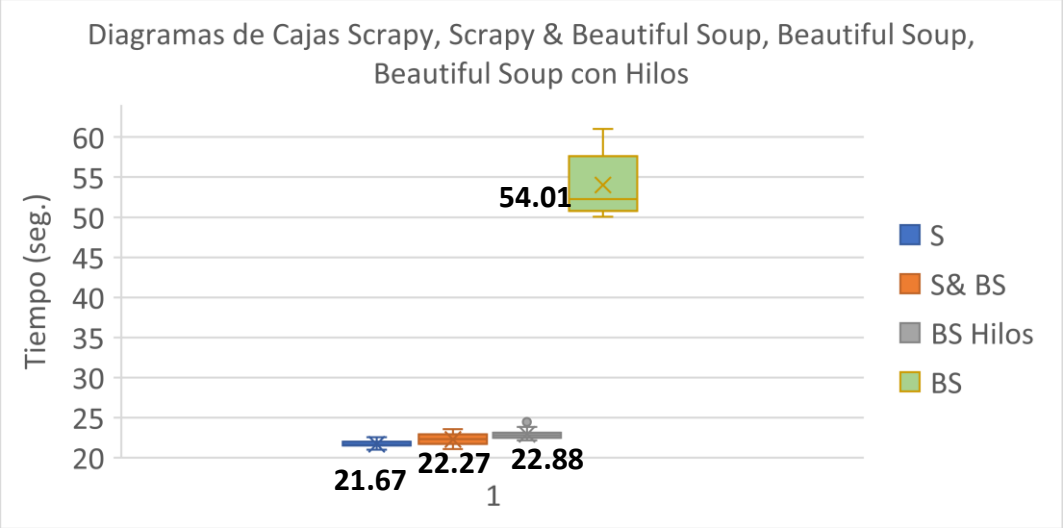

Figura 26. Diagramas de cajas del caso I.

b) Caso II: Extraer datos, guardar datos en CSV, realizar búsqueda y generar las respuesta. En este caso extraer los datos de la página del posgrado y guardarlos se realiza una sola vez, para realizar la búsqueda en la base de datos creada y generar la respuesta se realiza cada que el usuario genere una pregunta. Los resultados de este caso se muestran en la Figura 27, se puede observar que los tiempos son muy similares al caso anterior, pero se está considerando el tiempo de descarga de la información. Los tiempos de búsqueda eran demasiado pequeños por lo que son despreciables. De igual forma, la biblioteca más rápida en extraer los datos fue Scrapy. Es importante destacar que los tiempos también dependen del tipo de equipo utilizado y la conexión de internet.

Se realizaron un total de 30 pruebas en donde siempre se buscó la misma palabra para asegurarnos de que los tiempos fueran similares. El algoritmo se detenía una vez que la palabra era encontrada. Los resultados de los tiempos de búsqueda se muestran en el Anexo 2 Tabla 4.

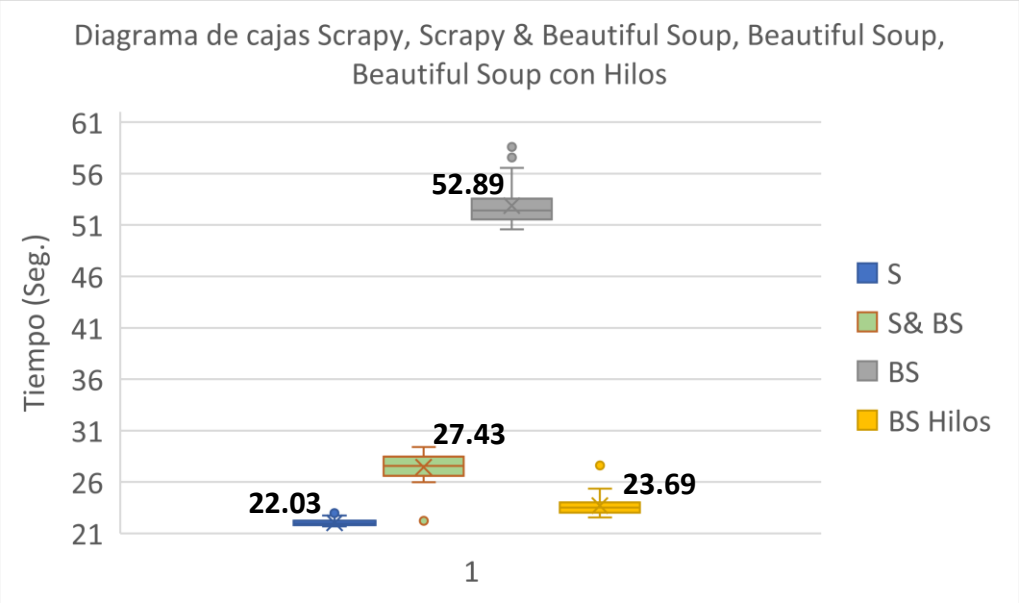

Figura 27. Diagramas de cajas del caso II.

Una vez teniendo los resultados anteriores se hicieron pruebas para la extracción de datos, en la Figura 28 se muestran los resultados de la cantidad de datos extraídos por cada una de las bibliotecas utilizadas. Se puede observar que Scrapy obtuvo la mayor cantidad de enlaces.

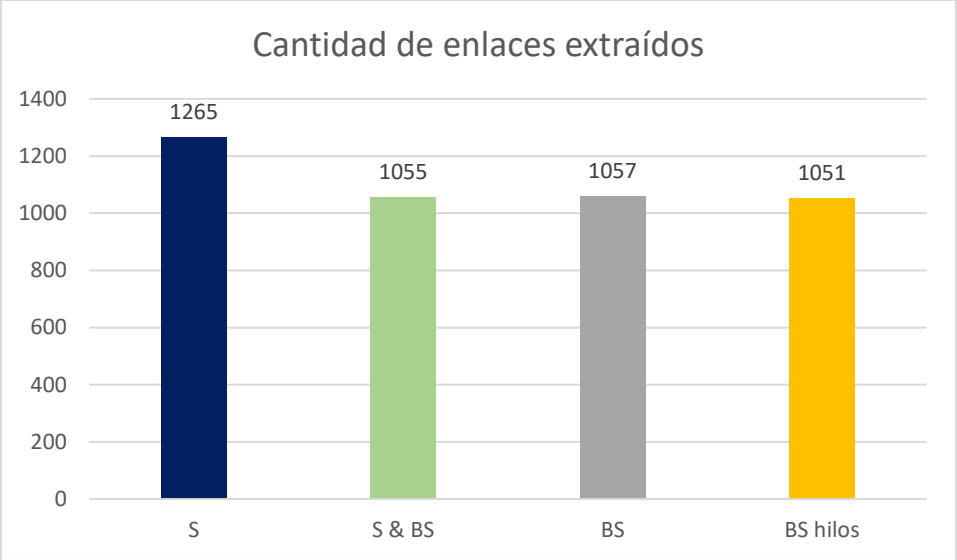

Figura 28. Cantidad de enlaces extraídos.

Con los resultados obtenidos anteriormente se eligió hacer la implementación del caso II con la biblioteca Scrapy en la página principal de la UAM Iztapalapa y continuar con la extracción de enlaces en todo el dominio izt.uam.mx. Al realizar la implementación, se identificaron tres problemas:

1. Enlaces a los que era necesario agregar “wordpress” o “wp” para poder ingresar y continuar con la búsqueda.
2. Páginas dinámicas a las que es más complicado entrar, para este caso se utilizó la biblioteca Selenium, la cual puede emular la interacción del usuario con la página, de esta manera es posible abrirla de manera correcta y posteriormente extraer la información.
3. Páginas con dominio distinto. Por ejemplo, la página de sistemas escolares, la cual es importante para alimentar al chatbot de información sobre trámites o fechas importantes para los alumnos. En este caso se agrega a una lista junto con la página principal para realizar la búsqueda y extracción de enlaces.

Para resolver los problemas antes mencionados y garantizar la correcta extracción de la mayor cantidad de enlaces se realizó un programa en Python, basado en una búsqueda en profundidad. El pseudocódigo del extractor de información se muestra en el Anexo 1. En total se descargaron 5395 enlaces.

Para realizar la evaluación y pruebas del chatbot, se dividió en tres bloques distintos (Figura 29): búsqueda y descarga de información, búsqueda en la base de datos y presentación de la información. Los resultados de las pruebas se muestran en las siguientes secciones.

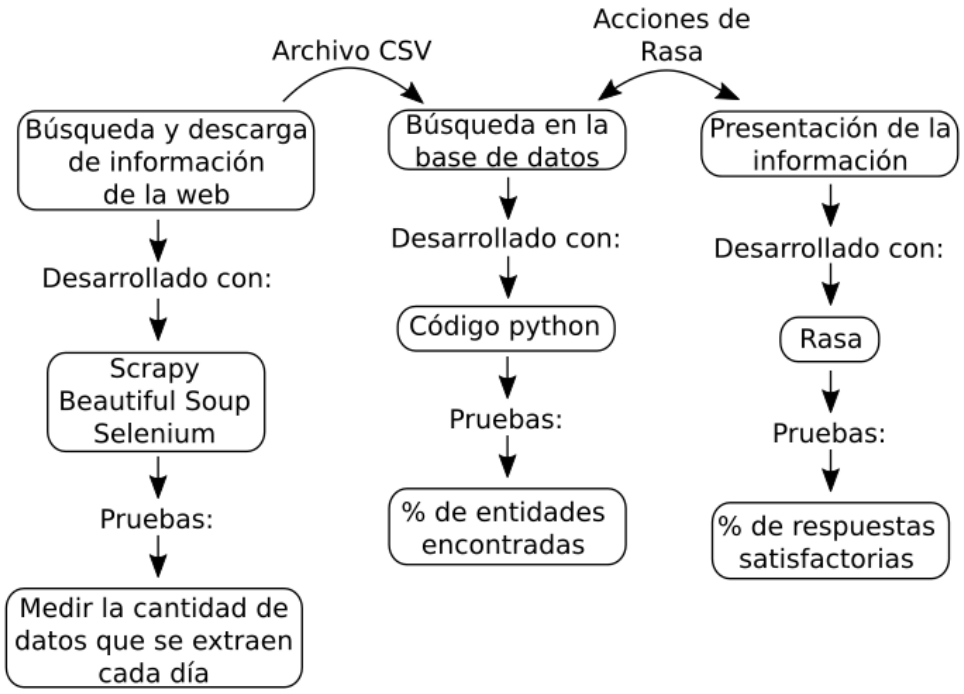

Figura 29. Bloques del chatbot propuesto.

De acuerdo con la Figura 29 el funcionamiento del chatbot se dividió en los siguientes bloques para realizar distintos tipos de pruebas.

- Búsqueda y descarga de información: Se descargan los enlaces de las páginas de la universidad con el fin de elegir la frecuencia con la que se debe de actualizar los datos
- Búsqueda en la base de datos: Una vez extraídos los enlaces son guardados en un archivo CSV el cual es necesario para realizar la búsqueda de la información. Esta prueba es importante para medir la efectividad del buscador implementado
- Presentación de información: Se muestra la respuesta encontrada en la base de datos. Las pruebas se realizan para determinar la efectividad y la calidad de las respuestas de ambos chatbots realizados (extractor de entidades con RASA y extractor de entidades con Python)

En las siguientes secciones se explica más detalladamente el proceso de cada una de las pruebas.

4.2 Pruebas de descarga de información

Estas pruebas fueron realizadas con el fin de verificar el número de enlaces que son extraídos por día y de esta manera elegir qué días o cada cuántos días se debe de extraer la información para alimentar el chatbot. En la Figura 30 se muestran los resultados de extracción de información de 30 días. Se puede observar que hay cambios significativos en los primeros 15 días de la prueba y en los siguientes 15 días se muestra un cambio constante, esto puede ser debido al periodo vacacional en el que se encontró la universidad los últimos diez días de prueba. En los primeros 15 días en donde se

encontraron cambios significativos puede deberse al cercano periodo de actividad de las primeras semanas del trimestre. Los resultados del número de enlaces extraídos se muestran en la Tabla 5 Anexo 2.

