



**UNIVERSIDAD DE CONCEPCIÓN
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA**



CHATBOT PARA CUIDADOR/FAMILIAR DE PACIENTE EN CUIDADOS PALIATIVOS (GES 4)

POR

Antonia Belén Baeza Acuña

Memoria de Título presentada a la Facultad de Ingeniería de la Universidad de Concepción
para optar al título profesional de Ingeniería Civil Biomédica

Profesor(es) Guía
Pamela Guevara Alvez

Profesional Supervisor
Jaime Jiménez Ruiz

Comisión

Eduardo Wiechmann
Fernández

Agosto 2022
Concepción
(Chile)

© 2022 Antonia Baeza Acuña

© 2022 Antonia Belén Baeza Acuña

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento

Agradecimientos

Quiero agradecer a mi familia, a mi madre Gicella y mi padre Victor los cuales son mi ejemplo a seguir y me han enseñado que todo en la vida con esfuerzo se puede lograr, a mi hermana Javiera por ser mi compañera de vida, de grandes batallas, mi confidente y la mejor hermana que me pudo tocar. A los tres les agradezco por ser un pilar fundamental en mi etapa universitaria y sobre todo en mi vida, por todo su amor, paciencia, consuelo y apoyo incondicional.

A mis amigos Tabata, Sofia, Ignacio, Monserrat, Sofia A. y Sergio, les quiero dar las gracias por estar en las buenas, en las malas y mantenerse hasta el día de hoy conmigo, sin ustedes esta etapa no hubiera sido lo mismo.

Agradecer a la empresa HealthTracker y a todo su equipo, quienes han sido fundamentales para el desarrollo de este proyecto y para mi etapa laboral, sobre todo a mi Supervisor Jaime Jiménez el cual estuvo siempre que tenía una duda, que necesitaba orientación o simplemente prestar un oído cuando lo necesite.

Finalmente, quiero dedicar unas palabras a la Antonia del pasado, la cual no se creería hasta donde ha llegado, decirle: “Anto, lo lograste, superaste tus miedos, hiciste lo que pudiste, pero llegaste a la meta. ¡Estoy muy orgullosa de ti!, ahora a creerse el cuento de todo y recuerda, lo limites solo se encuentran en tu propia mente”.

Resumen

Este trabajo tiene por objetivo el desarrollo e implementación de un chatbot dirigido al cuidador informal de pacientes en etapa GES 4, como una herramienta de apoyo tanto para el paciente como para su familia.

El chatbot permite una comunicación continua, directa y las 24 horas del día entre el centro de salud encargado y el entorno del paciente. La resolución de dudas comunes al instante da la facilidad de atender a muchos pacientes/cuidadores a la vez, lo que genera una eficiencia en la respuesta del centro asistencial y permite un descongestionamiento en las tareas del equipo de salud, logrando que estos se enfoquen en otras de mayor prioridad, entre otros beneficios.

A lo largo del trabajo se revisó bibliografía relacionada al desarrollo actual de los chatbot, proyectos asociados a los cuidadores informales, además de identificar y estudiar herramientas que permitieran el desarrollo del proyecto, como los modelos de incrustación de palabras o el uso de APIs para la conexión del chatbot con WhatsApp.

Dentro de todas las opciones para desarrollar el chatbot, se determinó que el modelo *SBERT* era la mejor opción para realizar la incrustación de frases. Después, se utilizan estas incrustaciones para permitirle al computador entender lo que el usuario quiere consultar, pasando de un lenguaje natural a uno entendido por este.

Luego en conjunto a un experto en cuidados paliativos de la empresa HealthTracker, se creó una base de datos de preguntas y respuestas a utilizar, la cual fue integrada al modelo y al resto del algoritmo.

Finalmente se realizaron pruebas por parte del evaluador para corroborar el correcto funcionamiento de la herramienta, realizando un total de 5 evaluaciones para evaluar en su totalidad las funcionalidades del chatbot, con preguntas escritas correctamente, con errores ortográficos y utilizando palabras inexistentes, en todos los casos obteniendo respuestas coherentes a las preguntas realizadas por el usuario.

Abstract

The objective of this work is the development and implementation of a chatbot aimed at the informal caregiver of patients in GES stage 4, as a support tool for both the patient and his family.

The chatbot allows continuous, direct and 24-hour communication between the health center in charge and the patient's environment. The instant resolution of common doubts makes it possible to attend to many patients/caregivers at the same time, which generates efficiency in the response of the health care center and allows the health care team to decongest tasks, allowing them to focus on other higher-priority tasks, among other benefits.

Throughout the work we reviewed literature related to the current development of chatbots, projects associated with informal caregivers, in addition to identifying and studying tools that would allow the development of the project, such as word embedding models or the use of APIs for the connection of the chatbot with WhatsApp.

Within all the options to develop the chatbot, it was determined that the SBERT model was the best option to perform the embedding of phrases. Then, these embeddings are used to allow the computer to understand what the user wants to consult, going from a natural language to one understood by the user.

Then, together with an expert in palliative care from the company HealthTracker, a database of questions and answers to be used was created and integrated into the model and the rest of the algorithm.

Finally, tests were carried out by the evaluator to corroborate the correct functioning of the tool, performing a total of 5 evaluations to fully assess the chatbot's functionalities, with questions written correctly, with spelling errors and using non-existent words, in all cases obtaining coherent answers to the questions asked by the user.

Tabla de Contenidos

CAPÍTULO 1. INTRODUCCIÓN	11
1.1. INTRODUCCIÓN GENERAL.....	11
1.2. OBJETIVOS.....	14
1.2.1 <i>Objetivo General</i>	14
1.2.2 <i>Objetivos Específicos</i>	14
1.3. ALCANCES Y LIMITACIONES.....	15
1.4. METODOLOGÍA.....	15
1.5. TEMARIO.....	16
CAPÍTULO 2. MARCO TEÓRICO	17
2.1. INTRODUCCIÓN.....	17
2.2. GARANTÍAS EXPLÍCITAS DE SALUD (GES) - EX PLAN AUGE.....	17
2.2.1 <i>Alivio del Dolor y Cuidados Paliativos por Cáncer Avanzado (GES 4)</i>	19
2.3. PROCESAMIENTO DEL LENGUAJE NATURAL.....	21
2.3.1 <i>Python</i>	21
2.3.2 <i>Inteligencia Artificial (IA)</i>	21
2.3.3 <i>Machine Learning</i>	23
2.3.4 <i>Redes Neuronales</i>	24
2.3.5 <i>Procesamiento del Lenguaje Natural (PLN o NLP)</i>	25
2.3.6 <i>Word Embeddings</i>	28
2.3.7 <i>Transformers o Transformadores</i>	31
2.3.8 <i>BERT (Bidirectional Encoder Representations from Transformers)</i>	33
2.4. CHATBOT.....	36
2.4.1 <i>Avances a través del Tiempo</i>	37
2.4.2 <i>Tipos de Tecnologías Chatbot</i>	38
2.4.3 <i>Otras herramientas que facilitan el desarrollo de Chatbot</i>	39
2.5. DISCUSIÓN.....	41
CAPÍTULO 3. ESTADO DEL ARTE	43
3.1.1 <i>Desarrollo de Chatbot</i>	43
3.1.2 <i>Manuales Asociados a Cuidados Paliativos</i>	44
3.1.3 <i>Herramientas de apoyo al cuidador</i>	45
3.2. DISCUSIÓN.....	46
CAPÍTULO 4. DISEÑO Y DESARROLLO DEL SOFTWARE	47
4.1. INTRODUCCIÓN.....	47
4.2. DISEÑO DEL CHATBOT.....	47
4.3. DESARROLLO DEL CHATBOT.....	49
4.3.1 <i>Adquisición de software y bibliotecas</i>	49
4.3.2 <i>Generación de datos</i>	50
4.3.3 <i>Incrustaciones o WordEmbeddings</i>	51
4.3.4 <i>Similitud entre la consulta ingresada por el usuario y aquellas disponibles en la base de datos del Chatbot</i>	52
4.3.5 <i>Selección de respuesta al usuario</i>	53
4.4. CONEXIÓN CON WHATSAPP.....	56
4.5. DISCUSIÓN.....	57
CAPÍTULO 5. EVALUACIÓN CHATBOT	60
5.1. INTRODUCCIÓN.....	60
5.2. RESULTADOS EVALUACIÓN.....	60
5.3. DISCUSIÓN.....	63
CAPÍTULO 6. CONCLUSIONES	65
6.1. SUMARIO Y DISCUSIÓN.....	65
6.2. CONCLUSIÓN.....	66

6.3. TRABAJOS FUTUROS.....	67
BIBLIOGRAFÍA	68
ANEXO A. REQUERIMIENTOS DEL CHATBOT	74
ANEXO B. PASO A PASO PARA LA DESCARGA DE LOS PAQUETES DE NLTK.....	75
ANEXO C. FUNCIONALIDADES DEL CHATBOT	76
ANEXO D. TABLA DE EVALUACIÓN CHATBOT Y RESULTADOS EVALUACIÓN.....	79

Lista de Tablas

Tabla 2.1 CONDICIONES DE SALUD ASOCIADAS AL CÁNCER INCLUIDAS EN EL RÉGIMEN GENERAL DE GARANTÍAS	18
Tabla 4.1 Requerimientos funcionales (RF) y Requerimientos No Funcionales (RNF) del Chatbot	48
TABLA 4.2 PREGUNTAS A UTILIZAR EN EL CHATBOT	51

Lista de Figuras

Figura 2.1: Pasos a seguir para atención GES en pacientes ISAPRE y los respectivos lugares de atención [17].....	19
Figura 2.2: Pasos a seguir para atención GES en pacientes FONASA y los respectivos lugares de atención [17].....	19
Figura 2.3: Beneficios para beneficiarios de GES 4 [17].....	20
Figura 2.4 Sub Campos de la inteligencia artificial [21].....	22
Figura 2.5 Estructura base de una red neuronal [29].....	24
Figura 2.6: Arquitectura que debería seguir el Procesamiento natural del lenguaje.	27
Figura 2.7: Ejemplo de representaciones distribuidas.....	28
Figura 2.8: Similitud de palabras según Word2Vec en un espacio 2D [81]	30
Figura 2.9: Arquitectura de un Transformador	32
Figura 2.10 Funcionamiento del sistema codificador-decodificador en los Transformers	33
Figura 2.11 Ejemplo de la bidireccionalidad de BERT para comprender el contexto de una oración [46]	34
Figura 2.12: Tendencia en la búsqueda y uso del término “Chatbot” entre los años 2010-2021 según Google Trends.	37
Figura 2.13: Sectores que más se beneficiarán de los chatbots [35]	38
Figura 2.14 Diagrama de flujo del proceso utilizado por Chatterbot.....	40
Figura 4.1 Diagrama de flujo de las etapas del chatbot.....	48
Figura 4.2 Visualización de la interfaz del usuario al ingresar una consulta	54
Figura 4.3 Resultados del Chatbot al ingresar la misma pregunta de variadas formas.....	55
Figura 4.4 Visualización del interfaz de usuario al darle la opción de realizar otra consulta	56
Figura 5.1 Resultados del Chatbot al ingresar la misma pregunta de variadas formas en WhatsApp	61

Abreviaciones

Mayúsculas

AIRC: Agencia Internacional para la Investigación de Cáncer, *International Agency of Research on Cancer*

ALCP: Asociación Latinoamericana de Cuidados Paliativos

API: Interfaz de Programación de Aplicaciones, *Application Programming Interfaces*

BERT: *Bidirectional Encoder Representations from Transformers*

CAEC: Cobertura Adicional para Enfermedades Catastróficas

CBOW: Modelo de bolsas de palabras, *Continuous Bag of Words*

ELA: Esclerosis Lateral Amiotrófica

FONASA: Fondo Nacional de Salud

GCO: Observatorio Global del Cáncer, *Global Cancer Observatory*

GES: Garantías Explícitas en Salud

IA: Inteligencia Artificial

ISAPRE: Instituciones de Salud Previsional

LSA: Análisis Semántico Latente, *Latent Semantic Analysis*

NNs: Redes Neuronales, *Neural Networks o NNs*

NLP: Procesamiento Natural del Lenguaje, *Natural Language Processing*

NLTK: Kit de herramientas de lenguaje natural, *Natural Language Toolkit*

OMS: Organización Mundial de la Salud

RF: Requisitos funcionales

RN: Redes Neuronales

RNF: Requisitos no funcionales

SBERT: *SentenceTransformers*

SECPAL: Sociedad Española de Cuidados Paliativos

Capítulo 1. Introducción

1.1. Introducción General

En la actualidad el cáncer es la segunda causa de muerte a nivel mundial, la cual provoca cerca de 10 millones de muertes cada año [1]. En el caso de nuestro país, esta enfermedad se ubica en el primer puesto dentro de las causas de muerte en la población chilena, desplazando después de 30 años a las afecciones del sistema circulatorio y cardiovascular [2]. Esta enfermedad no solo genera millones de muertes al año, sino que además conlleva altos costos económicos y sociales, tanto por los valores asignados a los respectivos tratamientos, como por el compromiso que produce en las vidas de las personas directamente afectadas y de su entorno familiar y/o cuidadores. Según los datos de Global Cancer Observatory (GCO) 2020, en Chile se diagnosticaron 54.227 nuevos casos de cáncer, de los cuales 28.779 corresponden a hombres y 25.448 a mujeres [3]. Por otra parte, según estimaciones realizadas por la Agencia Internacional para la Investigación del Cáncer (AIRC), para el año 2040 en Chile, habrán 94.807 nuevos casos de cáncer, lo que corresponde a un aumento del 74.8% en los próximos 20 años [4]. Además, durante el año 2020, en el país se produjeron 28.583 muertes a causa del cáncer, las que llegarán a 55.698 para el año 2049, según las estimaciones mencionadas anteriormente.

Estos datos son un claro ejemplo de que el cáncer será uno de los grandes desafíos sanitarios de los próximos años, tanto en Chile como el mundo. Estas proyecciones son importantísimas para que el Estado y el sistema de salud implementen las medidas necesarias, las cuales abarcan temas como: la prevención, diagnóstico, tratamiento, cuidados paliativos, entre otros.

El dolor en pacientes con cáncer, está presente en un tercio de los pacientes con tratamiento activo, y en situación avanzada/terminal llega a afectar entre 70% al 90% de los mismos. Pueden existir una variedad de tipos de dolor, pero de forma simplificada cerca del 80% del dolor es tumoral y el 20% restante es debido al tratamiento por el que pasa el enfermo [5]. Existe evidencia que sugiere que hay una relación directa entre el dolor que sienten los pacientes y su entorno social. Es decir, si por ejemplo los niveles de dolor son altos, hay factores que pueden aumentarlo, como la presión y el estrés presente en el cuidador y la sensación de carga sobre la familia [6]. Además, la prevalencia del dolor se incrementa con

la progresión de la enfermedad oncológica, mientras que la intensidad, tipo y localización varía de acuerdo a la progresión de la patología, el tratamiento empleado, sitio primario del cáncer y la extensión [7].

Según los datos generados por el Instituto Nacional de Estadística en el año 2018, la población total en Chile asciende a 18.751.405, de los cuales 14.242.655 de personas corresponden a usuarios del Fondo Nacional de Salud (FONASA), lo que representa al 79.2% de la población. [8]. Cabe destacar, que la población FONASA puede acceder a una variedad de prestaciones en todos los niveles de complejidad de la red asistencial. En el caso de las prestaciones relacionadas al cáncer, estas se dividen en niveles, los cuales son:

- En el primer nivel encontramos servicios asociados a la atención primaria, la cual tiene como objetivo la promoción y prevención del cáncer.
- En el nivel secundario se encuentra la oferta de servicios que incluyen diagnóstico, tratamiento ambulatorio, rehabilitación y cuidados paliativos.
- El tercer nivel se focaliza en la oferta de mayor complejidad, la cual se orienta a los tratamientos más intensivos [9].

Actualmente en Chile el acceso a estas prestaciones se realiza según el flujo de derivación de la red, tanto para cánceres incluidos en el programa de Garantías Explícitas en Salud (GES) -antes plan AUGE-, como para aquellos que cuentan con otro modo de financiamiento [10].

De acuerdo a la Organización Mundial de la Salud (OMS), dentro de los factores para reducir la carga de morbilidad por cáncer encontramos la prevención, detección y tratamiento temprano, y el cuidado adecuado de los pacientes [1]. A continuación, para efectos de este proyecto nos enfocaremos en este último punto mencionado.

La finalidad de los cuidados paliativos no es curar el cáncer, sino que aliviar todos aquellos síntomas posibles, lo que causa una mejora en la calidad de vida de los pacientes y sus respectivas familias. Estos cuidados permiten que las personas puedan vivir más cómodamente, y son sumamente importantes en aquellos lugares donde encontramos pacientes con etapas avanzadas de la enfermedad y con muy pocas probabilidades de cura. Que existan estrategias eficientes de salud pública que abarquen la atención fuera de los establecimientos de salud y en el propio hogar de los usuarios es fundamental para ofrecer

servicios de cuidados paliativos y de alivio del dolor que puedan ayudar a los enfermos y a sus familias [1].

Chile cuenta con la ley N° 19.966 de cuidados paliativos o “Ley de Garantías Explícitas en Salud”, la cual incluye a los cuidados paliativos, los cuales estaban disponibles exclusivamente para pacientes con patologías oncológicas, impidiendo el acceso a este a quienes la necesiten, pero el día 14 de octubre del 2021 se promulgó la ley N° 21.375, la cual “...reconoce que toda persona que padece una enfermedad terminal o grave tiene derecho a cuidados paliativos...” [11]. Por esto en la actualidad, y a contar de cinco meses posterior a la fecha de publicación en el Diario Oficial, la ley entrará en vigencia. Por lo tanto, en Chile los cuidados paliativos estarán disponibles no solo para la población oncológica, sino que para todos los enfermos terminales o con patologías graves.

Existen dos tipos de cuidadores paliativos. Por un lado, el cuidador formal, es un profesional de la salud especialista en este tipo de cuidados y que forma parte de un centro de salud especializado. Por otro lado, el cuidador informal, es generalmente un miembro de la familia del enfermo, que se encarga de los cuidados asociados al paciente, así como la contención y el apoyo a este, desarrollando todo este trabajo en el ámbito doméstico. No todos los cuidadores de pacientes en cuidados paliativos son directamente un familiar, pero cuando lo son, se ha estudiado que la presión y el estrés generado por tener a cargo a un enfermo con este tipo de cuidados, les afecta en distintos ámbitos como los relacionados a su salud física y mental, relaciones interpersonales, finanzas, entre otros. Sumado a lo anteriormente mencionado, y que además no son personas especialistas para tener a cargo a un paciente de esta categoría, surgen un sinnúmero de dudas acerca de cómo realizar bien ese trabajo o qué hacer ante una emergencia. Se tiene como solución actual acercarse a sus respectivos centros de salud, llamar a sus médicos de cabecera que muchas veces no cuentan con el tiempo suficiente para responder este tipo de consultas, realizar las respectivas dudas en teleconsultas sincrónicas cada cierto tiempo a sus especialistas, o bien buscar otro tipo de soluciones.

Dentro de los grandes avances que día a día surgen relacionados a las nuevas tecnologías, se encuentra un tipo de Inteligencia Artificial (IA) que en pocas palabras simula ser una persona detrás de un dispositivo, ya sea celular o un computador, contestando preguntas vía chat o mensajería, lo que se conoce como Chatbot. Los Chatbot se encargan de

simular una conversación entre humanos, generando respuestas automatizadas gracias al uso de algoritmos y que interactúan en tiempo real con las personas. Esta herramienta es innovadora para el sector salud, debido a que permite acortar tiempos de respuesta y distancia [12].

Con la llegada de la era digital y la situación de emergencia sanitaria, la asistencia médica a distancia (telemedicina) multiplica su presencia ante la dificultad de la atención convencional. En este sentido, con el uso de los “Chatbot” en salud, las compañías e instituciones relacionadas a la atención médica pueden proporcionar soluciones de forma rápida a las diferentes consultas y preocupaciones de los usuarios [13].

Por lo tanto, con toda la información entregada anteriormente se propone diseñar y desarrollar un Chatbot como herramienta de apoyo y educación para el cuidador a cargo de pacientes en GES 4 “Alivio del Dolor y Cuidados Paliativos”, con el objetivo de ayudar a los cuidadores a conocer e instruirse en técnicas de cuidado y autocuidado. La finalidad de este proyecto es poder brindar una ayuda a las personas encargadas de pacientes en GES 4 en tiempo real, permitiendo que los cuidadores sepan lo que deben hacer, de qué forma y en qué momento hacerlo, sin necesidad de acercarse a un centro de salud para responder estas interrogantes. Además, según el reporte “Latin America Digital Transformation Report 2021” de *Atlantico*, la aplicación móvil más utilizada en Latinoamérica es *WhatsApp*, es por esto que será el medio a través del cual se desarrollará el proyecto [14].

1.2. Objetivos

1.2.1 Objetivo General

Desarrollar un chatbot como herramienta de apoyo y educación para el cuidador a cargo de pacientes en GES 4 “Alivio del Dolor y Cuidados Paliativos”.

1.2.2 Objetivos Específicos

- Definir contenidos a incorporar en la herramienta en base a la información básica que se desea entregar a los usuarios y a la interacción que se quiere tener con estos.
- Implementar la herramienta de Chatbot en la plataforma seleccionada y demostrar que funciona de forma correcta según lo requerido.

- Evaluar el chatbot con un experto en el área de la salud que se encuentre desempeñando labores en áreas relacionadas a pacientes oncológicos que pudieran necesitar o que necesiten de cuidados paliativos.

1.3. Alcances y Limitaciones

- En este proyecto se trabajará en colaboración con la empresa HEALTH TRACKER ANALYTICS, la cual proporcionará los datos de preguntas y respuestas asociadas a los cuidados paliativos adquirida a través de una cuenta de *Whatsapp* corporativo de un centro de salud oncológico. Esta debe ser estructurada y analizada para poder generar una base de datos preprocesada que pueda ser utilizada en el chatbot. Además, la empresa facilitará un experto, que será el encargado de evaluar el funcionamiento del proyecto.
- El lenguaje de programación a utilizar será *Python*.
- La plataforma a utilizar para la implementación del Chatbot será *WhatsApp*.

1.4. Metodología

El presente proyecto se realizará haciendo uso del lenguaje de programación *Python* en su versión 3.8 y una serie de bibliotecas y modelos asociados al Procesamiento Natural del Lenguaje (NLP), tales como Natural Language Toolkit (NLTK), Scikit-learn y BERT. Estas, a lo largo del trabajo se evaluarán y definirá cuales finalmente se utilizarán para el desarrollo del chatbot.

El primer paso a seguir es realizar una búsqueda bibliográfica acerca de las distintas herramientas que en la actualidad se utilizan para la elaboración de un Chatbot y el estudio de la información necesaria para el desarrollo del algoritmo. Además, se seleccionará un tipo de chatbot a utilizar en el proyecto según la facilidad de uso e implementación, costos, entre otros factores a evaluar.

Luego, se desarrollará la base de datos de preguntas que se podrán responder por el chatbot, así como sus respectivas respuestas. Posterior al desarrollo de la base de datos, se dará a conocer el diseño escogido para la elaboración de la herramienta. La aplicación del Chatbot se realizará a través del uso de la API de la aplicación móvil *WhatsApp*, siendo esta

el intermediario entre el cuidador/familiar y el banco de preguntas y respuestas asociadas a los cuidados paliativos.

Finalmente, se desarrollará y evaluará el software por una experta en cuidados paliativos perteneciente a la empresa HealthTracker, a través de una rúbrica de evaluación desarrollada a lo largo del proyecto.

1.5. Temario

- Capítulo 1: En este capítulo se introduce el tema a tratar, se definen los objetivos, metodología, alcances y limitaciones para llevar a cabo este proyecto.
- Capítulo 2: Aquí se engloba el marco teórico, el cual consta de una revisión bibliográfica sobre los conceptos básicos a comprender para el desarrollo de este trabajo.
- Capítulo 3: En esta parte del proyecto se da a conocer el estado del arte de los chatbot y temas relacionados al proyecto, como los distintos manuales asociados a cuidados paliativos, herramienta para los cuidadores, entre otros.
- Capítulo 4: Este capítulo habla del diseño escogido para la elaboración del chatbot y su posterior desarrollo, asiendo mención de los elementos utilizados para la elaboración del algoritmo.
- Capítulo 5: Se presenta la evaluación de la herramienta por parte del experto escogido.
- Capítulo 6: Se presentan las conclusiones generales del proyecto. Además, se hace mención del trabajo futuro del chatbot, con el objetivo de mejorar la herramienta.

Capítulo 2. Marco Teórico

2.1. Introducción

Un cuidador informal se define como, “*Una persona que generalmente es un familiar del enfermo a cuidar, la cual convive y tiene la responsabilidad de proveer recursos que el paciente necesite, debido a la incapacidad de autosustentarse*” [15]. Cuidar a una persona con una enfermedad crónica como el cáncer implica un cambio en la vida del cuidador debido a que se modifican las responsabilidades o rutinas a las que está acostumbrado. Además, este debe tomar decisiones en medio de escenarios complicados, asumir responsabilidades o realizar tareas y acciones de cuidado físico, social, religioso y psicológico para atender las necesidades de la persona cuidada que día a día van variando. A todo lo mencionado anteriormente se debe agregar, además, el estrés que conlleva tener un familiar con cáncer que requiere de este tipo de cuidados, la poca experticia con la que cuentan los cuidadores, en el caso de ser un familiar, y la necesidad de ser ayudados por parte de los equipos de cuidados paliativos que existen en los respectivos centros de salud.

En este capítulo se introducen conceptos asociados a pacientes en GES 4 “Alivio del Dolor y Cuidados Paliativos”. También se presenta información relacionada al lenguaje de programación a utilizar y sus respectivas bibliotecas, las diferentes herramientas para la implementación del software a desarrollar y otros conceptos que son necesarios comprender para el presente trabajo.

Finalmente, se presenta información relacionada a otros trabajos, aplicaciones, programas o manuales que sirven como base para este proyecto y que tengan un objetivo relacionado o similar al del trabajo.

2.2. Garantías Explícitas de Salud (GES) - Ex plan AUGE

El *GES* es un derecho que tienen todas las personas que son beneficiarias de *FONASA* o de alguna Institución de Salud Previsional (*ISAPRE*), y que básicamente consiste en garantizar cuatro garantías exigibles por ley a 85 problemas de salud vigentes. Estas garantías son [10][16]:

- Acceso: “*Derecho por ley de la prestación de salud*”.
- Oportunidad: “*Tiempo máximo de espera para el otorgamiento de las prestaciones*”.

- Calidad: “*Otorgamiento de las prestaciones por un prestador acreditado o certificado*”.
- Protección Financiera: “*La persona beneficiaria cancelará un porcentaje de la afiliación*”.

Para que un paciente pueda ser atendido por el *GES*, debe cumplir ciertos requisitos mínimos los cuales son: ser beneficiario de *FONASA* o alguna *ISAPRE*, la enfermedad que afecta al paciente debe ser alguna de las 85 patologías *GES* vigentes, cumplir con las condiciones especiales de edad, estado de salud u otras, definidas para cada enfermedad, debe acceder a prestaciones debidamente prescritas por un profesional de salud y que estén consideradas en los tratamientos *GES* y por último el paciente debe atenderse solo en la red de prestadores *GES/AUGE* que determine *FONASA* o su *ISAPRE* [10][16].

En la actualidad en el *GES* se pueden encontrar una variedad de cánceres que se encuentran cubiertos, además del “Alivio del dolor y cuidados Paliativos por Cáncer Avanzado” que es el público objetivo del presente trabajo (ver Tabla 2.1).

Tabla 2.1 CONDICIONES DE SALUD ASOCIADAS AL CÁNCER INCLUIDAS EN EL RÉGIMEN GENERAL DE GARANTÍAS

Tratamiento de Cánceres	Número del Problema de Salud
Cáncer cervicouterino	GES-3
Alivio del dolor y cuidados paliativos por cáncer avanzado	GES-4
Cáncer de mama en personas de 15 años y más	GES-8
Cáncer en personas menores de 15 años	GES-14
Cáncer de testículo en personas de 15 años y más	GES-16
Linfomas en personas de 15 años y más	GES-17
Cáncer gástrico	GES-27
Cáncer de próstata en personas de 15 años y más	GES-28
Leucemia en personas de 15 años y más	GES-45
Cáncer colorrectal en personas de 15 años y más	GES-70
Cáncer de ovario epitelial	GES-71
Cáncer vesical en personas de 15 años y más	GES-72
Osteosarcoma en personas de 15 años y más	GES-73
Cáncer de pulmón en personas de 15 años y más	GES-81
Cáncer de tiroides diferenciado y medular en personas de 15 años y más	GES-82
Cáncer renal en personas de 15 años y más	GES-83
Mieloma múltiple en personas de 15 años y más	GES-84

Fuente: Plan Nacional de Cáncer, Gobierno de Chile [9]

2.2.1 Alivio del Dolor y Cuidados Paliativos por Cáncer Avanzado (GES 4)

El Alivio del Dolor y Cuidados Paliativos por Cáncer Avanzado es el problema de salud número 4 incluido en el *GES*, a él pueden acceder “...*personas de cualquier edad con diagnóstico confirmado de cáncer, ya sea progresivo o no progresivo...*” [17]. A este beneficio como se mencionó anteriormente, se puede acceder gracias a la confirmación diagnóstica de un profesional médico, y según la previsión médica del paciente, el lugar donde se realiza la atención *GES* cambia, como se puede observar en las Figura 2.1 y Figura 2.2.