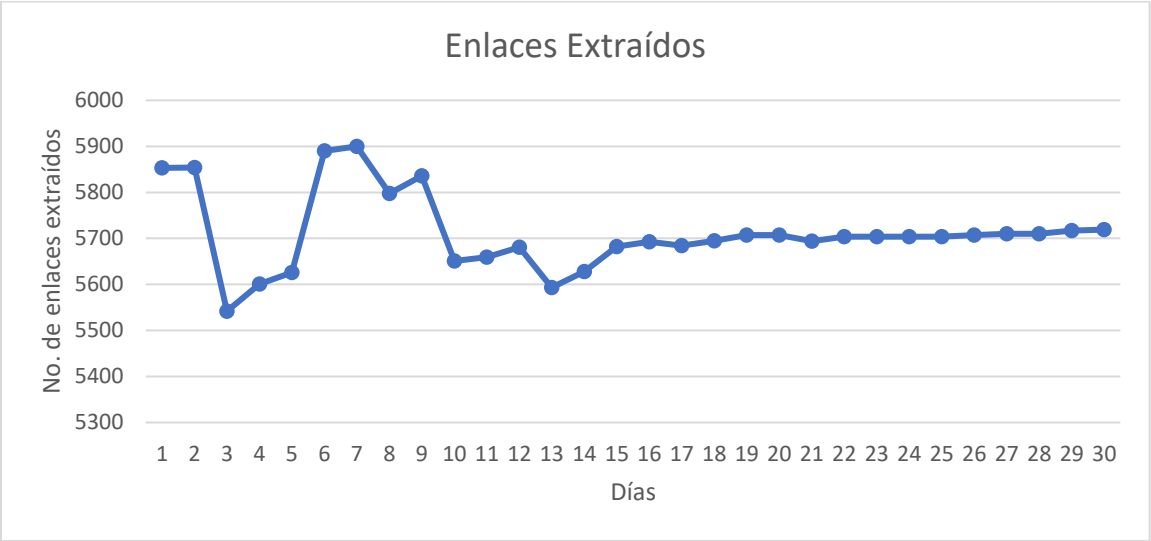

Figura 30. Enlaces extraídos.

4.3 Pruebas de búsqueda

La búsqueda de las preguntas de la base de datos que contiene los enlaces extraídos se realiza por medio de una comparación de caracteres, por lo que una vez extraídas las entidades del mensaje del usuario se procede a quitar los signos de puntuación y pasar las palabras a minúsculas de tal forma que se pueda realizar una búsqueda más acertada. Estas pruebas se realizaron haciendo 65 preguntas al chatbot de tal forma que todas ellas contenían entidades disponibles en la base de datos. En la Figura 31 se muestran los resultados obtenidos, donde se observa que el 95% de las entidades fueron encontradas, mientras que el 5% no fue localizado.

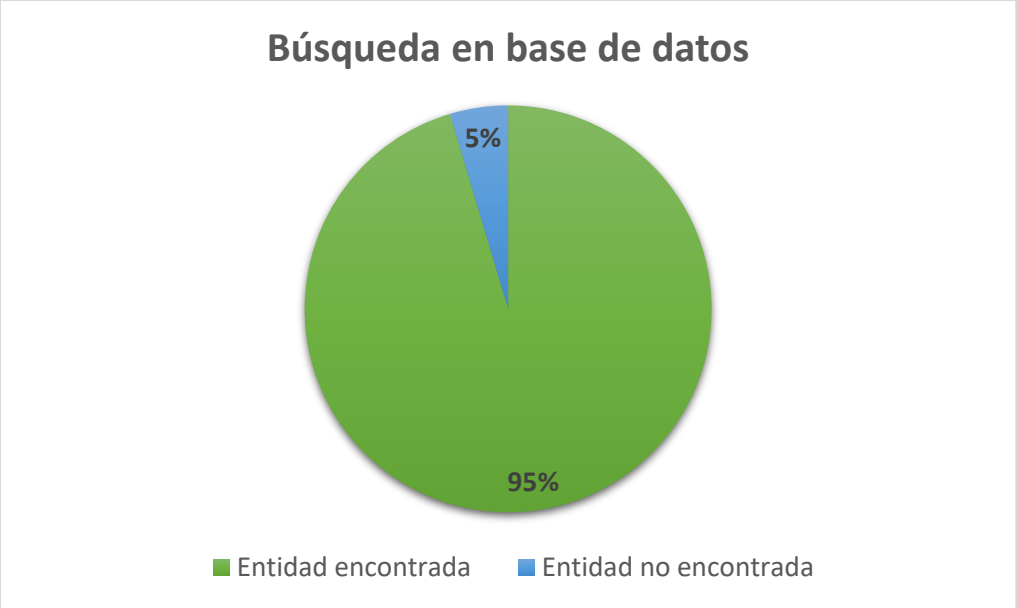

Figura 31. Búsqueda en base de datos.

El problema con esta técnica de búsqueda es que, si el usuario no escribe correctamente la palabra, esta nunca será encontrada, aunque si este presente en la base de datos. Algo muy similar pasa con los sinónimos, aunque sean escritos correctamente, el chatbot nunca será capaz de encontrar lo que el usuario está buscando.

4.4 Pruebas de respuesta

Como se mencionó anteriormente, se desarrollaron dos chatbots, uno con extractor de entidades con Python, y el otro con extractor de entidades con RASA. Para comparar su desempeño se realizaron 65 preguntas y las respuestas fueron divididas en cinco categorías que se describen a continuación.

- Ninguna respuesta: Sucede cuando el chatbot no responde con ningún mensaje al usuario, por lo que se necesita hacer una pregunta diferente para que el chatbot pueda volver a su funcionamiento normal, pero jamás se recibe la respuesta sobre la pregunta anterior
- Respuesta correcta: El usuario obtiene la respuesta a la pregunta realizada
- No en BD: La respuesta no se encuentra en la base de datos
- Respuesta incorrecta: Se obtiene una respuesta incorrecta a la pregunta del usuario
- No entendió: El chatbot no fue capaz de entender la petición del usuario y le indica que debe reformular la pregunta

En la Figura 32 se muestra los resultados del chatbot con un porcentaje del 75% de respuestas correctas. Con estos resultados se llegó a la conclusión de que el chatbot

era capaz de entender la intención del usuario, pero no lograba entender las entidades, por lo que no era capaz de dar una respuesta. Por esta razón, al recibir el mensaje del usuario se procedía a identificar las entidades de una manera externa a RASA ya que en algunas veces RASA identificaba las preposiciones (en, la, el, con, etc.) como una entidad. Los resultados experimentales del extractor de entidades de RASA se muestran en la Tabla 6 Anexo 2.

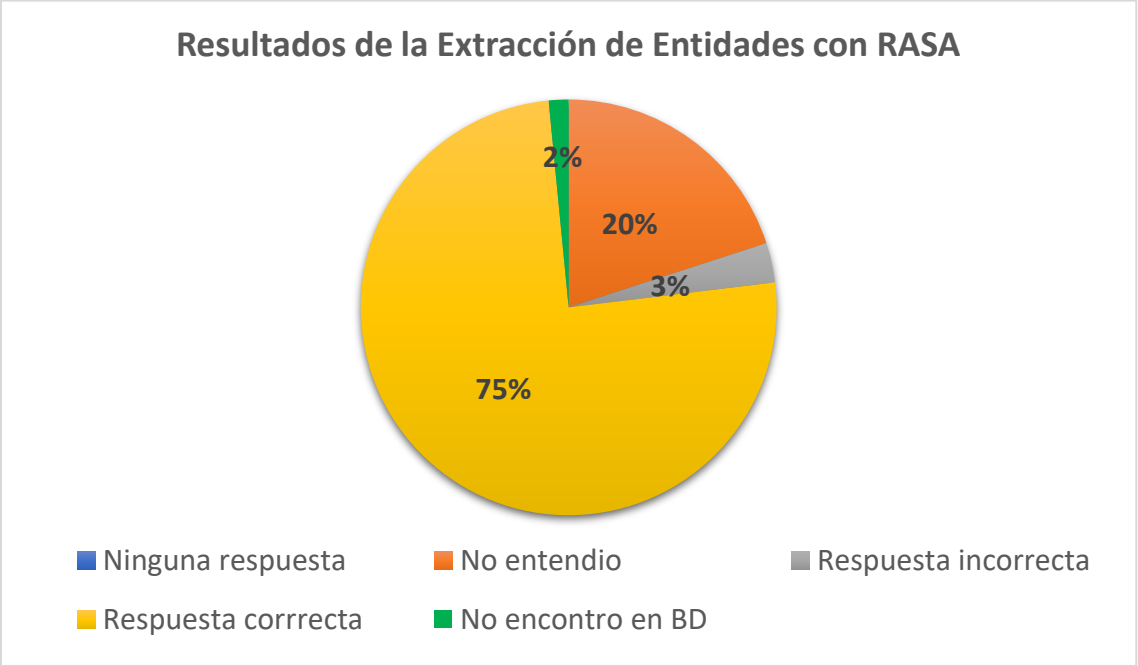

Figura 32. Resultados de calidad de respuesta con extracción de entidades con RASA.

Con la identificación de entidades realizada de manera externa se obtuvieron los resultados mostrados en la Figura 33. Realizando la extracción de entidades por medio del código de Python se obtuvieron mejores resultados que con la extracción con RASA, logrando hasta un 95% de efectividad en las respuestas y sólo un 5% con resultados no encontrados en la base de datos. Las preguntas realizadas fueron las mismas que con el chatbot anterior para lograr una comparación apropiada. Los resultados obtenidos en las pruebas experimentales con el extractor de entidades con Python se muestran en la Tabla 7 Anexo 2.

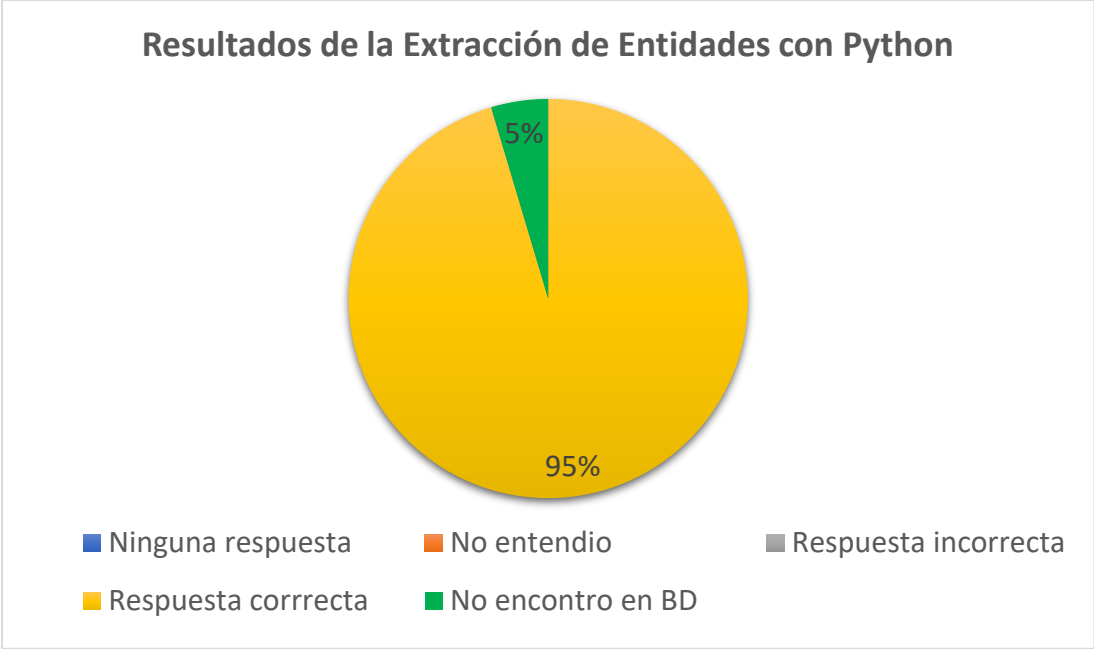

Figura 33. Resultados de respuestas con la extracción de entidades con Python.