Figura 2.1: Pasos a seguir para atención GES en pacientes ISAPRE y los respectivos lugares de atención [17]



Figura 2.2: Pasos a seguir para atención GES en pacientes FONASA y los respectivos lugares de atención [17]

Cabe destacar que los pacientes beneficiarios con el *GES 4* son receptores de ciertos beneficios como medicamentos, insumos y exámenes, según el listado de prestaciones específico para el problema de salud asociado, tanto para usuarios *FONASA* como *ISAPRE* (ver en Figura 2.3). Además, se generalizan los cuidados paliativos, los cuales consisten en asistencia al paciente y a su entorno por un equipo multidisciplinario, con el objetivo de ser una ayuda y mejorar la calidad de vida del paciente y su familia.



Figura 2.3: Beneficios para beneficiarios de GES 4 [17]

Como se mencionó anteriormente, en octubre del 2021 se promulgó la Ley de Garantías Universales para Cuidados Paliativos, permitiendo que todas aquellas personas con diagnósticos de enfermedades terminales o graves puedan acceder a servicios de cuidados paliativos de calidad, costo-efectivos y oportunos entregados por el estado de igual forma que los pacientes oncológicos. De esta manera se logró ampliar la cobertura que existía solo para pacientes oncológicos a pacientes con enfermedades no-oncológicas como pudieran ser enfermedades respiratorias crónicas, insuficiencia cardíaca crónica, esclerosis lateral amiotrófica (ELA), entre otras. Además, con esta nueva ley se deberían realizar cambios en las garantías *GES*, como, adaptar las canastas de fármacos entregadas para así incluir las enfermedades no oncológicas, permitiendo de esta forma que los pacientes tengan acceso a una variedad de medicamentos que les permitan mitigar los síntomas asociados a sus respectivas patologías. Otro punto a modificar es el fortalecer los servicios de atención paliativa en centros de salud y hospitales, además de reforzar la formación de la fuerza de trabajo necesaria para la atención paliativa, debido a que un muy bajo porcentaje de profesionales de la salud cuenta con las habilidades y capacidades necesarias para ejercer la atención o cuidados paliativos de forma efectiva. De hecho, según el “Atlas de Cuidados Paliativos en Latinoamérica 2020”, en Chile solo 3 de 20 escuelas de medicina, 2 de 45 escuelas de enfermería, 0 de 53 escuelas de psicología y 0 de 49 escuelas de trabajo social,

cuentan con cursos independientes de cuidados paliativos en sus respectivas mallas curriculares [18]. Se evidencia así el déficit que hay en nuestro país en esta materia y la poca iniciativa o baja prioridad de parte de las casas de estudio en la formación de esta área.

2.3. Procesamiento del Lenguaje Natural

2.3.1 Python

Python es un lenguaje de programación que permite trabajar rápidamente e integrar sistemas de forma eficaz. Está considerado dentro de los lenguajes de programación de alto nivel que se utiliza para el desarrollo de todo tipo de programas y aplicaciones, y su gran ventaja es que es un lenguaje interpretado. Dicho de otra manera, no necesita de un compilador para ejecutar los algoritmos escritos en él, ya que cuenta con un interpretador el cual permite que estos se ejecuten directamente en el computador. Por lo tanto, no es necesario llevar el algoritmo a lenguaje máquina, a diferencia de otros lenguajes de programación como C. Además, *Python* es muy fácil e intuitivo de usar debido a su similitud con el lenguaje humano, y al ser un lenguaje multiplataforma y de código abierto permite desarrollar una gran variedad de softwares. Adicionalmente, en los últimos años ha facilitado el trabajar con big data, machine learning, data science e inteligencia artificial, entre muchos otros campos [19][20].

2.3.2 Inteligencia Artificial (IA)

Si bien existen variadas definiciones para la inteligencia artificial (IA), se podría decir que es una disciplina que mezcla la ciencia y la ingeniería para hacer máquinas programadas e inteligentes, que imitan la inteligencia humana para poder ejecutar variadas tareas, pudiendo además mejorar su performance a partir de la información que pueden recopilar [21][22]. En su forma más simple, la IA es un campo que combina conjuntos de datos y la informática para la resolución de ciertos problemas, abarcando además subcampos como el machine learning y deep learning (Figura 2.4).

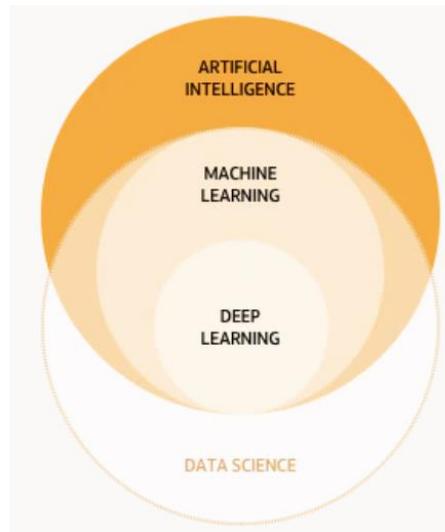


Figura 2.4 Sub Campos de la inteligencia artificial [21]

En la actualidad la *IA* tiene una gran variedad de aplicaciones en el mundo real, entre las que podemos encontrar: fabricación de robots, monitoreo de redes sociales, herramientas del lenguaje natural, mapeo de enfermedades como el COVID-19, vehículos autónomos, reconocimiento de voz, servicio al cliente, chatbot que utilizan *IA* para comprender y responder de forma más eficiente los problemas de los clientes, entre otros.

A continuación, se definirán las subáreas de la *IA*:

- ***Machine Learning***: Es un subcampo de la inteligencia artificial que faculta a un sistema para aprender de los datos en vez de aprender de la programación explícita. Mientras más datos de entrenamiento son ingresados al algoritmo, se pueden obtener modelos y predicciones más precisas, basándose solo en los datos [23].
- ***Deep Learning***: Es un subconjunto dentro del machine learning, que hace uso de redes neuronales, las cuales están inspiradas en el funcionamiento del cerebro humano para aprender grandes cantidades de datos. Su funcionamiento se basa en darle un modelo al computador que permita evaluar ejemplos junto a una pequeña cantidad de instrucciones, para que así el modelo se pueda modificar en los momentos que se produzcan errores. A medida que el tiempo pasa, estos modelos son capaces de dar solución a los diferentes problemas de una forma mucho más precisa, debido a que el sistema es capaz de extraer patrones [24].

2.3.3 Machine Learning

El Machine Learning tiene como objetivo principal que los algoritmos se capaciten para que así puedan encontrar correlaciones y patrones en grandes cantidades de datos, y según el análisis de estos datos tomar las mejores predicciones. Dependiendo de la naturaleza, el conjunto de datos que se utilice y qué resultados se esperan, se puede utilizar uno de los 3 tipos de modelos de aprendizaje conocidos: no supervisado, supervisado y de refuerzo.

- **Aprendizaje supervisado:** En este tipo de aprendizaje al computador o máquina se le enseña mediante la generación de modelos, los cuales predicen los resultados en base a ejemplos “etiquetados” o “históricos” de cierta variable de salida, intentando encontrar una dependencia y relación entre los datos de entrada y el resultado predictivo objetivo, con el propósito de predecir los valores de salida para nuevos datos, en función de lo aprendido en el conjunto de datos etiquetados. Este tipo de aprendizaje requiere en su entrenamiento de datos etiquetados, es decir, datos en los que se conoce su resultado y con los que se aprende a asignar una etiqueta de salida apropiada a un cierto valor [25][26][27].
- **Aprendizaje no supervisado:** Este tipo de modelo tiene lugar en aquellos datos que no disponen de etiquetas para el entrenamiento del algoritmo, y que además no están estructurados, es decir, utiliza los datos en bruto. El funcionamiento de este aprendizaje se basa en que la máquina analiza los datos de entrada para identificar correlaciones y patrones entre estos, en función de las características que puedan presentar. La gran diferencia con el aprendizaje supervisado es que, al no contar con etiquetas, no hay forma de evaluar los resultados directamente. En este caso, mientras más información es ingresada en el algoritmo, mejores predicciones y resultados se logran obtener de él [25][26][28].
- **Aprendizaje por refuerzo:** Este modelo se basa en mejorar la predicción del algoritmo haciendo uso de un proceso de retroalimentación o feedback, es decir, el sistema debe aprender por experiencia y recompensa. Este modelo se podría comparar a enseñarle matemáticas a alguien, en donde se le enseña las diferentes reglas que existen en esta materia, y la persona aumenta su habilidad y conocimiento a través de la práctica [25][26].

2.3.4 Redes Neuronales

Las redes neuronales (RN) pertenecen a una subárea del Machine Learning y están en el corazón de los algoritmos del Deep Learning. Su estructura se basa en el cerebro humano tratando de imitar la forma en que las neuronas se comunican entre sí, con el objetivo de reconocer patrones numéricos, los cuales están contenidos en vectores, a los que se deben traducir todos los datos del mundo real, ya sean texto, imágenes, sonidos o series temporales.

Estas redes están compuestas por capas de nodos, las cuales están constituidas por una capa de entrada, una o varias capas ocultas y una capa de salida (Figura 2.5).

En este sistema, cada neurona o nodo se conecta a otra y tiene un umbral y peso asociado, en donde si la salida de un nodo está por sobre el valor del umbral escogido, dicha neurona se activa, permitiendo que se envíen los datos a la siguiente capa de la red. En caso contrario, no se transmiten los datos.

Una característica que tienen estas redes es que dependen de los datos que uno les dé en su entrenamiento para poder aprender e ir mejorando la precisión de la red con el tiempo. La ventaja de este sistema es que una vez que los algoritmos logran ajustar su precisión son grandes herramientas para utilizar en áreas como la Inteligencia Artificial [29].

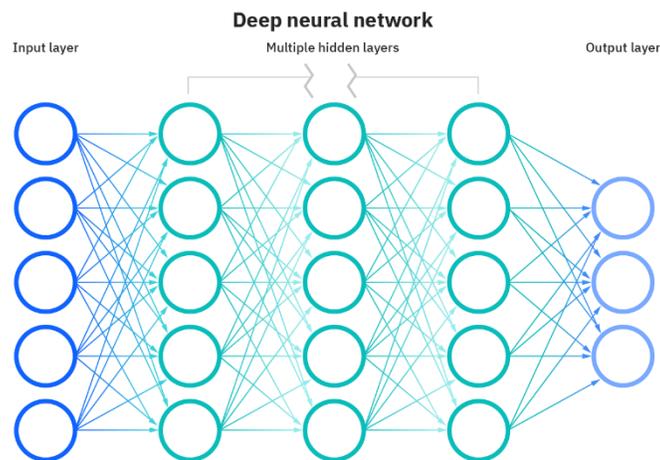


Figura 2.5 Estructura base de una red neuronal [29]

Las *Neural Networks* o *NNs* son un medio para el aprendizaje automático, en el que una computadora logra aprender a realizar alguna tarea analizando ejemplos de entrenamiento. Generalmente, los ejemplos utilizados para entrenar estas redes han sido

etiquetados manualmente con anticipación, lo que hace que estos sistemas aprendan a través de un aprendizaje supervisado.

Hoy en día existen variados tipos de redes neuronales, que van variando de acuerdo con el tipo de aprendizaje que se les aplicó y el fin con el que se las quiere usar:

- Perceptrón: Es la RN en su forma original y más sencilla que existe, conformada por una sola neurona, la cual se va modificando mediante el valor del umbral escogido y ponderaciones, para detectar características o tendencias en los datos ingresados [30].
- Redes feedforward o prealimentadas: Este tipo de RN solo puede transmitir información en una sola dirección y no existe ningún ciclo o bucle dentro de ellas. Estas redes pueden ser de una sola capa (de entrada o de salida) o estar conformadas por varias, constituidas por muchas capas ocultas [30].
- Redes neuronales recurrentes (RNN): La particularidad de este tipo de red es que la información puede recorrer ciclos feedback, es decir, volver atrás por medio de las capas. Este feedback o retroalimentación permite que la RN desarrolle memoria, en otras palabras, poder recordar valores previos. Por lo anterior, es que estas redes se utilizan por ejemplo en la traducción, reconocimiento de voz y texto, entre otras áreas [30].
- Redes neuronales convolucionales: Este tipo de RN es una subcategoría de redes conformadas por muchas capas (al menos 5), en donde en cada capa se realizan reconocimientos de patrones que se van transmitiendo de capa en capa. Esto hace que sean muy similares al funcionamiento de las neuronas en la corteza visual de un cerebro humano. Esta red es una modificación a un perceptrón de multicapa, pero ya que la gran mayoría de sus aplicaciones se da en matrices bidimensionales, son muy efectivas en áreas como clasificación de imágenes o visión artificial [30].

2.3.5 Procesamiento del Lenguaje Natural (PLN o NLP)

El procesamiento del lenguaje natural o NLP corresponde a un área de la Inteligencia Artificial, que consiste en la capacidad de una máquina de comunicarse con las personas mediante el uso del lenguaje natural, como lo es el inglés, español o alemán. Su funcionamiento se basa en “...*crear modelos computacionales del lenguaje suficientemente detallados que permitan escribir programas informáticos que den distintas órdenes o*

peticiones donde interviene el lenguaje natural...” [31]; es decir, darles a las computadoras la capacidad de comprender el texto de la misma manera que lo haría un ser humano.

El procesamiento del lenguaje se divide en dos grandes módulos: el nivel interpretativo, el cual considera el nivel semántico y el nivel pragmático; y el nivel gramatical, el cual contiene a los niveles morfológico y sintáctico.

- Nivel morfológico: *“Es un análisis interno de cada una de las palabras que conforman las distintas oraciones, para extraer lemas, unidades léxicas compuestas, rasgos flexivos, entre otros”* [32].
- Nivel sintáctico: *“Es un análisis de la estructura de la oración, según el modelo gramatical escogido”* [32].
- Nivel semántico: *“Permite generar la interpretación de las oraciones una vez eliminadas las ambigüedades morfosintácticas”* [32].
- Nivel pragmático: *“Incorpora el análisis del contexto de uso a la interpretación final”* [32].

El NPL cuenta con una gran cantidad de herramientas para el análisis de texto y voz. Dentro del procesamiento de texto se puede encontrar:

- **Tokenización:** Consiste en separar una frase u oración en palabras o tokens, para el posterior procesamiento de estas.
- **Lematización:** Consiste en relacionar una palabra derivada a su lema o forma canónica. Por ejemplo, el lema de *dijéramos*, *diré* y *dije* es “decir” [33].
- **Stopwords:** Las stopwords son palabras muy habituales que no aportan significado en una oración. Es decir, si estas no estuvieran, de igual forma se seguiría entendiendo una frase. Por esto es que el poder identificarlas y excluirlas, es una manera de “limpiar” el texto y que este sea más fácil de procesar. Un ejemplo puede ser “Me gusta leer, entonces leo” (con stopwords) → “Me gusta, leer, leo” (sin stopwords).
- **FreqDist:** Esta herramienta tiene como función el identificar la frecuencia de los tokens que componen distintas frases.
- Entre otras características asociadas al NPL que permiten el procesamiento tanto de texto como de voz.

A. Herramientas del NLP en Python

Se cuenta con una amplia variedad de bibliotecas y herramientas en *Python* para enfrentar tareas específicas de NPL. A continuación, se pueden mencionar 2 de ellas:

A ..1 *Natural Language Toolkit (NLTK)*

Muchas de estas herramientas se pueden encontrar en *NLTK* [34], el cual es un conjunto de código abierto de recursos educativos, bibliotecas y programas para crear algoritmos de NPL. Dentro de las actividades que se pueden realizar se pueden encontrar: análisis de texto, segmentación de palabras, métodos para recortar palabras hasta sus raíces (lematización y derivación), tokenización, la cual sirve para separar frases, palabras u oraciones en segmentos con el objetivo de que la computadora comprenda mejor el texto, entre otros. *NLTK* es especialmente una biblioteca de procesamiento de cadena o “strings”, donde cada función toma estas cadenas como entradas y luego las devuelve procesadas.

La arquitectura del NLP se guía por “una definición del lenguaje natural por niveles, que son los niveles fonológico, morfológico, sintáctico, semántico y pragmático” (ver Figura 2.6), que apoyan al usuario a darse a entender por la máquina y que esta pueda interpretar la información [35][36][37].



Figura 2.6: Arquitectura que debería seguir el Procesamiento Natural del Lenguaje.

A ..2 *Biblioteca spaCy*

Es otra conocida biblioteca gratuita y de código abierto, para el procesamiento del lenguaje natural (NLP) programada en *Python*. La biblioteca *spaCy* está diseñada para ser una ayuda a la hora de crear aplicaciones que comprendan y procesen grandes cantidades de texto. Puede ser utilizada para construir sistemas o algoritmos de extracción de información, para procesar texto para deep learning o bien comprender el lenguaje natural.

Al igual que *NLTK* cuenta con ciertas características que le permiten adquirir una serie de funcionalidades, como lo son: tokenización, análisis de dependencia, lematización, similitud, entre otras características que tiene esta biblioteca [38].

Cabe destacar, que a diferencia de *NLTK*, *spaCy* toma un enfoque orientado a objetos, en donde cada función utilizada devuelve objetos en lugar de un string o matrices, permitiendo una fácil exploración de la herramienta.

2.3.6 Word Embeddings

Las *Word Embeddings* o *Incrustaciones de palabras*, son técnicas en las que las palabras individuales se representan como vectores de valores reales en un espacio vectorial predefinido. Aquí cada palabra se asigna a un vector y los valores de los vectores se aprenden de tal forma que se asemeja a una red neuronal, y por ende esta técnica recae en el campo del *Deep Learning*. La gracia de las *Word Embeddings* es que hacen uso de una representación distribuida, la cual va aprendiendo en base al uso de las palabras. Lo anterior permite que palabras que se utilizan de manera similar resulten en representaciones vectoriales similares. Como se visualiza en la Figura 2.7, aquellas palabras que tienen alguna relación están más cercanas en el espacio vectorial, como es el caso de los meses o los días de la semana.

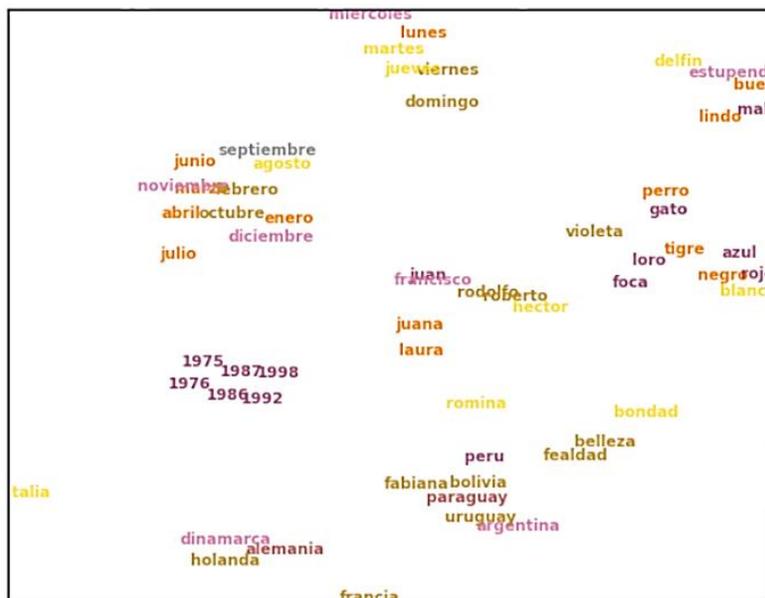


Figura 2.7: Ejemplo de representaciones distribuidas.

Como se visualiza aquellas palabras que tiene alguna relación están más cercanas en el espacio vectorial, como es el caso de los años o los países [80].

A. Algoritmos de incrustaciones de palabras

En la actualidad existen variados métodos de *Word Embeddings*, los cuales aprenden una representación vectorial de valores reales para un vocabulario predefinido de tamaño fijo a partir de un corpus de texto.

A continuación, se mencionan algunas técnicas que se utilizan para aprender una palabra incrustada a partir de datos de texto:

- **Word2Vec:** Es un modelo creado en el 2013 por Tomas Mikolov en Google. Este modelo se basa en redes neuronales de varias capas y tiene dos posibles arquitecturas: CBOW (modelo de bolsas de palabras) y Skip-gram, las cuales requieren de conjuntos de datos muy grandes para el cálculo de las incrustaciones. CBOW aprende la incrustación prediciendo la palabra actual basándose en su contexto, mientras que Skip-gram predice las palabras circundantes a partir de una palabra central dada. Ambas arquitecturas se enfocan en el aprendizaje de palabras a partir del contexto de uso local, en donde el contexto depende de una ventana de palabras vecinas. Esta ventana es uno de los parámetros configurables en el modelo. Podemos mencionar, que este modelo desde su desarrollo se ha vuelto un estándar para la generación de incrustaciones pre-entrenadas [39].
- **GloVe:** El modelo de Vectores Globales para la Representación de Palabras o *GloVe*, es una extensión de *Word2Vec*, desarrollado por Pennington, en Standford el año 2014. La característica de este modelo es que utiliza técnicas de factorización matricial, tales como el Análisis Semántico Latente (LSA) para la representación de los embeddings. Este modelo utiliza un enfoque que combina estadísticas globales del LSA con el aprendizaje local basado en el contexto de *Word2Vec*. La diferencia con *Word2Vec* es que en vez de hacer uso de ventanas para definir el contexto, ... “*construye una matriz explícita de palabra-contexto o co-ocurrencia de palabras, utilizando estadísticas a través de todo el corpus del texto*”... [39]. En palabras simples, *GloVe* cuenta con qué frecuencia aparece una palabra en cierto contexto. Esta diferencia permite que *GloVe* obtenga mejores embeddings [39].
- **FastText:** Es un modelo utilizado para el aprendizaje de incrustaciones de palabras y clasificación de texto creada por el laboratorio AI Research (FAIR) de Facebook en el año 2015. A diferencia de *Word2Vec* y *GloVe*, que tratan cada palabra como la unidad

más pequeña para entrenar, *FastText* hace uso de los n-gramas como la unidad mínima para representar el término. Esto permite generar mejores incrustaciones para palabras raras, mal escritas o incluso que no se consideraron a la hora del entrenamiento del modelo, ya que los vectores de caracteres n-grama se comparten con otras palabras. Esto es algo que no pueden lograr ninguno de los modelos anteriormente mencionados. Entonces, por ejemplo, si consideramos la palabra “artificial” con un $n=3$ (3-grama o tri-grama), la representación de *FastText* para la palabra indicada sería $\langle ar, art, rti, tif, ifi, fic, ici, cia, ial, al \rangle$, en donde los símbolos “ $\langle \rangle$ ” indican el inicio y fin de la palabra [40].

Como se pudo evidenciar, hoy en día hay una variada gama de modelos que permiten la generación de *Word Embeddings*. Estos se pueden desarrollar desde 0, donde se pueden escoger los parámetros, construir los embedding en base a conjuntos de datos escogidos (nuevos corpus) y entrenar los modelos. O bien se pueden utilizar modelos ya pre-entrenados en conjuntos de datos gigantescos, los cuales ya tiene representaciones vectoriales muy grandes. En la Figura 2.8 se puede visualizar un ejemplo de similitud de palabras utilizando el modelo *Word2Vec* pre-entrenado. Se pueden apreciar las 5 palabras más similares a las palabras indicadas en la leyenda de la figura, pudiendo observar que aquellas palabras similares mantienen una distancia semántica menor.

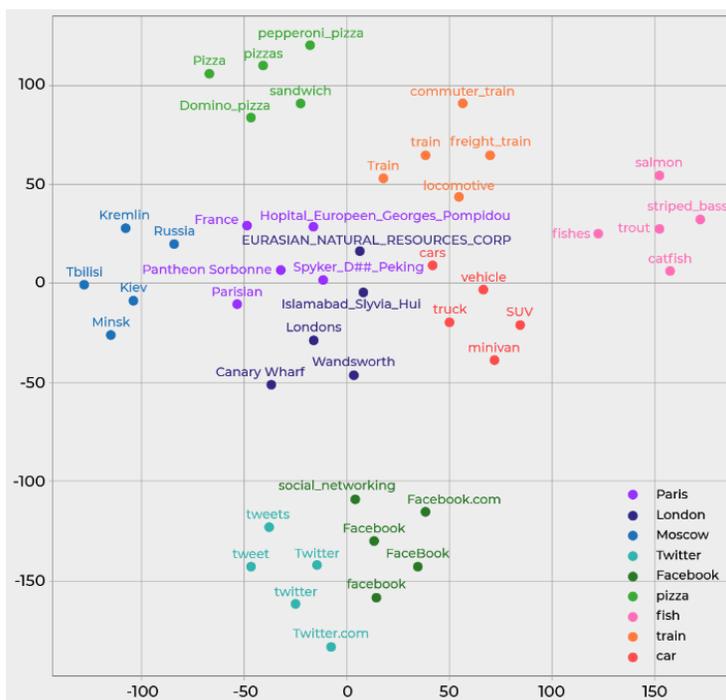


Figura 2.8: Similitud de palabras según Word2Vec en un espacio 2D [81]

2.3.7 Transformers o Transformadores

Los *Transformers* son una red neuronal que aprende el contexto y, por ende, el significado mediante el seguimiento de las relaciones en datos secuenciales como las palabras de cierta oración. Estos modelos utilizan un conjunto de técnicas matemáticas, llamadas *atención o auto-atención (attention or self-attention)*, para poder detectar formas sutiles de elementos que se influyen y dependen unos de otros [41].

Luego de ser descritos por primera vez el año 2017, los *Transformers* se encuentran dentro de los modelos más nuevos y poderosos inventados hasta la fecha, los cuales han impulsado una gran cantidad de avances en el aprendizaje automático. Hoy en día estos modelos tienen una gran área de aplicación, como la detección de tendencias y anomalías para evitar fraudes, traducción de texto y voz, análisis de sentimientos, reconocimiento de objetos, sistemas de preguntas-respuestas, extracción de información, similitud semántica, entre otros. [42].

Cualquier aplicación que utilice datos secuenciales de texto, imagen o video, es apta para los modelos transformadores.

Antes de su llegada, los usuarios debían hacer uso de redes neuronales con grandes conjuntos de datos etiquetados que requerían mucho tiempo para producir resultados, además de sus altos costos. En cambio, los *Transformers* al encontrar matemáticamente patrones entre elementos, eliminan la necesidad de tener los datos etiquetados, y esto permite que puedan ser entrenados con petabytes de datos de texto y billones de imágenes de la web, bases de datos corporativas, entre otros datos de entrenamiento [43].

Cabe destacar, que las matemáticas utilizadas por este modelo se realizan en procesamiento paralelo, lo que permite que se ejecuten rápidamente.

Como la gran mayoría de redes neuronales, estos modelos son en palabras simples grandes bloques codificadores/decodificadores que procesan los datos ingresados. La diferencia es que los *Transformers* incorporan adiciones pequeñas pero estratégicas a estos bloques (se pueden observar en Figura 2.9 en color verde (*Multi-headed Attention*) y rojo (*Self-Attention*)), lo que hace que sean mucho más poderosos que las redes neuronales. El objetivo de utilizar codificadores en los *Transformers* es el poder etiquetar elementos de datos que entran y salen de la red [44].

Las unidades de *atención* o *autoatención* siguen estas etiquetas, y calculan una especie de mapa algebraico de cómo cada uno de los elementos que conforman la secuencia se relacionan con los demás. Las consultas realizadas en las unidades de atención generalmente se van realizando en paralelo mediante el cálculo de una matriz de ecuaciones denominada “*Atención de Múltiples cabezas*” o “*Multi-headed Attention*” [43][44].