4.5 Funcionamiento general

RASA cuenta con comandos que son capaces de evaluar el chatbot, para esto se necesitan algunas historias de ejemplos para poder hacer la autoevaluación. En la Figura 34 se muestran dos ejemplos de estas historias para realizar la autoevaluación. En cada una de ellas se debe de indicar un mensaje de ejemplo que podría introducir un usuario, en este mensaje se deben de indicar las entidades entre corchetes "[]" e indicar el tipo entre paréntesis "(info)". Posteriormente se debe de indicar el tipo de intención del mensaje y todos los pasos (acciones) que debe de seguir el chatbot para dar una respuesta.

```
stories:
- story: dar info licenciaturas
  steps:
  - user: |
      dame informacion sobre las [licenciaturas](info)
    intent: dar_info_general
  - action: action_dar_info_general
  - action: utter_necesitas_algo_mas

- story: dar info calendario
  steps:
  - user: |
      dame informacion sobre el [calendario](info)
    intent: dar_info_general
  - action: action_dar_info_general
  - action: utter_necesitas_algo_mas
```

Figura 34. Historias para la autoevaluación.

Al realizar la autoevaluación RASA emulará esa interacción entre el usuario y el chatbot de acuerdo con los ejemplos dados y se obtendrán distintas matrices de confusión de la clasificación de entidades, intenciones y acciones dependiendo su funcionamiento y arrojará los errores que hubo en las interacciones en caso de haberse presentado. En este caso no se presentó ningún problema.

En la Tabla 8 Anexo 2 se muestran la lista de intenciones y precisión de cada una de ellas. Estos datos fueron obtenidos con una herramienta de RASA.

En la Figura 35 se muestra la matriz de confusión de entidades extraídas con RASA. Se puede ver que el número de entidades de todos los ejemplos son predichas de manera correcta. De acuerdo con la matriz se presenta una precisión de 1 para cada una de las clases.

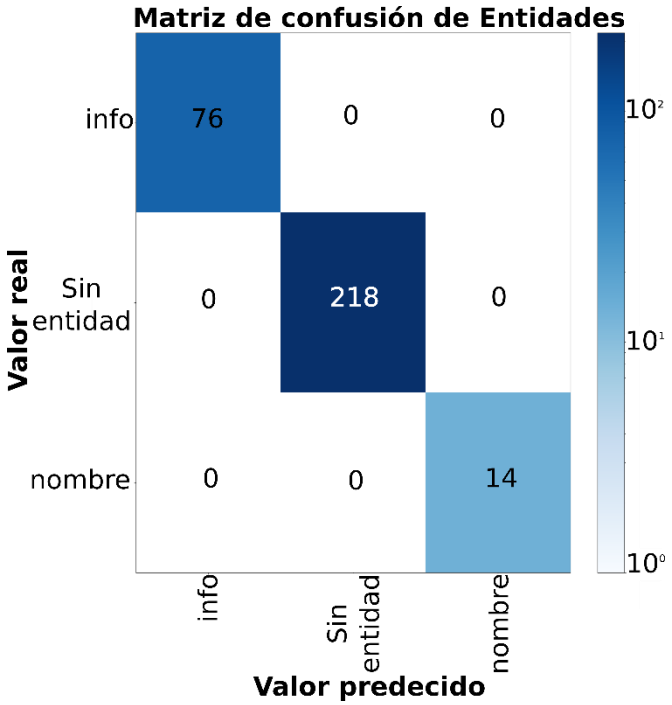

Figura 35. Matriz de confusión de entidades.

En la Figura 36 se muestra la matriz de confusión de intenciones extraídas. Se puede ver que el número de intenciones de todos los ejemplos son predichas de manera correcta. De acuerdo con la matriz se presenta una precisión de 1 para cada una de las clases.

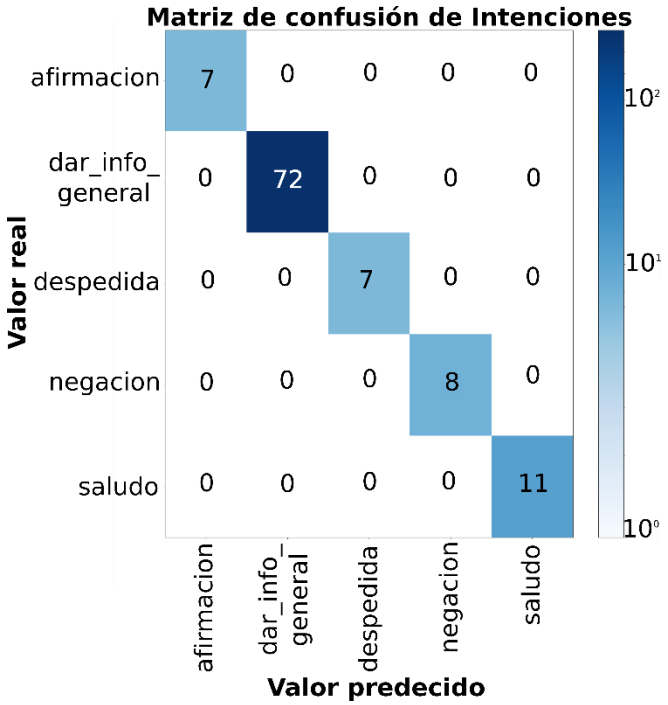

Figura 36 Matriz de confusión de intenciones.

En la Figura 37 se muestra la matriz de confusión de las acciones predichas con RASA. Se puede ver que el número de acciones de todos los ejemplos son predichas de manera correcta. De acuerdo con la matriz se presenta una precisión de 1 para cada una de las clases.

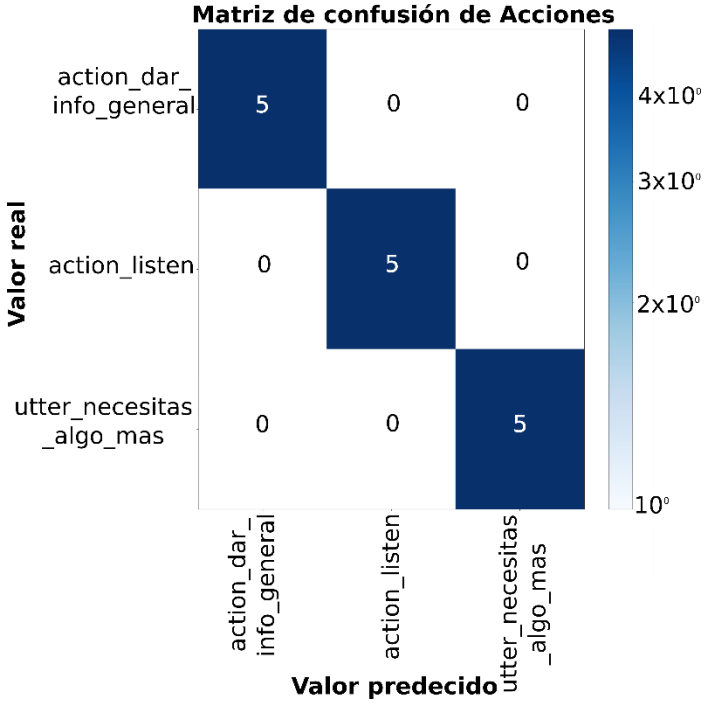

Figura 37. Matriz de confusión de acciones

Una vez verificado el funcionamiento del chatbot y obteniendo buenos resultados respecto al porcentaje de efectividad de las respuestas se procedieron a realizar algunas pruebas de campo, las cuales se describen a continuación.

4.6 Pruebas de campo

Se eligieron distintos grupos de alumnos de la UAM Iztapalapa para evaluar el funcionamiento de ambos chatbots, extractor de entidades vía RASA y extractor de entidades por código Python.

Las preguntas para ambos chatbots fueron similares, de esta manera se puede saber cuál de los dos será capaz de entender de mejor manera las peticiones de los usuarios y/o ser capaz de responder de manera correcta. Para estas pruebas solo se dividieron los datos en tres categorías para facilitar la evaluación por parte de los usuarios. Las categorías se describen a continuación.

- Se obtuvo respuesta: Cuando el chatbot fue capaz de responder a la solicitud del usuario
- No se encontró nada en BD: Cuando el chatbot entendió la entidad del mensaje del usuario, pero no encontró nada relacionado en la base de datos
- No se obtuvo respuesta: Cuando el chatbot no fue capaz de reconocer la entidad o la intención del mensaje del usuario y por esa razón no se obtuvo ninguna respuesta

De acuerdo con lo anterior se puede observar en la Figura 38 que hubo un porcentaje del 14% de preguntas que no fueron contestadas y un 61% de preguntas que fueron contestadas. Los resultados de las pruebas de campo con el chatbot con extractor de entidades con RASA se muestran en la Tabla 9 Anexo 2.

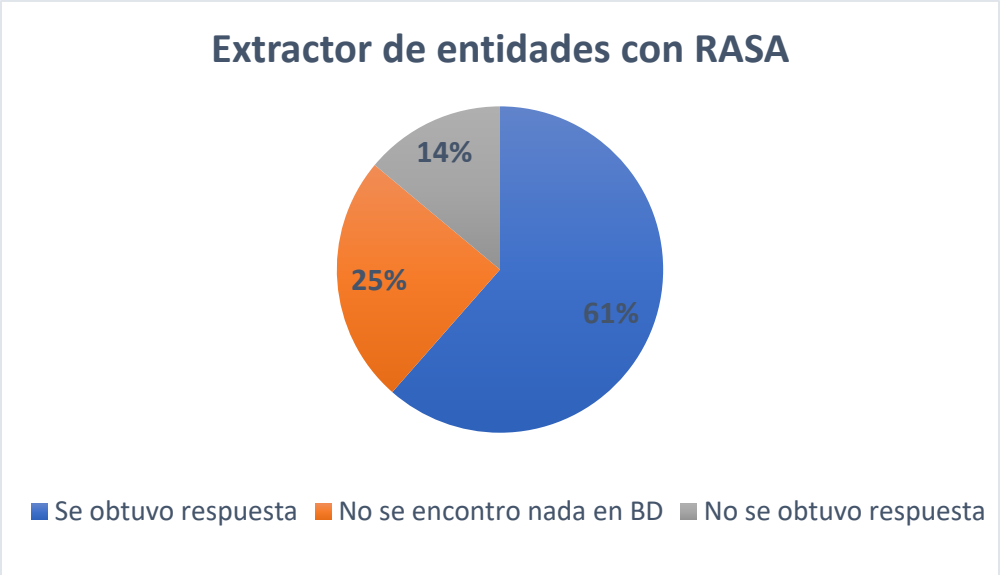

Figura 38. Resultados de las pruebas de campo del chatbot con extractor de entidades con RASA.

En la Figura 39 se muestra que el 73.3% de las personas que interactuaron con el chatbot consideran que las respuestas dadas por el mismo satisfice a sus necesidades.

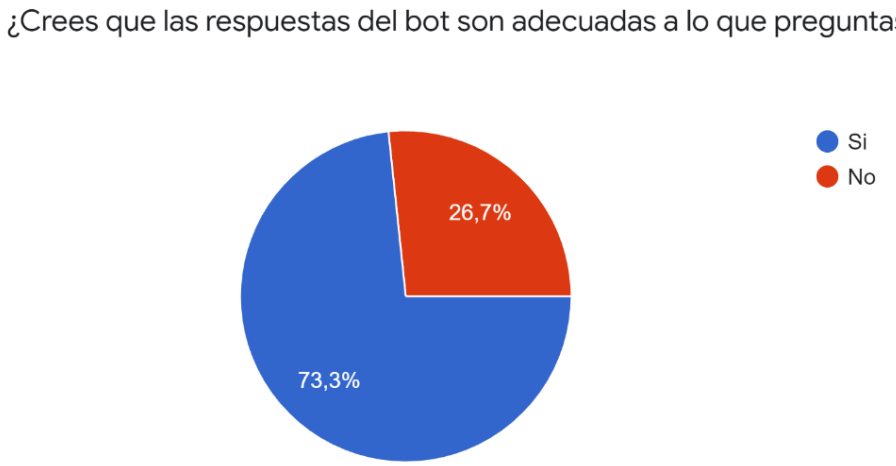

Figura 39. Porcentaje de respuestas efectivas del chatbot.