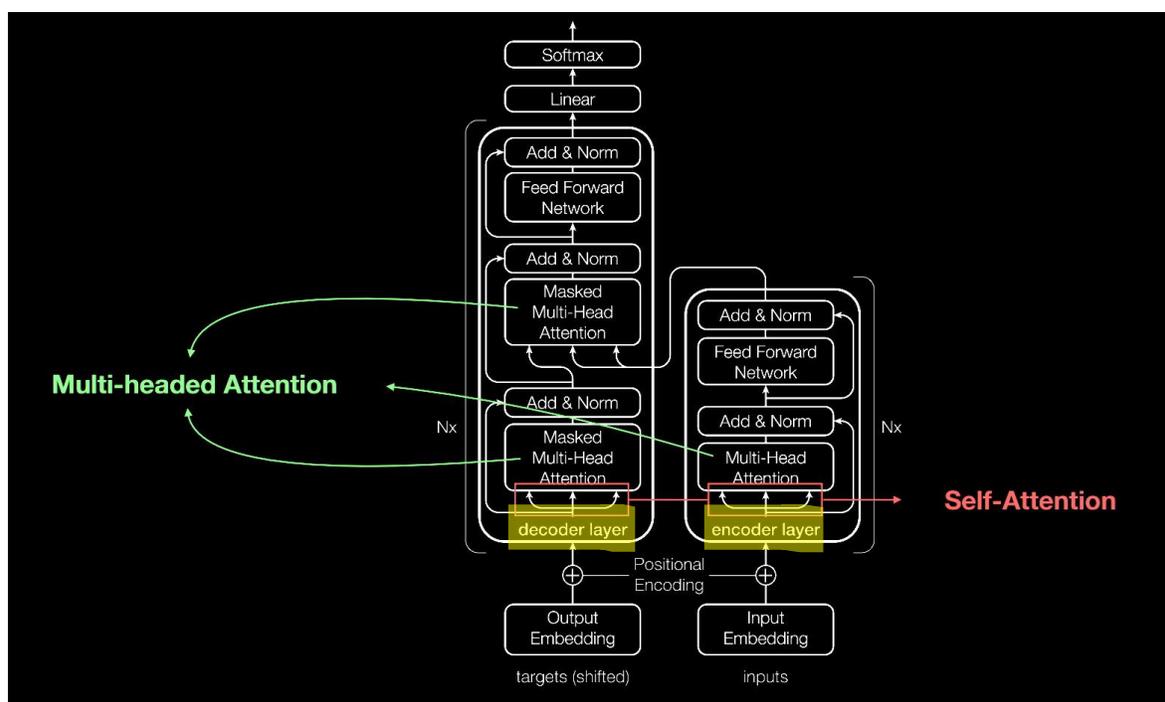


Figura 2.9: Arquitectura de un Transformador

En la parte derecha de la imagen se visualiza el codificador (encoder) y en la parte izquierda el decodificador (decoder) que conforman el transformador [43].

Los bloques de codificador y decodificador en realidad están conformados por múltiples codificadores y decodificadores idénticos apilados uno encima del otro, y la cantidad de unidades es un hiperparámetro que se puede ir variando en el modelo.

Por ejemplo, al utilizar los *Transformers* para una tarea de traducción de un idioma a otro, como por ejemplo de alemán a inglés, los pasos a seguir serían los siguientes:

- Las incrustaciones de las palabras de la secuencia de entrada se pasan al primer codificador, estas luego se transforman y se propagan al siguiente, y así sucesivamente hasta llegar al último codificador.

- Luego, la salida del último codificador pasa a todos los decodificadores que conforman la pila de decodificadores, como se puede visualizar en la Figura 2.10 [41].

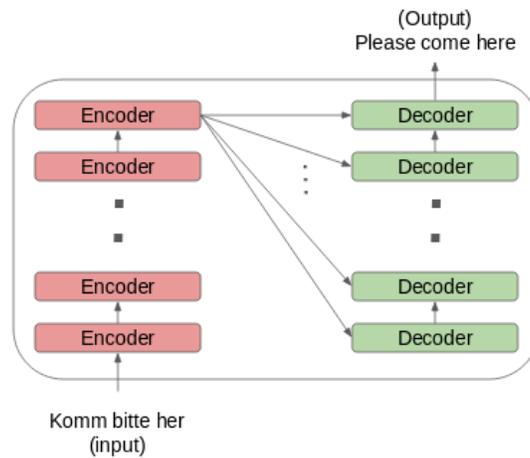


Figura 2.10 Funcionamiento del sistema codificador-decodificador en los Transformers

Se puede visualizar la arquitectura encoder-decoder de los transformers en la traducción de una frase en alemán (“Komm bitte her”) al inglés (“Please come here”) [41]

Como se mencionó anteriormente la autoatención se computa no una única vez, sino que varias veces en la arquitectura de los transformadores, en paralelo y de forma independiente. Por lo tanto, se denomina “Multi-Head Attention” o “Atención de Múltiples cabezas” como se ve en la Figura 2.9. Por lo anterior, es que las múltiples salidas se deben concatenar y transformar linealmente para poder obtener así la respuesta del modelo [43].

Finalmente, el modelo al vector resultante le aplica una *función Softmax*, la cual “comprime” un vector N-dimensional de valores reales arbitrarios en un vector N-dimensional de valores reales en el rango [0,1] [41].

Con estas herramientas mencionadas anteriormente los transformadores permiten que las computadoras puedan visualizar los mismos patrones que vemos los humanos a la hora de relacionar distintas secuencias

2.3.8 BERT (Bidirectional Encoder Representations from

Transformers)

Bidirectional Encoder Representations from Transformers o *BERT* es un sistema basado en *IA*, que fue desarrollado con el objetivo de ayudar a los algoritmos de Google Search a comprender de mejor manera el lenguaje utilizado por los usuarios a la hora de realizar búsquedas mediante oraciones. Este hace uso de una red neuronal de código abierto para procesar el lenguaje natural de lo que se busca en Google [45].

BERT logra su objetivo debido a que goza de una característica llamada “bidireccionalidad”, la que le permite analizar una frase u oración en dos direcciones. Es decir, analiza las palabras que se encuentran a la derecha como a la izquierda de una palabra clave, permitiendo comprender el contexto y la temática de toda la oración ingresada por el usuario en la búsqueda de Google. Un ejemplo de lo anterior es lo que se visualiza en la Figura 2.11, donde la palabra “cura” se encuentra en ambas oraciones, pero *BERT* al utilizar la bidireccionalidad permite que el modelo pueda identificar que la palabra subrayada no significa lo mismo en ambas frases [46].



Figura 2.11 Ejemplo de la bidireccionalidad de BERT para comprender el contexto de una oración [46]

La característica de bidireccionalidad que contempla este sistema es muy importante a la hora de comprender el real significado de un idioma, ya que, logra entender en profundidad el contexto y la temática de cada oración. Además, este modelo está pre-entrenado en un gran corpus de texto sin etiquetar que incluye a *Book Corpus* (cuenta con cerca de 800 millones de palabras), *Wikipedia* completa (incluye cerca de 2500 millones de palabras) y todas las búsquedas y archivos asociados a *Google*. Lo anterior hace que este modelo en su versión pre-entrenada sea capaz de dar respuesta a una gran gama de tareas del NLP [47].

Otra característica con la que cuenta *BERT* es que se basa en los *Transformers* o *Transformadores*, lo que le permite entender palabras como nexos, preposiciones y pronombres, las cuales tienen un papel fundamental a la hora de interpretar el sentido de una frase [48].

Lo anterior implica que el sistema no se enfoca solo en las palabras claves que conforman las distintas oraciones (como sí lo hacen otros modelos), debido a que dejan de tener importancia al verlos como elementos individuales. Un ejemplo claro podría ser el indicado en la Figura 2.11, donde si solo se considerara la palabra “cura”, cualquier otro modelo como el *Word2Vec* podría considerar que ambas frases al contener la misma palabra quizás sean similares, dado que este tipo de modelos solo consideran las palabras como tokens individuales y no toman en cuenta el contexto de la frase [47].

BERT al ser código abierto, permite que cualquier usuario lo pueda utilizar. Esto implica que el modelo pueda ser especializado para ciertas tareas al entrenarlo previamente con ciertas representaciones contextuales [49]. Algunos ejemplos de *BERT* especializado en distintas áreas son los siguientes:

- **docBERT**: es un modelo *BERT* especializada en la clasificación de documentos [49].
- **bioBERT**: es un modelo de representación de lenguaje biomédico previamente entrenado para la minería de textos biomédicos [49].
- **G-BERT**: es un modelo *BERT* previamente entrenado usando códigos médicos con representaciones jerárquicas usando redes neuronales gráficas (GNN) y posteriormente ajustado para hacer recomendaciones de medicamentos [49].
- **SBERT**: es un modelo *BERT* especializado en el cálculo de representaciones vectoriales densas para oraciones, párrafos e imágenes [50].

Para efectos de este proyecto en el siguiente subcapítulo se hablará del modelo *SBERT* o *SentenceTransformers*.

A. **SBERT o SentenceTransformers**

Como se mencionó anteriormente, *SBERT* es una especialización del modelo *BERT* que permite calcular representaciones vectoriales densas para oraciones, párrafos e imágenes. En palabras simples, lo que hace este modelo es incrustar el texto o una imagen en el espacio vectorial, de tal forma que el texto/imagen o frases similares se encuentran cerca y las que difieren entre sí se encuentran lejos. De este modo se pueden identificar de forma fácil y

eficiente las similitudes semánticas, imágenes similares, así como implementar búsqueda de estas, realizar una eficiente minería de paráfrasis (textos con significado idéntico/similar), entre otras aplicaciones.

Además, otra característica que tiene *SBERT* es que cuenta con una gran cantidad de modelos pre-entrenados en más de 100 idiomas, los cuales se ajustan a variados casos de uso. Asimismo, permite un ajuste de los diferentes modelos de incrustaciones con los que cuenta, lo que permite lograr un máximo rendimiento en tareas de uso específico.

Dentro de los modelos pre-entrenados que podemos encontrar en *SentenceTransformers* se pueden encontrar los “Modelos Multilingües”, los que generan espacios vectoriales alineados. En otras palabras, frases similares en distintos idiomas se encuentran próximos en el espacio vectorial.

Lo anteriormente mencionado es una ventaja a la hora de trabajar con proyectos en un idioma que no sea el inglés, debido a que la gran mayoría de modelos pre-entrenados como los de *Word2Vec*, *FastText*, *Glove*, o *BERT multilingüe (mBERT)* constan de un preentrenamiento el cual se genera en la lengua materna de sus desarrolladores. Luego, el poder utilizar estos modelos en un lenguaje distinto es un gran desafío, debido a que no se genera un alineamiento entre los distintos idiomas o bien, en el caso de los *Word Embeddings*, no se generan buenas representaciones a la hora de evaluar frases en vez de palabras individuales [50].

2.4. Chatbot

En palabras simples un Chatbot es un programa informático basado en inteligencia artificial que simula y procesa conversaciones con personas, generando respuestas automáticas a las preguntas generadas por los usuarios a través de dispositivos digitales. El objetivo de los chatbot es que los humanos puedan interactuar con este software como si estuvieran interactuando con otra persona. Desde sus comienzos los chatbot pasaron de ser programas básicos que se encargaban de responder preguntas simples con respuestas cortas, a programas sumamente sofisticados, como son en la actualidad los asistentes digitales como Siri, Alexa o Cortana [51].

2.4.1 Avances a través del Tiempo

En 1966 Joseph Weizenbaum desarrolló *ELIZA*, el primer Chatbot, el cual procesaba las palabras clave recibidas como entrada y luego desencadenaba las respuestas de salida. Pero, a pesar de que los chatbot existen desde hace décadas, no fue hasta hace algunos años que estos tomaron popularidad. Lo anterior se puede visualizar en Figura 2.12, donde se observa la tendencia incremental en la búsqueda web del término “chatbot” [52] [53].

La respuesta a su creciente uso se puede encontrar en que no es un software complejo, y es una herramienta productiva a la hora de realizar distintas tareas. Al momento de su elaboración se debe tener en cuenta que será utilizado por distintos tipos de usuarios, y el único requisito para poder utilizarlo, a diferencia de otros programas, es saber hablar con otra persona. Además, los chatbot son fácilmente accesibles, donde el usuario solo hace preguntas a través de sus dispositivos digitales. Permiten que las personas sean eficientes y productivas debido a que desde cualquier lugar se puede hacer uso de ellos, están disponible 24 horas al día y 7 días a la semana y permite dar respuestas instantáneas a las diferentes consultas de los usuarios. Como se puede observar en la Figura 2.13, las industrias que más se benefician con este tipo de herramientas son e-commerce o comercio electrónico, insurance o aseguradoras y healthcare o la industria de la salud [35].

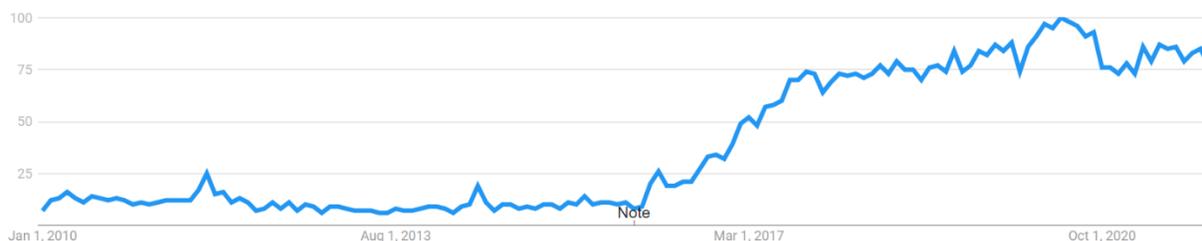


Figura 2.12: Tendencia en la búsqueda y uso del término “Chatbot” entre los años 2010-2021 según Google Trends.

En la imagen el eje de las ordenadas representa el interés de búsqueda en relación con el punto más alto del gráfico para la región (todo el mundo) y el tiempo seleccionado (2010-2021). Un valor de 100 indica el peak de popularidad del término, mientras que un valor de 50 indica que el termino es la mitad de popular [52].

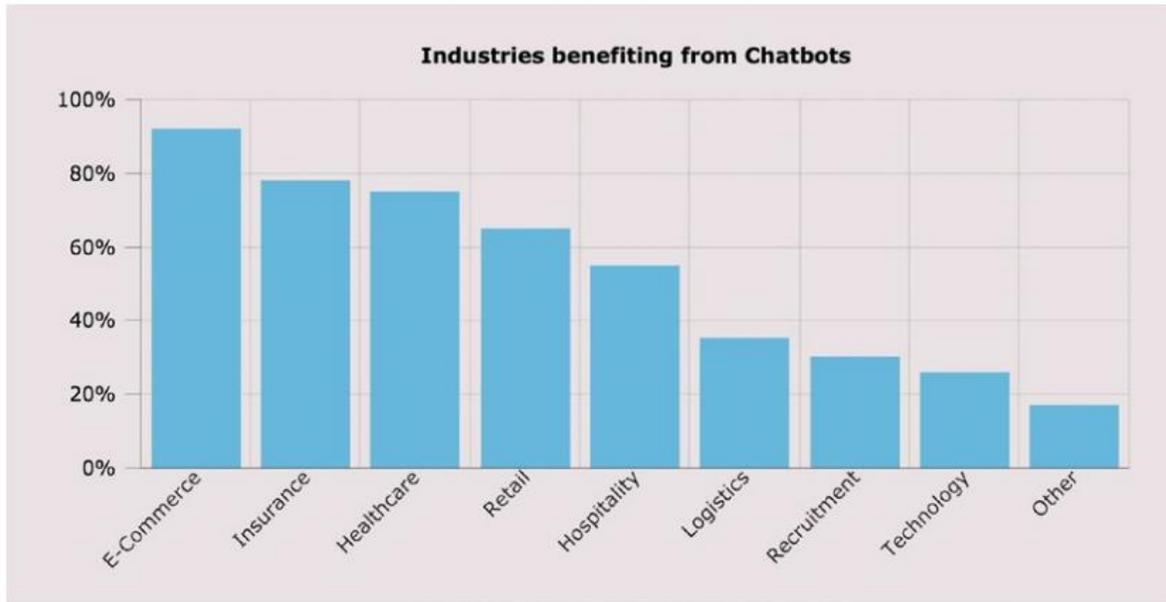


Figura 2.13: Sectores que más se beneficiarán de los chatbot [35]

2.4.2 Tipos de Tecnologías Chatbot

Si se consideran todos los chatbot que en la actualidad existen, la gran mayoría de estos se encuentra dentro de dos grandes tipos: los basados en modelos lingüísticos, es decir, basados en reglas, y los de aprendizaje automático, es decir, que utilizan IA.

A. *Basados en Modelos Lingüísticos:*

Con este tipo de chatbot se puede decidir de antemano cuál será la respuesta correcta a cierta duda o pregunta, además se pueden diseñar pruebas para poder comprobar la consistencia y calidad del software. Su funcionamiento se basa en reglas “if-else” o “si/entonces” para generar la conversación. Para asegurar que las preguntas que quieren decir lo mismo, tengan la misma respuesta, utiliza condiciones del lenguaje con el objetivo de examinar las palabras que componen el mensaje, conocer sinónimos, formas comunes de formular una misma pregunta, entre otros. Dentro de sus desventajas es que estos chatbot pueden ser lentos de desarrollar y rígidos, pero a pesar de esto son los más comunes. Se pueden encontrar en sitios web de comercio electrónico o chat en vivo en Facebook Messenger [54].

B. Basados en Aprendizaje Automático:

Este tipo de chatbot son más complejos y más sofisticados que los basados en modelos lingüísticos, tienden a ser más fluidos, utilizar más datos y ser más predictivos. Son más personalizados e interactivos, y gracias al tiempo de implementación del software y a los datos adquiridos, son más conscientes del contexto de la conversación, debido a que aprenden de patrones y experiencias previas. Además, aprovechan la inteligencia predictiva y la comprensión del lenguaje natural para personalizar la interacción con el usuario. Dentro de las desventajas que se pueden encontrar es que esta herramienta necesita de grandes cantidades de datos de aprendizaje, y a diferencia del anterior, es más complicado encontrar errores en el modelo, optimizarlo o bien mejorarlo [54].

C. Modelo Híbrido:

El modelo híbrido nos brinda la mejor parte de los casos anteriores, permitiendo el desarrollo de chatbot conversacionales con IA mucho más complejos. Este enfoque permite que las respuestas o soluciones se construyan aún sin datos, permite que la integración del aprendizaje automático vaya más allá de utilizar reglas lingüísticas, con el objetivo de permitir inferencias en áreas donde usar el enfoque lingüístico sea imposible. Asimismo, el generar aplicaciones usando exclusivamente aprendizaje automático o lingüístico, requiere de muchos recursos y es muy costoso [54].

2.4.3 Otras herramientas que facilitan el desarrollo de Chatbot

A. Chatterbot

Es una biblioteca de *Python* diseñada para facilitar la creación de un software que pueda entablar una conversación, es decir, el desarrollo de un chatbot. Utiliza una variedad de algoritmos de aprendizaje automático para generar los distintos tipos de respuestas. Esta herramienta logra facilitar el trabajo en el desarrollo de chatbot y la automatización de conversaciones con los usuarios.

Debido a que *Chatterbot* fue diseñado independiente del idioma, permite que se pueda entrenar en cualquier lengua. Además, puesto que cuenta con variados algoritmos de aprendizaje automático, permite que el agente desarrollado mejore su conocimiento sobre las

posibles respuestas a medida que va interactuando con distintos usuarios y otras fuentes de datos.

En un comienzo esta herramienta inicia como una instancia no capacitada, es decir, sin saber cómo comunicarse. A medida que los usuarios ingresan una pregunta o una declaración, la biblioteca guarda ese texto ingresado y la respectiva respuesta a tal declaración. Mientras más entradas reciba *Chatterbot*, aumenta la cantidad de preguntas que puede responder y la precisión de cada respuesta en relación a lo declarado [55].

El proceso en el que se basa esta biblioteca se puede visualizar en Figura 2.14.

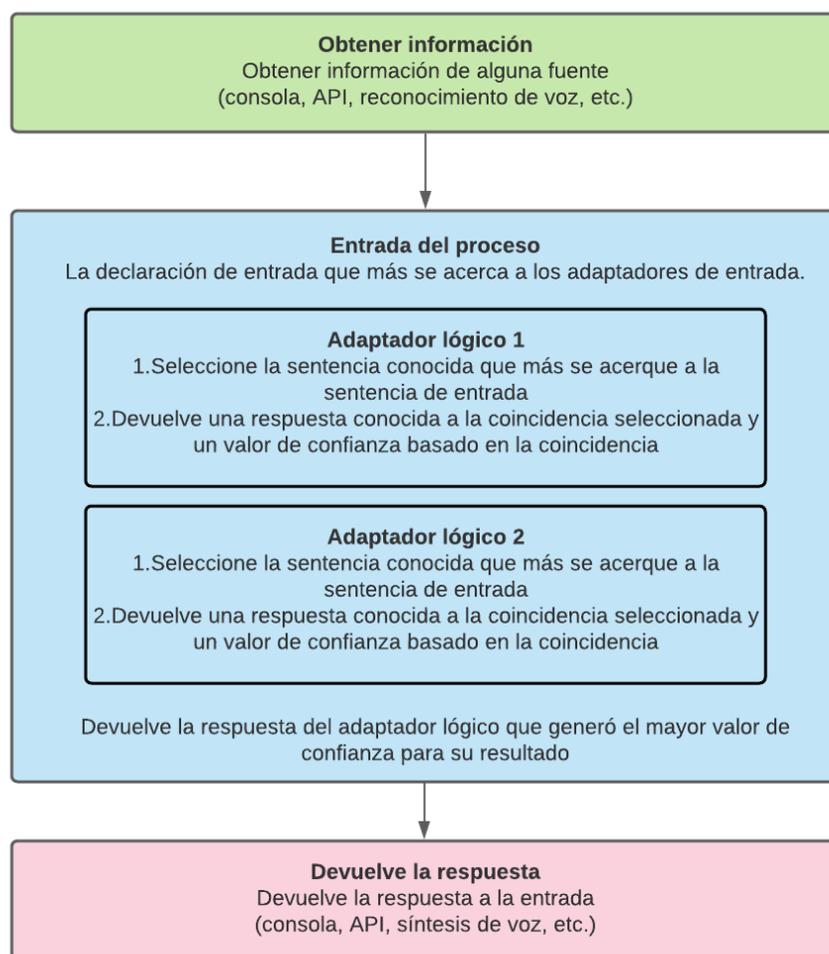


Figura 2.14 Diagrama de flujo del proceso utilizado por Chatterbot

Cabe destacar que existe una comunidad de desarrolladores que aportan en *Chatterbot*, que ha generado una serie de base de datos de preguntas y respuestas sobre

distintos temas como los relacionados a psicología, IA, emociones, saludos, entre otros, en una variedad de idiomas, permitiendo así que esta herramienta pueda contar con una base al momento de iniciar. Estos archivos creados, pueden descargarse de forma gratuita directamente de la página oficial de *Chatterbot* en *GitHub*, permitiendo aumentar los horizontes de esta biblioteca, o bien se pueden crear desde cero con las preguntas y respuestas que se deseen.

B. Landbot

Landbot es una de tantas páginas web que permiten desarrollar chatbot sin necesidad de un código o en otras palabras un “No-Code Chatbot Builder” [56].

Landbot es capaz de construir bots que se pueden añadir a otras páginas web o aplicaciones móviles, solo indicando paso a paso lo que se quiere realizar en la interfaz del desarrollador. Evita así tener problemas asociados a códigos o bien tener que aprender a manejar las distintas bibliotecas necesarias para el desarrollo de un chatbot. Cabe destacar que la interfaz del desarrollador es bastante intuitiva a la hora de trabajar en ella.

Esta plataforma cuenta con variadas suscripciones entre las que se puede encontrar una versión gratuita en la que se cuenta con características y elementos básicos para el desarrollo de un chatbot. Además, en el caso de los usuarios que están empezando en el desarrollo de bots, cuenta con un instructivo claro para comenzar a trabajar en la página, el cual permite crear chatbot básicos posterior a la introducción generada. Otro de los beneficios de esta herramienta es que se puede integrar con otras APIs, páginas, aplicaciones, entre otras.

Entre las observaciones negativas que se encuentran es que, si bien es posible desarrollar el chatbot en español, la herramienta en si solo está disponible en inglés, es decir, si no se cuenta con el dominio de esta lengua se dificulta el aprender a utilizarla y el desarrollar el proyecto que se desee [56].

2.5. Discusión

La revisión bibliográfica realizada a distintas fuentes ha dado a conocer que los chatbot son una herramienta sumamente poderosa y que ha sido bien recibida en distintos ámbitos. Un ejemplo es la industria e-commerce, la cual al no contar con personal que asesore

al cliente debido a que son tiendas web, permite que la atención al usuario sea directamente desde la página, 24/7 y que el proceso sea más automatizado.

Además, esta inteligencia puede tener conversaciones en lenguaje natural con una persona, pudiendo aplicarse en una variedad de páginas web, aplicaciones de mensajería o aplicaciones móviles. Los chatbot al trabajar con machine learning e inteligencia artificial más allá de responder con datos almacenados en su base de datos, pueden dar respuestas coherentes, breves y directas con las que se obtengan mejores resultados en la conversación, interpretar al usuario y son capaces de aprender de cada interacción o conversación con la persona.

Además, en la actualidad existe una variedad de software, bibliotecas, modelos y páginas web que tienen como objetivo ayudar en el desarrollo de los chatbot, y día a día estas herramientas van a ir en aumento debido al boom y a la demanda creciente de este tipo de algoritmos.

El aplicar este tipo de tecnologías en áreas como el *GES 4*, permite fortalecer la atención paliativa, siendo un apoyo para el equipo de profesionales encargados de los pacientes, en tareas como: responder preguntas frecuentes, mantener una comunicación 24/7 entre la familia y el centro de salud respectivo, cubrir la falta de personal especializado en sectores extremos, acortar los tiempos de acción frente a alguna emergencia, entre otros. Por lo anteriormente mencionado, se aprecia la utilidad y la aplicabilidad de los chatbot en el área de la salud y gestión de pacientes.

Capítulo 3. Estado del Arte

A continuación, se presentan trabajos similares realizados sobre el desarrollo de chatbot, manuales asociados a los cuidados paliativos y herramientas de apoyo al cuidador.

3.1.1 Desarrollo de Chatbot

“*Development of a chatbot for informing students of the schedule*”, escrito por Andrii O. Priadko, Kateryna P. Osadcha, Vladyslav S. Kruhlyk y Volodymyr A. Rakovych el 2019 [57]. Describe el proceso de desarrollo de un chatbot para proporcionar información a los estudiantes sobre sus horarios, utilizando la aplicación móvil *Telegram*. En el trabajo se da a conocer los métodos utilizados para la selección de la aplicación móvil, la obtención y el manejo de los datos y la implementación del mecanismo de presentación de la información en el mensajero. También informa la manera en la que se genera la comunicación entre *Python* y la API de *Telegram* para la implementación del programa.

El libro “*Building Chatbots with Python: Using Natural Language Processing and Machine Learning*”, de Sumit Raj escrito el 2019 [35]. Da a conocer una variedad de alternativas para diseñar y desarrollar un Chatbot utilizando *Python* y herramientas de código abierto. Además, genera una recopilación de toda la información necesaria sobre la arquitectura del software y el uso de la biblioteca *NLTK* para la construcción de una plataforma de procesamiento del lenguaje personalizada para un chatbot. Finalmente, el libro enseña cómo desarrollar, entrenar y desplegar un chatbot de creación propia.

“*Chatbot for healthcare system using Artificial Intelligence*”, es un trabajo realizado el año 2019 por Kavitha B. R y Dr. Chethana R. Murthy [58]. Tiene como objetivo dar a conocer la creación de un chatbot enfocado en salud. Este hace uso de la *IA*, en base al cálculo de palabras claves, ranking y similitud de frases utilizando n-gram, Term frequency-Inverse documet frequency (TF-IDF) y la similitud del coseno para poder generar las respuestas a las consultas generadas por el usuario a través de una aplicación móvil.

“*Python-Bot: A Chatbot for Teaching Python Programming*”, es un trabajo desarrollado por Okonkwo Chinedu y Abejide Ade-Ibijola en el año 2021 [59]. Propone un Chatbot que ayuda a los programadores novatos a entender las estructuras sintácticas básicas de *Python* y su semántica. Además, da a conocer otra estrategia para la elaboración de los

Chatbot la cual es a través de una plataforma pagada encargada de crear chatbot inteligentes para mensajería multicanal.

“*A Chatbot for Medical Purpose using Deep Learning*”, de Himanshu Gadge, Vaibhav Tode, Sudarshan Madane, Prateek Kachare y Prof. Anuradha Deokar [60]. Fue elaborado el año 2021 con el objetivo de proporcionar una herramienta en base a la IA para el diagnóstico médico utilizando *Deep Learning* en las zonas rurales de la India, así como en las personas pobres y necesitadas de este país. Lo que se destaca de este proyecto es que el sistema tiene la capacidad de entender los síntomas del paciente y comunicarse con él a través de una web, haciendo uso del módulo *NLTK* en *Python* para el procesamiento del lenguaje natural.