En la Figura 40 se muestran los resultados del chatbot con extractor de entidades con Python. La gráfica indica que disminuyó el 10% de preguntas que no se entendieron y aumentó el porcentaje al 46% de respuestas no encontradas en la base de datos. A su vez disminuyo el porcentaje de respuestas correctas. Los resultados de las pruebas de campo obtenidas con el chatbot con extractor de entidades con Python se muestran en la Tabla 10 Anexo 2.

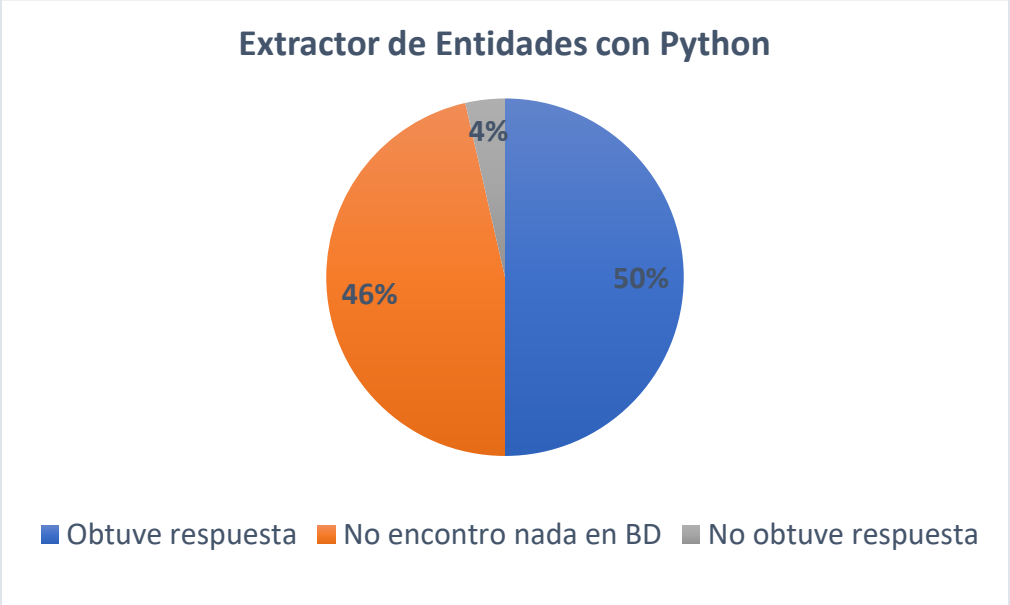

Figura 40. Resultados de las pruebas de campo del chatbot con extractor de entidades con Python.

En la Figura 41 se muestra que el 88.9% de las personas que interactuaron con el chatbot consideran que las respuestas dadas por el mismo satisfice a sus necesidades.

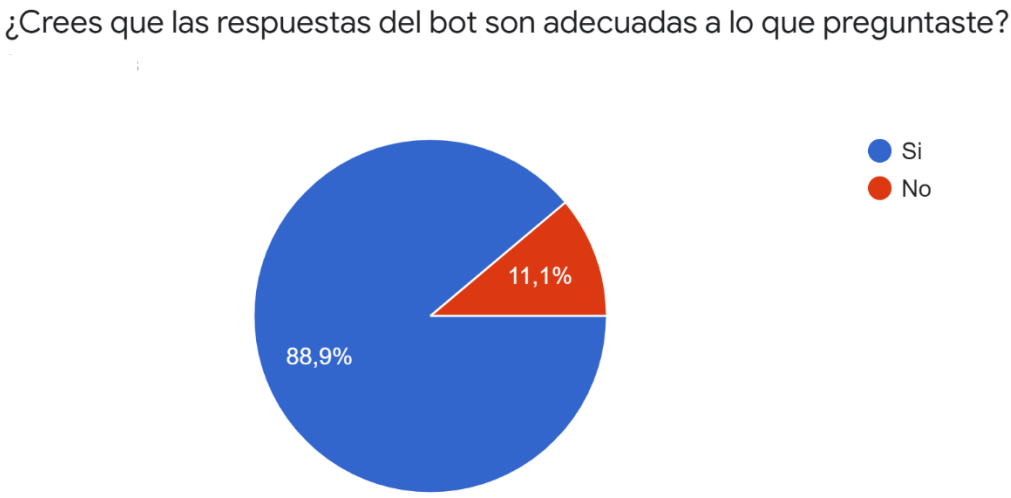

Figura 41. Porcentaje de respuestas efectivas del chatbot.

Se puede observar que en ambas pruebas se obtuvieron resultados muy distintos a los obtenidos en la Figura 32 y Figura 33, ya que en esas pruebas se les hacían preguntas sobre información que se sabía que estaba en la base de datos y en las pruebas de campo algunos alumnos preguntaban sobre información que podría ser externa a la UAM Iztapalapa, o información que no podía encontrarse en las páginas de la universidad.

Si se analizan las gráficas obtenidas anteriormente (Figura 32 y Figura 33) con las obtenidas en las pruebas de campo (Figura 38 y Figura 40), los resultados continúan siendo similares ya que el extractor de entidades creado con Python hace que la mayor parte de las entidades se puedan entender a comparación con el extractor de entidades con RASA.

Finalmente, se les preguntó a los usuarios cuál chatbot les dio respuestas más efectivas. Como se muestra en la Figura 42 el 62% de los usuarios prefirieron el chatbot con extractor de entidades de RASA ya que es el que pudo contestar la mayoría de las preguntas.

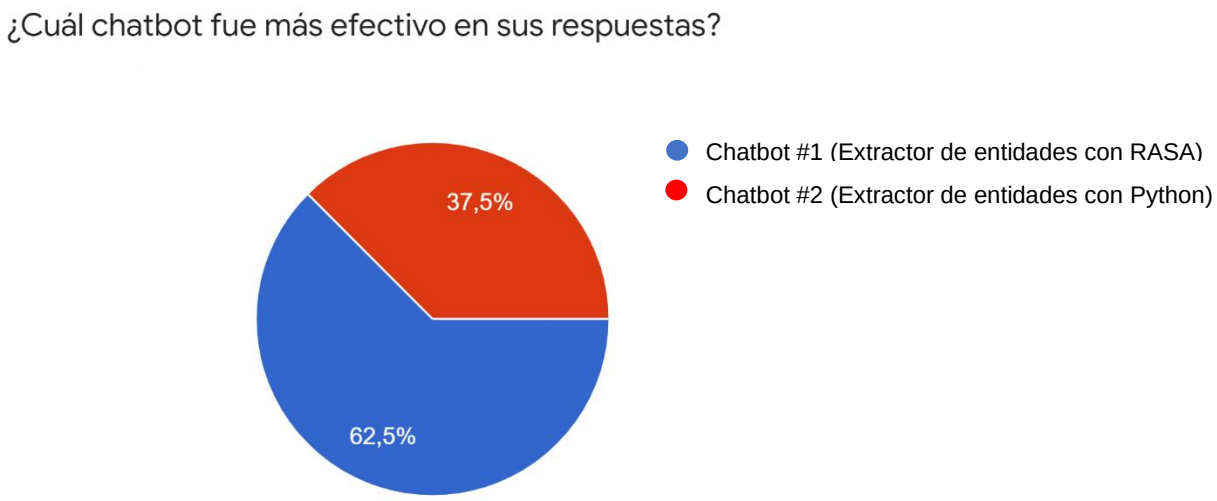

Figura 42. Comparación de efectividad de ambos chatbots.

Capítulo 5

5. Conclusiones

En este trabajo se realizó una arquitectura de chatbot, cuyo objetivo es implementar los componentes más importantes encontrados en la literatura. Esta arquitectura es importante por los beneficios que se pueden presentar para los usuarios por las respuestas actualizadas constantemente y para los desarrolladores por el ahorro de trabajo de codificación para las bases de datos.

El objetivo general de esta investigación fue diseñar e implementar una arquitectura de chatbot con elementos de búsqueda externa, obteniendo buenos resultados en las respuestas dadas.

Se probaron diferentes técnicas de extracción web para poder determinar la más rápida y la que sea capaz de extraer la mayor cantidad de datos para ser implementada en la arquitectura. Esto permite la consulta de información extraída de la web y ser usada para la generación de respuesta sin necesidad de dar demasiados datos de entrenamiento y respuestas predefinidas. Lo antes mencionado presenta una gran ventaja tanto para los usuarios al contar con respuestas actualizadas de la web, como para los desarrolladores al requerir menor trabajo para llenar y/o actualizar las bases de datos.

Aunque la búsqueda del chatbot ha arrojado un 95% de palabras encontradas, aún cuenta con algunas limitaciones, como el no poder dar una respuesta con una palabra mal escrita o con un sinónimo, por lo que se continuará el desarrollo y mejora de la respuesta o interpretación de la pregunta para mejorar el desempeño del chatbot.

Este trabajo comparó dos técnicas de extracción de entidades para mejorar la calidad de las respuestas. Se efectuaron pruebas experimentales en donde se le realizaban preguntas que se sabía que sí eran capaces de responder y que las respuestas se encontraban en las bases de datos. También se realizaron pruebas de campo con diferentes grupos de alumnos para tener distintas perspectivas de la manera de realizar preguntas, para aumentar la cantidad de datos y volver a entrenar los chatbots, así como para tener una mejor comparación de ambas técnicas de extracción de entidades.

Los resultados experimentales muestran que es mejor utilizar el extractor de entidades con Python ya que puede mejorar hasta un 95% de respuestas obtenidas. Las pruebas de campo muestran que es mejor realizar un extractor de entidades externo a usar el clasificador de entidades de RASA, ya que disminuyó el porcentaje hasta un 10% en las preguntas no contestadas. Aunque el porcentaje de respuestas satisfactorias con el extractor de entidades de Python (50%) es menor con el extractor de entidades de RASA (61%) es posible mejorarlo aumentando la cantidad de palabras que no aportan

en el mensaje del usuario para ser eliminadas a diferencia de RASA que, aumentando los ejemplos de entidades no es posible mejorar la efectividad de las respuestas.

El chatbot es capaz de ayudar a los alumnos de la UAM-I a obtener información de cualquier tipo de información relacionada con la Universidad, pero el proyecto tiene la ventaja de ser escalable y poder ser utilizado para múltiples áreas dependiendo de la información que se encuentre en las páginas web para alimentar la base de datos interna.

Trabajo futuro propuesto: probar diferentes técnicas de búsqueda para la base de datos, implementar los elementos faltantes de la arquitectura ideal propuesta, utilizar las preguntas obtenidas en las pruebas de campo para reentrenar los chatbots esperando que se obtengan mejores resultados.

Referencias

[1] B. Gmage, R. Pushpananda, and R. Weerasinghe, “The impact of using pre-trained word embeddings in Sinhala chatbots,” *20th Int. Conf. Adv. ICT Emerg. Reg.*, pp. 161–165, 2020, doi: 10.1109/ICTer51097.2020.9325440.

[2] E. Adamopoulou and L. Moussiades, “Chatbots: History, technology, and applications,” *Mach. Learn. with Appl.*, vol. 2, pp. 1–18, 2020, doi: https://doi.org/10.1016/j.mlwa.2020.100006.

[3] J. Weizenbaum, “ELIZA—a computer program for the study of natural language communication between man and machine,” *Commun. ACM*, 1966, doi: https://dl.acm.org/doi/10.1145/365153.365168.

[4] K. M. Colby, S. Weber, and F. D. Hilf, “Artificial Paranoia,” *Artif. Intell.*, vol. 2, no. 1, pp. 1–25, 1971, doi: https://doi.org/10.1016/0004-3702(71)90002-6.