3.1.2 Manuales Asociados a Cuidados Paliativos

El “*Manual de Cuidados Domiciliarios para Paciente Oncológico y Familia*”, en colaboración con el Programa Alivio del Dolor y Cuidados Paliativos del Hospital Regional de Talca (HRT) y la Universidad Católica del Maule a través de su Plan de Mejoramiento Institucional (PMI) en Oncología [61]. Funciona como una herramienta de apoyo tanto para los pacientes como para sus familias durante el transcurso de la enfermedad desde el hogar, guiándolos según las necesidades psicológicas o físicas que se presenten en el transcurso del cuidado.

“*TIC para cuidadores informales de pacientes paliativos*”, es un trabajo desarrollado el año 2017 por María de la Cabeza y María José Calero [62]. Tiene como objetivo principal el desarrollo de un manual enfocado en cuidadores informales de pacientes con cuidados paliativos, refiriéndose a aquel que suele ser un familiar del paciente y que no recibe remuneración por su trabajo. Lo que destaca de este trabajo es que está desarrollado en un lenguaje no tan técnico, de forma que el cuidador pueda entender de buena manera la información entregada en este escrito. El manual se enfoca principalmente en el control de algunos síntomas presentes en los pacientes terminales paliativos y qué puede hacer ante estas situaciones el cuidador, además de entregar medidas de autocuidado para el lector.

Finalmente, el “*Manual de Cuidados Paliativos para Cuidadores*”, desarrollado por el Instituto Nacional del Cáncer, la Secretaria de Gobierno de Salud y el Ministerio de Salud y Desarrollo Social Presidencial de la Nación de Argentina [36]. Está pensado para fines

educativos e informativos, y está centrado en otorgar recomendaciones y consejos para mejorar la confianza y las capacidades técnicas de los cuidadores informales que se encargan del cuidado de un paciente paliativo.

3.1.3 Herramientas de apoyo al cuidador

En la actualidad existen una gran variedad de cursos, talleres y capacitaciones enfocadas en ser un apoyo para los cuidadores informales de adultos mayores, pero muy pocos relacionados a los cuidadores informales de pacientes paliativos oncológicos o pacientes paliativos en general. Dentro de los encontrados podemos mencionar el “*Curso para cuidadores de personas en situación de dependencia moderada y severa*”, impartido por la Escuela de Enfermería de la Universidad de Valparaíso [64]. Este tiene como finalidad principal entregar una variedad de contenidos relacionados al cuidado de pacientes dependientes en el contexto doméstico y permitir que el cuidador tenga acceso a información relacionada a su autocuidado. Algunos de los tópicos que se enseñan en el curso son tópicos relacionados al movimiento de personas postradas, actividades de higiene, prevención de escaras, entre otros. Otro ejemplo es el “*Programa Cuidador Primario*” impartido por la Pontificia Universidad Católica de Chile [65]. Este se enfoca en todas aquellas personas que quieren mejorar sus habilidades formativas relacionadas al cuidador primario a través de una serie de cursos realizados a través de módulos y aplicación de la teoría en centros de simulación, logrando de esta manera que el cuidador entregue cuidados integrales de calidad y permita su inserción laboral.

Al realizar una búsqueda web sobre aplicaciones o distintas herramientas de apoyo para los cuidadores, se encontraron las siguientes:

La aplicación móvil “*Cuidar y cuidarse*”, es un proyecto desarrollado por la Fundación General Universidad de Alcalá [66], enfocada en dispositivos Android. Permite consultar sobre los cuidados a personas que tienen lesiones medulares, que tiene como público objetivo aquellos cuidadores que ejercen sus labores en domicilios particulares o bien en centros de salud, permitiendo solucionar dudas relacionadas al cuidado a través de material audiovisual entregado en la app móvil.

“*acude: Aplicación móvil para cuidadores de adultos mayores con demencia*”, es una tesis desarrollada por María José Gidi Lantadilla el año 2017 en la Facultad de Diseño

de la Pontificia Universidad Católica de Chile [67]. Busca desarrollar una plataforma digital colaborativa e informativa para cuidadores familiares de adultos mayores con demencia, con el objetivo de disminuir el desconocimiento de los cuidadores en torno a la enfermedad y a los cuidados asociados, y a su vez formar una red de apoyo entre las personas que desempeñan esta tarea.

El Instituto Mexicano del Seguro Social (IMSS) desarrolló un chatbot para padres de pacientes pediátricos oncológicos, con el fin de agilizar la comunicación con los familiares de los enfermos. En este pueden realizar consultas, solicitudes y mantener contacto con los hospitales e instituciones de salud. Dentro de los temas que abarca este chatbot se encuentra el abastecimiento de medicamentos, licencias, banco de sangre, enfermedad médica, entre otros contenidos [68].

Cabe destacar que no se encontraron aplicaciones, cursos u otras herramientas con un enfoque específico en ser un instrumento de ayuda y apoyo para los cuidadores informales de pacientes oncológicos que pudiera ser una guía para el desarrollo de este trabajo.

3.2. Discusión

La mayoría de las investigaciones revisadas sobre chatbot con aplicaciones en salud, son asociadas al área de atención al cliente, diagnósticos médicos y funciones administrativas. Si bien se encontraron aplicaciones en educación, ninguna tiene como enfoque principal el ser una ayuda para el cuidador informal de pacientes oncológicos paliativos, en términos de dar respuestas sobre dudas relacionadas a los cuidados en sí, tanto para el paciente como para el cuidador.

Además, existe una gran variedad de tipos de chatbot y bibliotecas que se pueden utilizar para el desarrollo de este trabajo, y la elección de cuál se utilizará, se abordará en profundidad en el siguiente capítulo.

Finalmente, se puede mencionar que la existencia de manuales para cuidados paliativos o instrumentos de apoyo al cuidador informal, son fundamentales para la base de este tipo de cuidados, ya sea, si son dirigidos a profesionales de la salud o bien a cuidadores informales. Estos permiten consultar dudas más globales, y facilitar que los profesionales encargados de los centros de salud oncológicos se enfoquen en su trabajo y agilizar procedimientos relacionados a lo administrativo, entre otros.

Capítulo 4. Diseño y Desarrollo del Software

4.1. Introducción

En este capítulo se presenta y se dan a conocer los requisitos funcionales y no funcionales, el diseño y el desarrollo del chatbot elaborado en el presente proyecto.

4.2. Diseño del chatbot

Para el diseño de la conversación generada por el chatbot se consideró el uso de respuestas guiadas. Lo anterior de forma más clara, se refiere a que, de acuerdo a la pregunta ingresada, el chatbot le enviará un mensaje de vuelta al usuario con las 3 preguntas de la base de datos más similares a lo consultado (ordenadas por orden de similaridad). Además, se incluye una cuarta opción que permite comunicarse directamente con una persona del centro de salud, en caso de que el paciente requiera el asesoramiento de un profesional. Lo anteriormente indicado se visualiza en la etapa 1, 2, 3 y 4 de la Figura 4.1.

A continuación, el chatbot queda a la espera de la respuesta del usuario (opción escogida), y de acuerdo a esta tiene dos caminos a seguir. En caso de que seleccione la 4ta opción (comunicarse directamente con un profesional), la herramienta le envía un mensaje a través de WhatsApp al profesional a cargo, indicándole en el mensaje el celular del usuario que requiere de su ayuda y con esto se da por finalizada la conversación con el chatbot, quedando este a la espera de un nuevo chat.

Por otro lado, si el usuario selecciona una de las 3 primeras opciones, el chatbot envía la respuesta respectiva a la pregunta escogida de la base de datos. Luego, la herramienta envía otro mensaje en el que se consulta si es que se tiene alguna otra pregunta. En caso de que el usuario quiera realizar otra interrogante, el chatbot repite todo el proceso anterior desde la etapa 2 en adelante. Finalmente, si el usuario no quiere consultar nada más, el chatbot envía un mensaje de despedida y da por finalizada la conversación, quedando a la espera de un nuevo chat. El proceso indicado anteriormente se puede visualizar en las etapas 5, 6, 7, 8, 9 y 10 de la Figura 4.1.

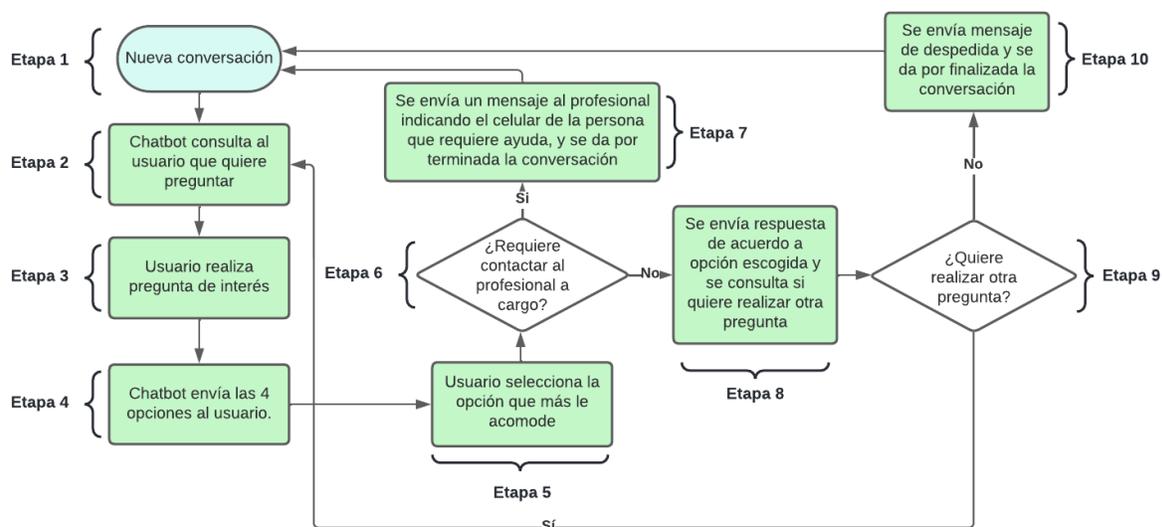


Figura 4.1 Diagrama de flujo de las etapas del chatbot

Para el diseño del chatbot se consideraron dos tipos de requisitos, los requisitos funcionales (RF), que son aquellos que tienen relación con las funcionalidades del chatbot y los requisitos no funcionales (RNF), que son características deseables del chatbot y que no afectan su funcionamiento. Estos requisitos están detallados en la Tabla 4.1.

Tabla 4.1 Requerimientos funcionales (RF) y Requerimientos No Funcionales (RNF) del Chatbot

Requerimiento	Descripción del requerimiento
RF1	El chatbot permite la generación de una conversación dando respuesta a cierta consulta ingresada por el usuario
RF2	Permite una conversación fluida dando respuestas coherentes a lo preguntado
RF3	En caso que el usuario requiera contactarse con un profesional del área, el chatbot genera la derivación correspondiente
RF4	La herramienta es capaz de dar respuesta a aquellas preguntas indicadas en la base de datos
RNF1	El chatbot es fácil de usar e intuitivo
RNF2	El chatbot requiere de respuestas por parte del usuario para generar la conversación una vez iniciada esta
RNF3	El diseño del chatbot debe guiar al usuario en su uso
RNF4	El chatbot está diseñado para implementarse en la plataforma de WhatsApp

4.3. Desarrollo del Chatbot

4.3.1 Adquisición de software y bibliotecas

A. Instalación de Anaconda y Spyder

Para la instalación de este software se utilizó el entorno de *Anaconda* en su versión 1.9.12 [69]. Anaconda incluye una serie de paquetes, aplicaciones, softwares y bibliotecas como por ejemplo el software *Spyder*, el cual “*es un entorno de desarrollo integrado multiplataforma de código abierto*” [70], que nos permitirá realizar programación en *Python*. A continuación, se mencionarán los pasos para la instalación de este software:

1. Dirigirse a la página oficial de Anaconda, en la categoría de productos y generar la descarga del software según el sistema operativo a utilizar (en este caso es Windows-64-Bit).
2. Una vez instalado *Anaconda*, se debe abrir Anaconda Navigator, donde se visualizarán una variedad de paquetes que ofrece este software, para instalar y los que ya vienen preinstalados.
3. Finalmente, se puede abrir el entorno de *Spyder* y observar la versión de *Python* que está instalada (en este caso la versión 3.8).

B. Instalación de NLTK

Como se mencionó anteriormente, *NLTK* es un conjunto de programas y bibliotecas para el Procesamiento del Lenguaje Natural (NPL) para *Python*. Según los requerimientos indicados en la página oficial, se requiere de las versiones 3.6, 3.7, 3.8 o 3.9 de *Python* para su instalación. A continuación, se darán a conocer los pasos para su instalación:

1. Se debe abrir Anaconda prompt, e ingresar el comando `<<conda install -c anaconda nltk>>`, para generar la instalación de la biblioteca. Luego se indican la actualización de algunos paquetes, y finalmente se descarga e instala *NLTK*.
2. Para descargar todos los paquetes de esta biblioteca, en la consola de Anaconda prompt se deben ingresar los siguientes comandos `<< import nltk>>` y luego `<< nltk.download() >>`. A continuación, se abrirá una pestaña de NLTK Downloader, y luego se debe seleccionar “All packages” y presionar descargar, como se ve en la Fig.

B. 1 del Anexo B. De esta forma la biblioteca y todos sus paquetes están disponibles para su uso.

C. Otras bibliotecas

Para el desarrollo del proyecto se requiere de la instalación de otras bibliotecas. Entre las más destacadas tenemos: *pandas* (herramienta que permite el análisis y la manipulación de datos, construida para *Python*), *Numpy* (permite la creación de vectores y matrices grandes multidimensionales, y que junto a una gran variedad de funciones permite la manipulación de los mismos) [71], *sentence_transformers* (este framework de *Python* contiene el modelo a utilizar para la generación de las incrustaciones de las distintas frases), entre otras bibliotecas utilizadas. La instalación de lo anteriormente mencionado se realiza indicando el siguiente comando en la consola cmd (símbolo del sistema) `<< pip install + "nombre de la biblioteca" >>` (Ejemplo: `<< pip install numpy >>`).