[5] J. Jiyou, “CSIEC: A computer assisted English learning chatbot based on textual knowledge and reasoning,” *Knowledge-Based Syst.*, vol. 22, pp. 249–255, 2009, doi: https://doi.org/10.1016/j.knosys.2008.09.001.

[6] R. S. Wallace, “The Anatomy of A.L.I.C.E.,” in *Parsing the Turing Test*, 2009, pp. 181–2010.

[7] M. S. Satu, M. H. Parvez, and Shamim-Al-Mamun, “Review of integrated applications with AIML based chatbot,” in *2015 International Conference on Computer and Information Engineering (ICCIE)*, Nov. 2015, pp. 87–90, doi: 10.1109/CCIE.2015.7399324.

[8] G. Molnar and S. Zoltán, “The Role of Chatbots in Formal Education,” in *2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY)*, 2018, pp. 197–202, [Online]. Available: https://doi.org/10.1109/SISY.2018.8524609.

[9] M. Haenlein and A. Kaplan, “A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence,” *Calif. Manage. Rev.*, pp. 1–10, 2019, doi: http://dx.doi.org/10.1177/0008125619864925.

[10] L. Rouhiaine, *Inteligencia Artificial 101 cosas que debes saber hoy sobre nuestro futuro.*, 1st ed. 2018.

[11] Oracle, “¿Qué es la inteligencia artificial-IA?,” 2021.

[12] G. Chowdhury, “Natural language processing,” in *Information Science and Technology*, asis&t, 2003, pp. 51–89.

[13] M. Lombardi, F. Pascale, and D. Santaniello, “An application for Cultural Heritage using a Chatbot,” *2nd Int. Conf. Comput. Appl. Inf. Secur.*, 2020, doi: https://doi.org/10.1109/CAIS.2019.8769525.

[14] S. Santhana Lakshmi, “A Study on Machine Learning based Conversational Agents

and Desisning Techniques," *Proc. Fourth Int. Conf. I- SMAC (IoT Soc. Mobile, Anal. Cloud)*, pp. 965–968, 2020, doi: http://dx.doi.org/10.1109/I-SMAC49090.2020.9243577.

[15] P. Smutny and P. Schreiberova, "Chatbots for learning: A review of educational chatbots for the Facebook Messenger," *Comput. Educ.*, vol. 151, pp. 1–11, 2020, doi: https://doi.org/10.1016/j.compedu.2020.103862.

[16] HelpShift, "¿Qué son las entidades e intenciones?," *¿Qué son las entidades e intenciones?*, 2020. https://www.helpshift.com/glossary/intent-in-chatbot/.

[17] S. Abhishek, K. Ramasubramanian, and S. Shivam, "Introduction to RASA," in *Building an Enterprise Chatbot*, 2019, pp. 292–295.

[18] Octoparse, "Servicios De Web Scraping: Cómo Comenzó y Qué Sucederá en El Futuro," 2020. https://www.octoparse.es/blog/como-comenzo-y-sucedera-en-futuro#.

[19] V. N. Gudivada, V. V. Raghavan, W. I. Grosky, and R. Kasanagottu, "Information retrieval on the World Wide Web," *IEEE Internet Comput.*, vol. 1, no. 5, pp. 58–68, 1997, doi: 10.1109/4236.623969.

[20] M. I. Mauldin, "Lycos: design choices in an Internet search service," *IEEE Expert*, vol. 12, no. 1, pp. 8–11, Jan. 1997, doi: 10.1109/64.577466.

[21] I. Hernández, C. R. Rivero, and D. Ruiz, "Deep Web crawling: a survey," *World Wide Web*, pp. 1577–1610, 2018, doi: https://doi.org/10.1007/s11280-018-0602-1.

[22] D. Glez-Peña, L. Anália, H. López-Fernandez, M. Reboiro-Jato, and F. Fdez-Riverola, "Web scraping technologies in an API world," *Brief. Bioinform.*, vol. 15, pp. 788–797, 2013, doi: 10.1093/bib/bbt026.

[23] V. Smith, "Why do you need a web scraper?," in *Go Web Wcraping Quick Start Guide*, Packt Publishing Ltd., 2019, pp. 6–7.

[24] Goutte, "Goutte, a simple PHP Web Scraper," 2021. https://goutte.readthedocs.io/en/latest/.

[25] M. Dowling, "Guzzle Documentation," 2015. https://docs.guzzlephp.org/en/stable/.

[26] "Beautiful Soup Documentation," 2020. https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[27] Scrapy developers, "Scrapy 2.5 documentation," 2021. https://docs.scrapy.org/en/latest/.

[28] Selenium, "The Selenium Browser Automation Project," 2021. https://www.selenium.dev/documentation/.

[29] S. Abhishek, K. Ramasubramanian, and S. Shivam, "Introduction to Chatbots Architecture," in *Building an Enterprise Chatbot*, 2020, pp. 73–75.

[30] S. Hussain and A. Ginige, "Extending a conventional chatbot knoledge base to

external knoledge source and introducing user based sessions for diabetes education," *32nd Int. Conf. Adv. Inf. Netw. Appl. Work.*, pp. 698–703, 2018, doi: 10.1109/WAINA.2018.00170.

[31] M. Manilal, B. R V, S. A J, A. A. Mathew, and B. Babu, "A Graph Based Chatbot for Cancer Patients," *2019 5th Int. Conf. Adv. Comput. Commun. Syst.*, pp. 717–721, 2019.

[32] H. Collins and S. Alam, "Using Web Harvested Semi-Structured Data to Build an Inspirational Chatbot," *ResearchGate*, pp. 1–10, 2019.

[33] R. Bathija, P. Agarwal, and R. Somanna, "Guided Interactive Learning through Chatbot using Bi-directional Encoder Representations from Transformers (BERT)," *Proc. Second Int. Conf. Innov. Mech. Ind. Appl.*, pp. 82–87, 2020, doi: https://doi.org/10.1109/ICIMIA48430.2020.9074905.

[34] C. Fabio, M. Lombardi, F. Colace, and F. P, "Chatbot: An Education Support System for Student," *10th Int. Symp. CSS*, pp. 291–302, 2018, doi: 10.1007/978-3-030-01689-0_23.

[35] C. Segura, Á. Palau, J. Luque, M. Costa-Jussá, and R. E. Banchs, "Chatbol, a Chatbot for the Spanish 'La Liga,'" *9th Int. Work. Spok. Dialogue Syst. Technol. Lect. Notes Electr. Eng.*, pp. 319–330, 2019.

[36] A. Aviral Nigam, "Web Crawling Algorithms," *Int. J. Comput. Sci. Artif. Intell.*, vol. 4, no. 3, pp. 63–67, Sep. 2014, doi: 10.5963/IJCSAI0403001.

[37] R. Baeza-Yates, C. Castillo, M. Marin, and A. Rodriguez, "Crawling a country," in *Special interest tracks and posters of the 14th international conference on World Wide Web - WWW '05*, 2005, p. 864, doi: 10.1145/1062745.1062768.

[38] scalable minds, "React-based Chatroom Component for Rasa Stack." 2019, [Online]. Available: https://www.npmjs.com/package/@scalableminds/chatroom.

[39] C. Musciano and B. Kennedy, *HTML & XHTML: The Definitive Guide: The Definitive Guide*, 5th ed. O'REILLY, 2002.

Anexo 1: Pseudocódigos

Pseudocódigo para la extracción de información de la web.

Variables

Por_Visitar: Pila de enlaces por visitar, se agrega por default a la página izt.uam.mx y cseuami.org
No_Visitados: Lista de enlaces que no pudieron ser abiertos con ningún método
Visitados: Enlaces visitados

```
1. Mientras por_visitar no este vacia
2.      Sacar de por_visitar al último elemento añadido
3.      SI enlace se puede abrir correctamente y enlace no está en visitados: // abrir con Scrapy o Beautiful Soup
4.              Agregar enlace a visitados
5.              Extraer enlaces/hijos
6.      Si hijos no están en visitados e hijos no están en por_visitar:
7.                      Agregar hijos a por_visitar
8.      SI NO
9.              Enlace_nuevo = enlace + 'wordpress'
10.             SI enlace_nuevo se puede abrir correctamente y enlace no está en visitados: // abrir con S o BS
11.                     Agregar enlace_nuevo a visitados
12.                     Extraer enlaces/hijos
13.                     Si hijos no están en visitados e hijos no están en por_visitar:
14.                             Agregar hijos a por_visitar
15.             SI NO
16.                     Enlace_nuevo = enlace + 'wp'
17.                     SI enlace_nuevo se puede abrir correctamente y enlace no está en visitados: // abrir con S o BS
18.                             Agregar enlace_nuevo a visitados
19.                             Extraer enlaces/hijos
20.                             Si hijos no están en visitados e hijos no están en por_visitar:
21.                                     Agregar hijos a por_visitar
22.                     SI NO
23.                             SI enlace se puede abrir correctamente y enlace no está en visitados //abrir con Selenium
24.                                     Agregar enlace a visitados
25.                                     Extraer enlaces/hijos
26.                                     Si hijos no están en visitados e hijos no están en por_visitar:
27.                                             Agregar hijos a por_visitar
28.                             SI NO
29.                             Agregar a no_visitados
```

Pseudocódigo de la extracción de información con código Python

1. Recibir mensaje del usuario
2. Eliminar stopwords y palabras que no sean consideradas como entidades
3. Realizar la búsqueda de entidades en la BD
4. SI se encontró en la BD:
5. Presentar resultados al usuario
6. SI NO:
7. Indicar que no se encontró en BD

Anexo 2: Tablas de Resultados

Resultados de los tiempos de búsqueda de la palabra “Areli” con diferentes técnicas sin la descarga de información.

Scrapy (seg)	Scrapy & Beautiful Soup (seg)	Beautiful Soup (seg)	Beautiful Soup con hilos (seg)
22.53	22.18	50.47	23.4
21.57	22.9	55.05	24.44
21.32	22.99	57.65	22.86
21.77	23.54	61.01	25.21
21.47	21.63	57.62	22.23
22.08	21.6	57.62	22.46
21.51	23.03	60.07	22.75
20.97	22.63	59.55	22.4
21.55	22.02	55.43	23.11
21.94	22.71	59.48	22.63
22.02	21.72	59.15	23.19
21.54	22.45	59.71	22.99
21.61	22.96	52.74	22.79
21.39	23.05	54.15	23.1
22.18	21.07	50.48	23.12
21.56	22.44	51.14	22.22
21.58	22.9	52.43	22.22
21.38	21.86	50.62	23.84
21.53	22	50.17	22.64
22.01	22.72	51.18	22.48
21.66	21.66	50.8	22.96
21.72	22.48	50.65	22.84
22.06	21.9	52.05	23
21.98	22.44	51.15	22.14
21.63	21.62	50.06	22.87
21.02	21.92	50.58	22.28
21.3	21.43	51.58	22.42
21.57	21.42	51.1	22.74
22.31	22.97	51.09	22.71
21.52	21.86	55.53	22.59

21.676	22.27	54.01033333	22.88766667	**Promedio**

Tabla 3. Resultados de los tiempos de búsqueda.

Resultados de los tiempos de búsqueda de la palabra “Areli” con diferentes técnicas con la descarga de información en archivo CSV.

Scrapy (seg)	Sscrapy & Beautiful Soup (seg)	Beautiful Soup (seg)	Beautiful Soup con hilos (seg)	
22.48	29.39	54.58	23.71	
22.05	22.21	52.14	24.36	
21.87	28.2	57.57	23.55	
21.98	28.96	51.5	25.35	
21.74	28.97	54.32	23.41	
21.92	29.18	52.87	23.24	
21.72	27.89	51.77	23.74	
21.75	27.05	51.45	23	
21.92	26.49	52.96	23.02	
21.96	26.92	51.63	23.04	
21.99	26.62	50.59	22.94	
22.23	26	56.57	22.69	
21.92	25.98	53.24	23.99	
22.25	28.64	52.46	22.98	
21.67	27.31	51.55	25.3	
21.76	26.5	51.71	23.68	
21.75	28.36	52.43	24.06	
21.83	28.45	51.3	24.01	
22.74	27.83	51.05	22.56	
22.09	27.06	51.79	23.13	
21.96	26.97	58.6	22.98	
21.85	26.43	55.19	24.21	
21.68	28.55	52.42	27.6	
22.63	28.32	53.02	23.17	
22.24	28.33	53.34	22.86	
22.26	28.65	54.28	23.19	
22.97	27.05	51.18	23.7	
21.9	26.07	50.78	24.15	
21.96	27.86	51.63	23.47	
21.94	26.85	52.9	23.66	
22.03366667	27.4363333	52.894	23.6916667	**Promedio**

Tabla 4. Resultados de búsqueda con distintas técnicas de extracción.

Tabla de número de enlaces extraídos por día.

Fecha	Tiempo (hrs)	Enlaces extraídos	Enlaces no visitados
25/11/2021	2.25	5853	14
26/11/2021	2	5854	14
27/11/2021	2.17	5542	22
28/11/2021	2.08	5601	17
29/11/2021	2.14	5626	17
30/11/2021	2.08	5890	14
01/12/2021	2.07	5900	14
02/12/2021	2.08	5798	15
03/12/2021	2.04	5836	14
06/12/2021	2	5651	14
07/12/2021	2.08	5659	14
08/12/2021	1.97	5681	14
09/12/2021	1.92	5593	14
12/12/2021	1.93	5628	14
13/12/2021	2.14	5682	16
14/12/2021	2.06	5693	14
15/12/2021	2.07	5684	14
16/12/2021	1.96	5695	14
18/12/2021	2	5707	14
19/12/2021	2.03	5707	14
20/12/2021	2.03	5694	14
21/12/2021	2.07	5704	14
22/12/2021	2.06	5704	14
23/12/2021	2.08	5704	14
24/12/2021	2.14	5704	14
27/12/2021	2.06	5707	14
30/12/2021	2.06	5710	14
31/12/2021	2.08	5710	14
03/01/2022	2.1	5717	14
05/01/2022	1.94	5719	14

Tabla 5. Enlaces extraídos por día.

Resultados experimentales del chatbot con extractor de entidades de RASA.

Entidad	Ninguna respuesta	No entendió	Respuesta incorrecta	Respuesta correcta	No encontró en BD
aula digital	0	0	0	1	0
sobre danza	0	0	0	1	0
filosofia	0	0	0	1	0
calculo	0	0	0	1	0
optativa	0	0	0	1	0
examen	0	0	0	1	0
Julio	0	0	0	1	0
Informe	0	0	0	1	0
admision	0	1	0	0	0
eventos	0	1	0	0	0
colegiaturas	0	0	0	1	0
protocolo	0	1	0	0	0
tesis	0	0	0	1	0
redes de comunicaciones	0	0	0	1	0
departamento de sociologia	0	0	0	1	0
biblioteca	0	0	0	1	0
correo	0	0	0	1	0
Recomendaciones para alumnos	0	0	0	0	1
marco regulatorio	0	0	0	1	0
Omar villa	0	0	0	1	0
reuniones	0	1	0	0	0
matematicas	0	0	0	1	0
plan de estudios	0	0	0	1	0
transformaciones quimicas	0	0	0	1	0
coordinaciones	0	0	0	1	0
pandemia	0	0	0	1	0
dictamen 2021	0	0	0	1	0
libros	0	0	0	1	0
informacion general	0	1	0	0	0
unidad iztapalapa	0	0	0	1	0
guia bibliografica	0	0	0	1	0
otras becas	0	0	0	1	0
costos	0	1	0	0	0

procesos administrativos	0	0	0	1	0
planta academica	0	0	0	1	0
historia	0	1	0	0	0
quimica	0	0	0	1	0
publicaciones	0	0	0	1	0
monto de cuotas	0	0	0	1	0
fisica de liquidos	0	0	0	1	0
posgrado	0	0	0	0	0
ingenieria electronica	0	0	0	1	0
alumnos	0	0	0	1	0
rector	0	1	0	0	0
clases	0	1	0	0	0
avisos	0	1	0	0	0
calendario escolar	0	0	0	1	0
lineas de investigacion	0	0	0	1	0
david islas	0	0	0	1	0
votaciones	0	0	0	1	0
celex	0	0	0	1	0
cbi	0	0	0	1	0
csh	0	0	0	1	0
posgrados	0	1	0	0	0
becas	0	0	0	1	0
areli	0	0	0	1	0
enrique	0	0	0	1	0
movilidad	0	1	0	0	0
sistemas escolares	0	0	0	1	0
licenciatura en computacion	0	0	0	1	0
biomedica	0	0	0	1	0
pago del trimestre	0	0	0	1	0
reincripcion	0	0	0	1	0
renuncia de uea	0	1	0	0	0
quita oportunidad	0	1	0	0	0
Total	0	15	0	49	1

Tabla 6. Resultados experimentales del chatbot con extractor de entidades de RASA.

Resultados experimentales del chatbot con extractor de entidades con código Python.

Entidad	Ninguna respuesta	No entendió	Respuesta incorrecta	Respuesta correcta	No encontró en BD
aula digital	0	0	0	1	0
sobre danza	0	0	0	1	0
filosofia	0	0	0	1	0
calculo	0	0	0	1	0
optativa	0	0	0	1	0
examen	0	0	0	1	0
Julio	0	0	0	1	0
Informe	0	0	0	1	0
admision	0	0	0	1	0
eventos	0	0	0	1	0
colegiaturas	0	0	0	1	0
protocolo	0	0	0	1	0
tesis	0	0	0	1	0
redes de comunicaciones	0	0	0	1	0
departamento de sociologia	0	0	0	1	0
biblioteca	0	0	0	1	0
correo	0	0	0	1	0
Recomendaciones para alumnos	0	0	0	0	1
marco regulatorio	0	0	0	1	0
Omar villa	0	0	0	1	0
reuniones	0	0	0	1	0
matematicas	0	0	0	1	0
plan de estudios	0	0	0	1	0
transformaciones quimicas	0	0	0	1	0
coordinaciones	0	0	0	1	0
pandemia	0	0	0	1	0
dictamen 2021	0	0	0	1	0
libros	0	0	0	1	0
informacion general	0	0	0	1	0
unidad iztapalapa	0	0	0	1	0
guia bibliografica	0	0	0	1	0

otras becas	0	0	0	1	0
costos	0	0	0	1	0
procesos administrativos	0	0	0	1	0
planta academica	0	0	0	1	0
historia	0	0	0	1	0
quimica	0	0	0	1	0
publicaciones	0	0	0	1	0
monto de cuotas	0	0	0	1	0
fisica de liquidos	0	0	0	1	0
posgrado	0	0	0	1	0
ingenieria electronica	0	0	0	1	0
alumnos	0	0	0	1	0
rector	0	0	0	1	0
clases	0	0	0	1	0
avisos	0	0	0	1	0
calendario escolar	0	0	0	1	0
lineas de investigacion	0	0	0	1	0
david islas	0	0	0	1	0
votaciones	0	0	0	1	0
celex	0	0	0	1	0
cbi	0	0	0	1	0
csh	0	0	0	1	0
posgrados	0	0	0	1	0
becas	0	0	0	1	0
areli	0	0	0	1	0
enrique	0	0	0	1	0
movilidad	0	0	0	1	0
sistemas escolares	0	0	0	1	0
licenciatura en computacion	0	0	0	1	0
biomedica	0	0	0	1	0
pago del trimestre	0	0	0	0	1
reincripcion	0	0	0	1	0
renuncia de uea	0	0	0	1	0
quita oportunidad	0	0	0	0	1
Total	0	0	0	62	3

Tabla 7. Resultados experimentales del chatbot con extractor de entidades con Python.

Tabla de precisión de Intenciones, tabla realizada con comandos de RASA. Datos utilizados para el entrenamiento del chatbot.

Intención esperada	Texto	Intención clasificada	Precisión
afirmacion	si	afirmacion	100.00%
afirmacion	s	afirmacion	100.00%
afirmacion	claro	afirmacion	100.00%
afirmacion	SI	afirmacion	100.00%
afirmacion	correcto	afirmacion	100.00%
afirmacion	sip	afirmacion	100.00%
afirmacion	sipi	afirmacion	100.00%
dar_info_general	necesito informacion de los seminarios	dar_info_general	100.00%
dar_info_general	quiero saber sobre los seminarios	dar_info_general	100.00%
dar_info_general	información sobre Plan de estudios	dar_info_general	100.00%
dar_info_general	información sobre calendario	dar_info_general	100.00%
dar_info_general	información sobre celex	dar_info_general	100.00%
dar_info_general	quiero saber sobre el celex	dar_info_general	100.00%
dar_info_general	información sobre biblioteca	dar_info_general	100.00%
dar_info_general	quiero saber sobre la división cbi	dar_info_general	100.00%
dar_info_general	dame infor sobre CBI	dar_info_general	100.00%
dar_info_general	dame infor sobre dictamen	dar_info_general	100.00%
dar_info_general	dame infor sobre libros	dar_info_general	100.00%
dar_info_general	que es la unidad iztapalapa	dar_info_general	100.00%
dar_info_general	unidad iztapalapa	dar_info_general	100.00%
dar_info_general	informacion sobre el PEER	dar_info_general	100.00%
dar_info_general	información sobre laboratorios	dar_info_general	100.00%
dar_info_general	información sobre egresados	dar_info_general	100.00%
dar_info_general	información sobre alumnos	dar_info_general	100.00%
dar_info_general	información sobre publicaciones	dar_info_general	100.00%
dar_info_general	quien es Areli	dar_info_general	100.00%
dar_info_general	dame informacion sobre Eric Rincón	dar_info_general	100.00%
dar_info_general	Enrique	dar_info_general	100.00%
dar_info_general	Miguel	dar_info_general	100.00%
dar_info_general	david	dar_info_general	100.00%

dar_info_general	juan	dar_info_general	100.00%
dar_info_general	Miguel Angel	dar_info_general	100.00%
dar_info_general	quiero saber sobre los [avisos]	dar_info_general	100.00%
dar_info_general	dame info sobre las [clases] de la unidad	dar_info_general	100.00%
dar_info_general	profesora Graciela	dar_info_general	100.00%
dar_info_general	info sobre los sistemas escolares	dar_info_general	100.00%
dar_info_general	información sobre aula digital	dar_info_general	100.00%
dar_info_general	quiero saber que necesito para la [quinta oportunidad]	dar_info_general	100.00%
dar_info_general	información sobre filosofia	dar_info_general	100.00%
dar_info_general	profesor Alfonso	dar_info_general	100.00%
dar_info_general	Dr. Omar	dar_info_general	100.00%
dar_info_general	Dr. Eric	dar_info_general	100.00%
dar_info_general	Julio	dar_info_general	100.00%
dar_info_general	dame informacion sobre cbs	dar_info_general	100.00%
dar_info_general	dame informacion sobre correo	dar_info_general	100.00%
dar_info_general	dame informacion sobre correo institucional	dar_info_general	100.00%
dar_info_general	dame informacion sobre informe	dar_info_general	100.00%
dar_info_general	quiero saber sobre la division CBS	dar_info_general	100.00%
dar_info_general	información sobre constancias	dar_info_general	100.00%
dar_info_general	como hago mi baja?	dar_info_general	100.00%
dar_info_general	como se hace la baja definitiva?	dar_info_general	100.00%
dar_info_general	información sobre lic. en biomedica	dar_info_general	100.00%
dar_info_general	ingenieria en biomedica	dar_info_general	100.00%
dar_info_general	firma de titulo	dar_info_general	100.00%
dar_info_general	donde realizo la firma de titulo	dar_info_general	100.00%
dar_info_general	información sobre credencial	dar_info_general	100.00%
dar_info_general	información sobre CSH	dar_info_general	100.00%
dar_info_general	dame informacion sobre la division csh	dar_info_general	100.00%
dar_info_general	información sobre certificado total	dar_info_general	100.00%
dar_info_general	información sobre ingeniería	dar_info_general	100.00%
dar_info_general	información sobre física	dar_info_general	100.00%

dar_info_general	información sobre Antropología social	dar_info_general	100.00%
dar_info_general	información sobre Geografía humana	dar_info_general	100.00%
dar_info_general	donde se hacen las encuestas?	dar_info_general	100.00%
dar_info_general	¿Cuándo se realiza la inscripción?	dar_info_general	100.00%
dar_info_general	cuando se realiza la inscripción	dar_info_general	100.00%
dar_info_general	quiero una rectificación de califiación	dar_info_general	100.00%
dar_info_general	necesito rectificar mi calificación	dar_info_general	100.00%
dar_info_general	revisón de califiación	dar_info_general	100.00%
dar_info_general	revisión	dar_info_general	100.00%
dar_info_general	información del seguro	dar_info_general	100.00%
dar_info_general	informacion del IMss	dar_info_general	100.00%
dar_info_general	información sobre titulacion	dar_info_general	100.00%
dar_info_general	info titulación	dar_info_general	100.00%
dar_info_general	informacion sobre celex	dar_info_general	100.00%
dar_info_general	celex	dar_info_general	100.00%
dar_info_general	dame informacion sobre celex	dar_info_general	100.00%
dar_info_general	informacion danza	dar_info_general	100.00%
dar_info_general	necesito que me des informacion sobre las becas	dar_info_general	100.00%
despedida	bye	despedida	100.00%
despedida	adios	despedida	100.00%
despedida	hasta pronto	despedida	100.00%
despedida	nos vemos	despedida	100.00%
despedida	chau	despedida	100.00%
despedida	adios gracias	despedida	100.00%
despedida	gracias	despedida	100.00%
negacion	no	negacion	100.00%
negacion	n	negacion	100.00%
negacion	nunca	negacion	100.00%
negacion	no lo creo	negacion	100.00%
negacion	jamás	negacion	100.00%
negacion	NO	negacion	100.00%
negacion	nop	negacion	100.00%
negacion	nopi	negacion	100.00%
saludo	hey	saludo	100.00%
saludo	hola	saludo	100.00%
saludo	hi	saludo	100.00%

saludo	que tal	saludo	100.00%
saludo	buenos días	saludo	100.00%
saludo	buenas tardes	saludo	100.00%
saludo	buenas	saludo	100.00%
saludo	que onda	saludo	100.00%
saludo	buenosdías	saludo	100.00%
saludo	buenosdias	saludo	100.00%
saludo	holis	saludo	100.00%

Tabla 8. Precisión de intenciones.

Tabla de preguntas realizadas en las pruebas de campo con el chatbot extractor de entidades con RASA

Chatbot # 1: Extractor de entidades con RASA			
Preguntas	**Obtuve respuesta**	**No encontró nada en BD**	**No reconoce entidad**
electronica	1	0	0
becas	1	0	0
calificaciones	0	1	0
en donde se encuentra el logo de yo soy uam	0	1	0
en donde se encuentra el edificio AT?	0	1	0
¿cuáles edificios tiene la uami?	0	1	0
cuáles son las evaluaciones de recuperacion	1	0	0
en donde puedo reponer mi credencial	0	1	0
profesores de ingenieria electronica	1	0	0
credenciales	0	1	0
consejo academico	1	0	0
renovacion de credencial	1	0	0
credencial	1	0	0
licenciatura	1	0	0
cómo es la carrera de electronica.	1	0	0
recursos en laboratorios	1	0	0
¿cómo puedo solicitar una constancia?	1	0	0
becas	1	0	0
becas	1	0	0
que tesis puedo hacer en la maestría del PCyTI?	1	0	0
becas	1	0	0
becas	1	0	0
becas	1	0	0
becas	1	0	0

becas	1	0	0
areli	1	0	0
Anzures	1	0	0
Villarreal	1	0	0
Villareal	0	1	0
miguel	1	0	0
Becas	1	0	0
Becas	1	0	0
Licenciaturas de la UAM.	1	0	0
Licenciaturas	1	0	0
Electronica	1	0	0
Becas	1	0	0
Biblioteca	1	0	0
Calendario	1	0	0
cómo me puedo titular	0	1	0
Cuáles son los requisitos para titularme	0	1	0
¿Cuál es la misión de la institución?	1	0	0
Información sobre David Luna Luna	1	0	0
Alma Rosa	1	0	0
beca	1	0	0
Titulación	1	0	0
¿Cuál es el plan de estudio de la licenciatura en Computación?	0	1	0
¿Cuál es el calendario de la UAM?	1	0	0
necesito informacion sobre las becas	1	0	0
Dame información de Anthony Perez	1	0	0
¿Cuántos alumnos tiene el Posgrado en Ciencia y Tecnologías de la Información?	0	1	0
¿Dónde puedo consultar el boligrama de electrónica?	0	1	0
¿En qué dias puedo recoger mi credencial?	1	0	0
Donde puedo consultar la planeación trimestral de tronco común	0	1	0
quien es el coordinador de la carrera de computación?	1	0	0
consultar la titulación por experiencia laboral	0	1	0
¿Cuál es la página de sistemas escolares de posgrado?	1	0	0
Página de biblioteca uami	0	1	0
¿Cuáles son los requisitos de titulación?	1	0	0
¿Cuál es el programa de estudios del pcyti?	0	1	0
¿Dónde encuentro las Tesis a nivel Licenciatura?	0	1	0

¿Cuál es el programa de estudios del pcyti?	0	1	0
¿Cuál es la página de sistemas escolares de posgrado?	1	0	0
inscripción uami	1	0	0
¿Dónde puedo encontrar el calendario escolar?	0	1	0
Información sobre los alumnos de maestría	1	0	0
Módulo de Información Escolar de Alumnos de Licenciatura	0	1	0
¿Cuál es la página del módulo de información escolar??	1	0	0
Luis Martin	1	0	0
LUIS Martín	1	0	0
Luis Martín Rojas Cárdenas	1	0	0
Electrónica	1	0	0
becas	1	0	0
movilidad academica	0	1	0
cursos de celex	1	0	0
tramite de titulacion	1	0	0
Cuál es el calendario actual	0	1	0
Calendario 2022	0	1	0
Calendario	1	0	0
¿Directorio de CBI?	1	0	0
¿Consejo Academico?	1	0	0
Hola, donde puedo tramitar mi credencial	0	1	0
Que carreras hay en la UAM	1	0	0
Eventos escolares	1	0	0
¿Correo electronico?	1	0	0
Cómo puedo tramitar una beca	1	0	0
Módulo de información escolar para licenciatura	0	1	0
Módulo de información escolar	0	1	0
Cómo puedo encontrar el directorio	1	0	0
Cuál es la oferta educativa que ofrecen	1	0	0
Dónde puedo ver a los egresados	1	0	0
Trámite de titulación	1	0	0
Dónde puedo encontrar información de celex	0	1	0
Módulo de información	1	0	0
Que es cosib	0	1	0
Coviuam	0	1	0
Que idiomas tienen	0	1	0

Que posgrados tienen	0	1	0
Planeación computación	0	1	0
Dónde puedo encontrar los posgrados	1	0	0
Dónde puedo encontrar los idiomas	1	0	0
Dónde puedo encontrar la credencial	0	1	0
Licenciatura	1	0	0
¿Qué licenciaturas ofrece la división de CBI?	1	0	0
Investigadores	1	0	0
¿Cuándo es la convocatoria para el examen de admisión?	1	0	0
¿Dónde se ubica la UAMI?	0	1	0
Oferta de posgrado	1	0	0
¿Dónde se encuentra la UAM?	0	1	0
¿Dónde se encuentra la unidad Iztapalapa?	0	1	0
¿Qué modalidad de estudio ofrece la UAM?	0	1	0
Cuál es el calendario actual	0	1	0
Cuál es la planeación de computación	1	0	0
¿Dónde puedo encontrar los idiomas?	1	0	0
Dónde está la uami	1	0	0
¿Cuántos posgrados tienen?	0	1	0
Qué idiomas enseñan	1	0	0
Cuáles son los profesores de CBI	1	0	0
Becas	1	0	0
¿En dónde puedo tramitar de nuevo mi credencial si la perdí?	0	1	0
¿Dónde puedo pedir prestado un libro dentro de la UAMI?	1	0	0
¿La UAMI cuenta con redes sociales?	1	0	0
¿Cómo tramito una reposición de mi credencial?	1	0	0
¿Cuál es el contacto de servicios escolares?	1	0	0
¿Cuáles son los métodos de titulación?	1	0	0
Alma Cordova	0	1	0
Eric del Rincón	0	1	0
Dr Eric	1	0	0
requisitos para entrar a licenciatura	1	0	0
cuales son las licenciaturas de CBI	1	0	0
total de créditos para toefl de posgrado	1	0	0
info sobre celex	1	0	0
Dr Enrique Rodríguez d la Colina	0	1	0

Alma Rosa Cordova Aguilar	1	0	0
costo de constancia de equivalencia de ingles	0	1	0
costo de constancia de maestria	0	1	0
costo de certificado de maestria	0	1	0
costo de título de licenciatura	1	0	0
¿Cuántas divisiones tiene la UAM-Iztapalapa?	0	1	0
¿Cómo puedo tramitar una constancia?	1	0	0
¿Cómo puedo tramitar una beca?	1	0	0
¿Dónde puedo ver mi plan de estudios?	0	1	0
Cuál es el plan de estudios de la licenciatura en Ingeniería electrónica	0	1	0
Donde queda ubicada la universidad	0	1	0
hay acceso al público en general	1	0	0
Licenciatura	1	0	0
Becas	1	0	0
titulación	1	0	0
celex	1	0	0
eventos culturales	1	0	0
proceso de titulación	1	0	0
ubicación UAMI	0	1	0
ubicación	1	0	0
Licenciatura	1	0	0
Oferta educativa	1	0	0
UBICACIÓN	1	0	0
castañeda villa	0	0	1
comunicaciones	1	0	0
becas	1	0	0
circuitos electricos	0	1	0
movilidad	1	0	0
electronica	1	0	0
uam-i	0	0	1
jalpa	0	0	1
ubicación	0	1	0
castañeda	0	0	1
parciales	0	0	1
circuitos 3	0	0	1
contro escolar	0	1	0
sistemas escolares	0	0	1
pascoe	0	0	1
manzur	0	1	0

guerrero poblete	0	1	0
celex	1	0	0
electronica	1	0	0
becas	1	0	0
cafeteria	1	0	0
calificaciones	0	0	1
que se acaba de inagurar en la uami	0	0	1
en donde se encuentra el logo de soy uam	0	0	1
en donde se encuentra el edificio AT	0	0	1
Cuales edificios tiene la uami	0	1	0
quien es el rector de la uami	0	0	1
cuáles son las evaluaciones de recuperacion	1	0	0
como me puedo cambiar de carrera	1	0	0
a que hr abre la librería	1	0	0
en donde puedo reponer mi credencial	0	0	1
quien es el rector de la uam	0	1	0
profesores de ingenieria electronica	1	0	0
horarios ueas	0	0	1
recuperacion	1	0	0
credenciales	0	0	1
consejo academico	1	0	0
electronica	1	0	0
estacionamiento	0	0	1
renocacion de credencial	0	1	0
credencial	1	0	0
direccion	0	0	1
licenciatura	1	0	0
carreras	0	0	1
terraza	0	0	1
horarios	0	0	1
electronica	1	0	0
licenciatura	1	0	0
celex	1	0	0
casa	0	0	1
electronica	1	0	0
carreras	0	0	1
ingles	0	0	1
idiomas	1	0	0
circuitos	0	0	1
kiosko	0	0	1

cafeteria	1	0	0
cómo es la carrera de electronica	1	0	0
becas	1	0	0
biblioteca	1	0	0
recursos en laboratorios	1	0	0
credencial	1	0	0
actividades deportivas	1	0	0
becas	1	0	0
servicio medico	1	0	0
biblioteca	1	0	0
profesor cesar jalpa	1	0	0
licenciaturas	1	0	0
ubicación	1	0	0
mapa curricular de ingenieria electronica	0	0	1
servicio de biblioiteca	0	0	1
servicio escolar	1	0	0
algun programa de becas	0	0	1
bolsa de tranajo	1	0	0
horarios de sistemas escolares	1	0	0
cálculo de varias variables	1	0	0
examenes de admision	1	0	0
bolsa de trabajo uam	1	0	0
dirccion de unidades uam	0	0	1
profesores calculo	1	0	0
carreras en la uam	0	0	1
horario de cafeteria	1	0	0
cómo puedo solicitar una constancia	1	0	0
coordinacion de posgrado	0	0	1
departamento de ingenieria electrica	1	0	0
horario de clases	0	0	1
contacto profesores	0	0	1
	Obtuve respuesta	**No encontro nada en BD**	**No reconoce entidad**
Suma total	**149**	**59**	**34**

Tabla 9. Pruebas de campo del chatbot con extractor de entidades de RASA.

Tabla de preguntas realizadas en las pruebas de campo con el chatbot extractor de entidades con Python

Chatbot #2 Extractor de Entidades con Python			
Pregunta	**Obtuve respuesta**	**No encontro nada en BD**	**No reconoce entidad**
electronica	1	0	0
becas	1	0	0
cafeteria	1	0	0
calificaciones	0	1	0
¿que se acaba de inaugurar en la uami?	0	1	0
en donde se encuentra el logo de yo soy uam	0	1	0
en donde se encuentra el edificio AT?	0	1	0
¿cuáles edificios tiene la uami?	0	1	0
¿Quién es el rector de la uami?	1	0	0
cuáles son las evaluaciones de recuperación	1	0	0
cómo me puedo cambiar de carrera	0	1	0
a que hr abre la librera	0	1	0
en donde puedo reponer mi credencial	0	1	0
¿quién es el rector de la uam?	1	0	0
profesores de ingeniería electrónica	0	1	0
recuperación	1	0	0
credenciales	0	1	0
consejo académico	1	0	0
estacionamiento	1	0	0
renovación de credencial	0	1	0
credencial	1	0	0
dirección	1	0	0
licenciatura	1	0	0
idiomas	1	0	0
cómo es la carrera de electrónica	0	1	0
recursos en laboratorios	0	1	0
actividades deportivas	1	0	0
horario de clases	0	1	0
¿cómo puedo solicitar una constancia?	0	1	0
becas	1	0	0
becas	1	0	0

luis martin	1	0	0
¿Culés son los requisitos para titulación?	1	0	0
¿Cuál es el número de contacto de sistemas escolares?	0	1	0
¿Cuáles son los requisitos para iniciar un posgrado?	0	1	0
¿Qué carreras ofrece la UAMI?	0	1	0
Becas	1	0	0
¿Quién es el coordinador de computación?	0	1	0
cuáles son los proyectos de investigación de maestría	1	0	0
quien es el rector de la uam	1	0	0
cuál es la oferta educativa de cbi?	0	1	0
Trámites para titulación	0	1	0
Convocatoria para becas	1	0	0
¿Cómo te llamas?	1	0	0
cuáles son las líneas de investigación?	1	0	0
¿Costo trimestral de la escuela?	0	1	0
¿Cuáles son los requisitos para titulación?	1	0	0
Calendario Escolar	1	0	0
¿Cuál es la página de módulo de Información escolar?	0	1	0
¿Cuál es la página de módulo de Información escolar?	0	1	0
¿Cuál es la página de módulo de Información escolar?	0	1	0
Módulo de Información escolar	0	1	0
Me siento triste	0	1	0
No me ayudas chatmis	0	1	0
¿Cuál es la página de sistemas escolares de posgrado?	0	1	0
Página de biblioteca uami	1	0	0
Reglamentación universitaria posgrado	0	1	0
¿Cuál es la dirección de la uam i?	0	1	0
¿Cuál es el programa de estudios del pcyti?	0	1	0
becas	1	0	0
Hay becas	1	0	0
Cuál es el calendario actual	0	1	0
Cuál es el calendario	1	0	0
Que idiomas tienen	0	1	0
Qué idiomas hay	1	0	0
Qué posgrados ofrecen	0	1	0
Planeación de computación	0	1	0

Módulo de información escolar	0	1	0
Plan de estudios de computación	0	1	0
Licenciatura	1	0	0
Maestra	1	0	0
Electrónica	1	0	0
Alma Rosa Córdova Aguilar	1	0	0
puntos de toefl para posgrado	0	1	0
titulación de posgrado con toefl	0	1	0
cuantos puntos de toefl necesito para maestría	0	1	0
toefl en celex	0	1	0
información sobre licenciaturas	1	0	0
puntos de toefl para titulación de posgrado	0	1	0
titulación con toefl	0	1	0
cursos en celex	0	1	0
cursos de celex istapalapa	0	1	0
cursis de celex iztapalapa	0	1	0
cursos de celex iztapalapa	0	1	0
puntos de toefl para titulación de posgrado	0	1	0
chatbot en uam	0	1	0
tramite de constancia	0	1	0
constancia	1	0	0
costo de título de posgrado	0	1	0
título de posgrado	0	1	0
Engomados vehiculares	0	1	0
tramite de credencial	0	1	0
credencial	1	0	0
engomado	0	1	0
vehicular	1	0	0
atención a alumnos	1	0	0
atención a alumnos CBI	0	1	0
biblioteca	1	0	0
Licenciatura	1	0	0
Luis Martin	1	0	0
Biblioteca	1	0	0
seminario	1	0	0
Te amo Arely	0	1	0
Sistemas escolares	0	1	0
Sistemas escolares	1	0	0
Firma de titulo	1	0	0
Titulación	1	0	0

facebook uam	0	1	0
facebook	1	0	0
calendario	1	0	0
Electronica	1	0	0
maestria	1	0	0
Posgrado en ciencias y tegnologias de la información	0	1	0
Posgrado en ciencias y tecnologías de la información	1	0	0
inteligencia artificial	1	0	0
cuáles son las licenciaturas que ofrece la universidad	1	0	0
cuál es el plan de estudios de la licenciatura en ingeniera electrónica	0	1	0
hay acceso al público en general	0	1	0
que posgrados hay en la uami	1	0	0
jose job cruz	0	1	0
ingreso doctorado	1	0	0
doctorado	1	0	0
calculo varias variables	1	0	0
becas uam	1	0	0
comunicaciones 1	1	0	0
comunicaciones	1	0	0
cual plan estudios calculo integral	0	1	0
circuitos electricos	0	1	0
norma castañeda	1	0	0
sistemas micropocesadores	0	1	0
sistemas escolares	1	0	0
cesar jalpa	1	0	0
tramitar constancia	0	1	0
renovación credencial	0	1	0
servicios escolares	1	0	0
puedo ver beca	0	1	0
matematicas	1	0	0
alan daniel saldivar muguia	0	1	0
edificio B	0	1	0
reposicion credencial	1	0	0
ecuaciones diferenciales parciales	1	0	0
credencial	1	0	0
dónde encuentro sistemas escolares	0	1	0
horario sistemas escolares	0	1	0
michael pascoe	1	0	0

miguel angel	1	0	0
reinscripcion	1	0	0
jalpa	1	0	0
planes de estudio	1	0	0
quisiera acerca profesores	0	1	0
elecronica	1	0	0
paola aleman quijano	0	1	0
profesores cbi	0	1	0
circuitos electricos 3	0	1	0
horario electronica 3	1	0	0
calculo integral	0	0	0
celex	0	0	1
electronica	1	0	0
becas	1	0	0
cafeteria	1	0	0
calificaciones	0	1	0
que se acaba de inagurar en la uami	0	1	0
en donde se encuentra el logo de soy uam	0	1	0
en donde se encuentra el edificio AT	0	1	0
Cuales edificios tiene la uami	0	1	0
quien es el rector de la uami	1	0	0
cuáles son las evaluaciones de recuperación	1	0	0
cómo me puedo cambiar de carrera	0	1	0
a que hr abre la librería	0	1	0
en donde puedo reponer mi credencial	0	1	0
quien es el rector de la uam	1	0	0
profesores de ingenieria electronica	1	0	0
horarios ueas	0	0	1
recuperacion	1	0	0
credenciales	0	1	0
consejo académico	1	0	0
electronica	1	0	0
estacionamiento	1	0	0
renocacion de credencial	0	1	0
credencial	1	0	0
dirección	1	0	0
licenciatura	1	0	0
carreras	0	1	0
terraza	0	0	1
horarios	0	0	1

electronica	1	0	0
licenciatura	1	0	0
celex	1	0	0
casa	0	0	1
electronica	1	0	0
carreras	0	0	0
ingles	1	0	0
idiomas	1	0	0
circuitos	0	0	1
kiosko	0	0	1
cafeteria	1	0	0
cómo es la carrera de electronica	0	1	0
becas	1	0	0
biblioteca	1	0	0
recursos en laboratorios	0	1	0
credencial	1	0	0
actividades deportivas	1	0	0
becas	1	0	0
servicio medico	1	0	0
biblioteca	1	0	0
profesor cesar jalpa	0	1	0
licenciaturas	1	0	0
ubicación	1	0	0
mapa curricular de ingenieria electronica	0	1	0
servicio de biblioiteca	0	1	0
servicio escolar	1	0	0
algun programa de becas	0	0	1
bolsa de tranajo	0	1	0
horarios de sistemas escolares	0	1	0
cálculo de varias variables	1	0	0
examenes de admision	0	1	0
bolsa de trabajo uam	0	1	0
dirección de unidades uam	0	1	0
profesores calculo	0	1	0
carreras en la uam	0	1	0
horario de cafeteria	0	1	0
cómo puedo solicitar una constancia	1	0	0
coordinación de posgrado	1	0	0
departamento de ingenieria electrica	1	0	0
horario de clases	0	1	0
contacto profesores	0	1	0

	Obtuve respuesta	No encontro nada en BD	No reconoce la entidad
Suma Total	112	104	8

Tabla 10. Pruebas de campo del chatbot con extractor de entidades con Python.

Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA

ACTA DE EXAMEN DE GRADO

No. 00100
Matrícula: 2202800380

Diseño e implementación de una Arquitectura de Chatbot

En la Ciudad de México, se presentaron a las 11:00 horas del día 29 del mes de septiembre del año 2022 en la Unidad Iztapalapa de la Universidad Autónoma Metropolitana, los suscritos miembros del jurado:

DR. ERIC ALFREDO RINCON GARCIA
DR. MARIO ANGEL SILLER GONZALEZ PICO
DR. EDWIN MONTES OROZCO

Bajo la Presidencia del primero y con carácter de Secretario el último, se reunieron para proceder al Examen de Grado cuya denominación aparece al margen, para la obtención del grado de:

MAESTRA EN CIENCIAS (CIENCIAS Y TECNOLOGIAS DE LA INFORMACION)

DE: ARELI ANZURES VILLARREAL

y de acuerdo con el artículo 78 fracción III del Reglamento de Estudios Superiores de la Universidad Autónoma Metropolitana, los miembros del jurado resolvieron:

Aprobar

Acto continuo, el presidente del jurado comunicó a la interesada el resultado de la evaluación y, en caso aprobatorio, le fue tomada la protesta.

ARELI ANZURES VILLARREAL
ALUMNA

REVISÓ

MTRA. ROSALIA SERRANO DE LA PAZ
DIRECTORA DE SISTEMAS ESCOLARES

DIRECTOR DE LA DIVISIÓN DE CBI

Román Linares Romero
DR. ROMAN LINARES ROMERO

PRESIDENTE

DR. ERIC ALFREDO RINCON GARCIA

VOCAL

DR. MARIO ANGEL SILLER GONZALEZ PICO

SECRETARIO

DR. EDWIN MONTES OROZCO