4.3.2 Generación de datos

Los datos sobre las preguntas y respuestas a utilizar para el chatbot fueron elaboradas junto a la empresa HEALTHTRACKER ANALYTICS [72]. Estas se relacionan a las consultas más comunes y repetitivas que realizan los cuidadores a la empresa, además de información extra que en conjunto con el equipo se consideró importante de agregar. En la TABLA 4.2 se pueden observar las preguntas seleccionadas en conjunto con un experto en cuidados paliativos de HealthTracker para realizar las pruebas en el chatbot.

Una vez recopiladas, se buscó en la web y en páginas oficiales relacionadas a cuidados paliativos, como la Sociedad Española de Cuidados Paliativos (SECPAL) [73] o la Asociación Latinoamericana de Cuidados Paliativos (ALCP) [74], información para responder a estas interrogantes. Luego que el documento estuvo listo, fue enviado a la empresa para la revisión de este, siendo aprobado para las pruebas a realizar.

TABLA 4.2 PREGUNTAS A UTILIZAR EN EL CHATBOT.

Enumeración	Pregunta
1	¿Por qué usted o su familiar han sido derivados a Cuidados Paliativos?
2	¿Qué quiere decir mitigar o contener los síntomas?
3	Cuidados de la boca
4	Cuidados del cuidador principal o de la familia del paciente
5	¿Qué hacer ante una hemorragia?
6	¿Qué situaciones de emergencias se pudieran vivir?
7	¿Qué es una lesión por presión (LPP)?
8	¿Qué hacer si el paciente no quiere comer o tiene falta de apetito?
9	¿Qué cuidados se deben tener en las vías subcutáneas?
10	¿Cuáles son los cuidados básicos que se deben tener con un paciente en cama?
11	¿Qué es la Cobertura Adicional para Enfermedades Catastróficas (CAEC)?
12	¿Qué otros beneficios incluye el GES?
13	¿Dónde puedo comprar los medicamentos prescritos
14	¿Qué es una enfermedad terminal?
15	¿Qué son los cuidados paliativos?
16	Síntomas asociados al tratamiento del dolor
17	¿Por qué es importante mantener informado al paciente?
18	¿Como saber que el paciente va a morir?
19	¿Qué hacer en una descompensación del paciente y/o de sus familiares?
20	¿Qué es el duelo?
21	¿Cómo tratar una lesión por presión (LPP)?
22	¿Cuál es el horario de atención?
23	¿Cómo sé que una vía subcutánea está infectada?
24	¿Cada cuánto tiempo se debe mover a un paciente en cama?
25	¿Qué sucede si la Isapre no cumple con el GES?
26	¿Qué cubre mi Isapre?
27	¿Cuántas atenciones tengo garantizadas?

4.3.3 Incrustaciones o WordEmbeddings

Como primer paso para el desarrollo de este proyecto, se requiere entregar al computador las frases o consultas que los cuidadores quisieran realizar, pero en un lenguaje entendible por el algoritmo. Por esto se necesitó vectorizar las consultas realizadas para que, de esta manera, y por medio de otros procedimientos mencionados más adelante, el computador de alguna forma entienda lo que quiere decir el usuario.

Por lo anterior, se hizo uso del modelo multilingüe pre-entrenado de *SBERT*, debido a que tiene como ventaja el generar incrustaciones en espacios vectoriales alineados. En otras palabras, entradas similares en diferentes idiomas (del inglés al español) se asignan próximos en el espacio vectorial. Específicamente se utilizó el modelo llamado “*distiluse-base-multilingual-cased-v1*” [75], el cual tiene la particularidad de aceptar cerca de 15 idiomas distintos, entre ellos el español, y obtener mejores resultados que otros modelos multilingües disponibles.

Una vez escogido el modelo se realizó la instalación de *SBERT* en *Python*, lo cual se hizo haciendo uso del comando `<< pip install -U sentence-transformer >>`. Posterior a esto, solo se debe indicar el modelo a utilizar, como se visualiza en el Algoritmo 1.

Algoritmo 1 Cómo cargar un modelo pre-entrenado de SBERT en Python

//Luego de cargar la biblioteca “SentenceTransformer”, simplemente se indica el nombre del modelo pre-entrenado a utilizar, y luego de un par de segundos este está listo para ser utilizado.

```
model = SentenceTransformer('distiluse-base-multilingual-cased-v1')
```

Después de que está cargado el modelo, se pueden utilizar todas las herramientas que están asociadas a él. Para el caso de este trabajo, se hará uso del comando `<<model.encode()>>`, el cual nos genera las incrustaciones de las frases completas. Es decir, se obtienen los vectores necesarios para que el computador pueda entender lo ingresado por el usuario (de lenguaje natural a lenguaje de computadora).

Cabe destacar que una ventaja que tiene el utilizar *SBERT* es que permite ingresar una misma frase escrita con faltas de ortografía, con mayúsculas o minúsculas o incluso el escribir mal una palabra, y el modelo permite que lo ingresado siga siendo comprendido por el computador, a pesar de encontrarnos en los casos mencionados anteriormente. Esto se debe a que *SBERT* hace uso de herramientas del *NPL* como la tokenización y normalización del texto, los *Transformers*, uso de *embeddings*, entre otros. Estos le permiten considerar el contexto de las frases completas y no solo enfocarse en una sola palabra que pudiera afectar la comprensión del computador, como lo hacen otros modelos mencionados anteriormente.

4.3.4 Similitud entre la consulta ingresada por el usuario y aquellas disponibles en la base de datos del Chatbot

Luego de que se obtienen las incrustaciones de las frases, el siguiente paso es el encontrar la consulta más similar a la ingresada por el usuario. Para esto se hizo uso de la distancia coseno o similitud coseno, la cual es “una medida de similitud entre dos vectores distintos de cero en un espacio que posee un producto interior con el que se evalúa el valor coseno del ángulo comprendido entre ellos” [76]. Este se calcula como se indica en la

ecuación (4.1). En la minería de textos esta métrica se utiliza mucho para identificar semejanza entre textos, por lo que se escogió como medida de similitud.

$$\text{Similitud Coseno} = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_1^n A_i B_i}{\sqrt{\sum_1^n A_i^2} \sqrt{\sum_1^n B_i^2}} \quad (4.1)$$

donde,

A y B : vectores multidimensionales que representan a dos palabras

$A \cdot B$: es el producto escalar entre los vectores A y B

$\|A\|$: es la norma o magnitud del vector A (distancia entre el punto inicial y el final)

$\|B\|$: es la norma o magnitud del vector B (distancia entre el punto inicial y el final)

n : es la dimensión de los vectores A y B

Similitud Coseno: medida de distancia que representa la cercanía de 2 palabras en un espacio multidimensional.

El valor de esta medida se encuentra entre el intervalo $[-1, 1]$, en donde si el resultado es 1, quiere decir que el ángulo entre ambos vectores es 0, por lo que ambos vectores apuntan en una misma dirección. En cambio, si el resultado da -1, quiere decir que ambos vectores apuntan en sentido contrario. Para el caso de este proyecto, obtener un resultado igual a 1 quiere decir que la consulta ingresada por el usuario es idéntica a alguna pregunta de la base de datos del chatbot, y en caso de obtener -1, se refiere a que lo consultado no se encuentra dentro de alguna pregunta de la base de datos.

Para poder calcular esta distancia en *Python* se desarrolló una función, la cual hace uso de la biblioteca *Numpy*. Esta biblioteca permite “la creación de matrices y vectores multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas” [71]. En esta función se calcula el producto escalar de dos vectores escogidos y este valor se divide por la multiplicación de la norma vectorial de ambos, obteniendo de esta manera la similitud coseno, y resultados entre $[-1,1]$.

4.3.5 Selección de respuesta al usuario

Una vez que se logró calcular la distancia coseno entre dos vectores, el siguiente paso es el identificar aquellas preguntas contenidas en la base de datos, que sean más similares a lo ingresado por el usuario. Para esto se decidió calcular la similitud coseno entre lo

consultado y las preguntas de la base de datos del chatbot, y escoger las 3 mayorías, es decir, las 3 interrogantes más similares a la consulta realizada por el usuario.

Se decidió realizar lo anterior como una forma de darle opciones al usuario en el caso de que existan 1 o más preguntas que tengan una alta semejanza con lo consultado. Además, se aprovechó esto para agregar una cuarta opción en la que se le permite al cuidador comunicarse directamente con alguien del equipo de coordinación de HealthTracker (Figura 4.2). Esto se agregó ya que puede que el usuario requiera de una especialista para resolver sus dudas y no de un sistema automático.

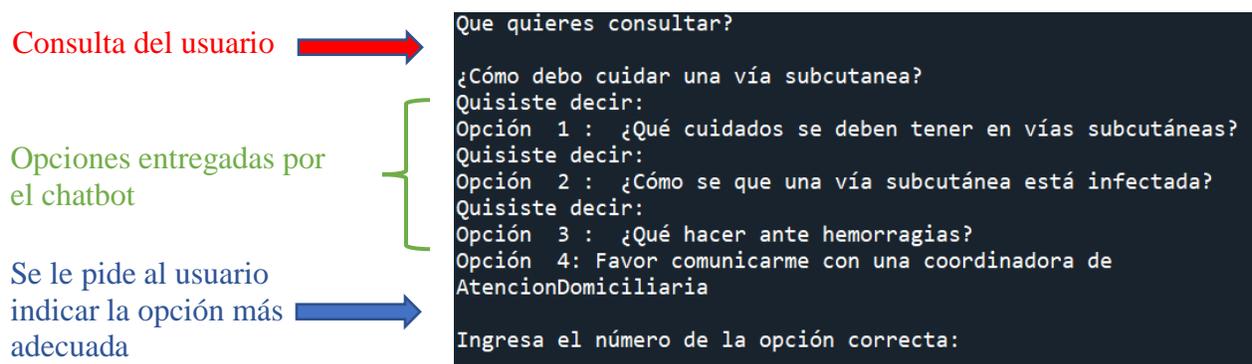


Figura 4.2 Visualización de la interfaz del usuario al ingresar una consulta

En la imagen se puede observar que al preguntar ¿Cómo debo cuidar una vía subcutánea?, el chatbot le da 4 opciones al usuario, las primeras 3 son las 3 preguntas de la base de datos más semejantes a lo consultado y la 4ta opción es el poder comunicarse con algún coordinador de la empresa.

Luego de que se pudo comprobar el correcto funcionamiento del modelo utilizado al generar las incrustaciones, el siguiente paso es el realizar pruebas en donde se vayan ingresando las mismas consultas escritas de distintas maneras e incluso con faltas de ortografía. De esta forma, se busca demostrar que efectivamente *SBERT* hace uso del contexto de la palabra y de otros métodos para poder identificar el correcto sentido de la palabra en la oración. Con los resultados visualizados se puede observar que independiente de cómo se ingrese lo consultado el algoritmo da la misma respuesta para todos los casos (Figura 4.3).

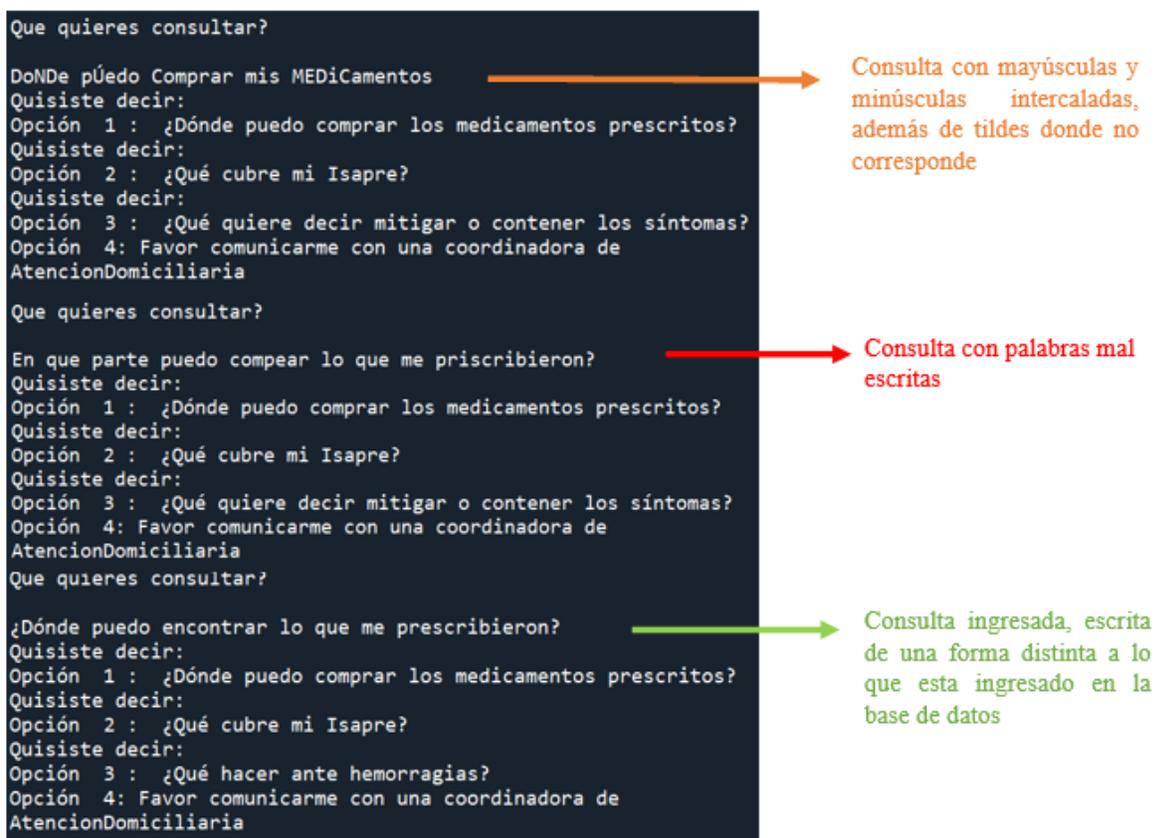


Figura 4.3 Resultados del Chatbot al ingresar la misma pregunta de variadas formas

Como se puede ver en la imagen al consultar de diferentes maneras “¿Dónde puedo comprar los medicamentos prescritos?”, se observa que el chatbot independiente de si lo ingresado tiene faltas ortográficas o está escrito con mayúsculas o minúsculas sigue comprendiendo que en general es la misma consulta, por lo que en los 3 casos da las mismas opciones.

Como una forma de hacer más interactiva la comunicación entre el chatbot y el usuario, se decidió estructurar el algoritmo dentro de un bucle. En este, una vez que el cuidador realiza una consulta y esta es respondida, se le da la opción de generar otra pregunta con el objetivo de que la conversación fluya. De esta manera se logra que la charla sea más natural (Figura 4.4).

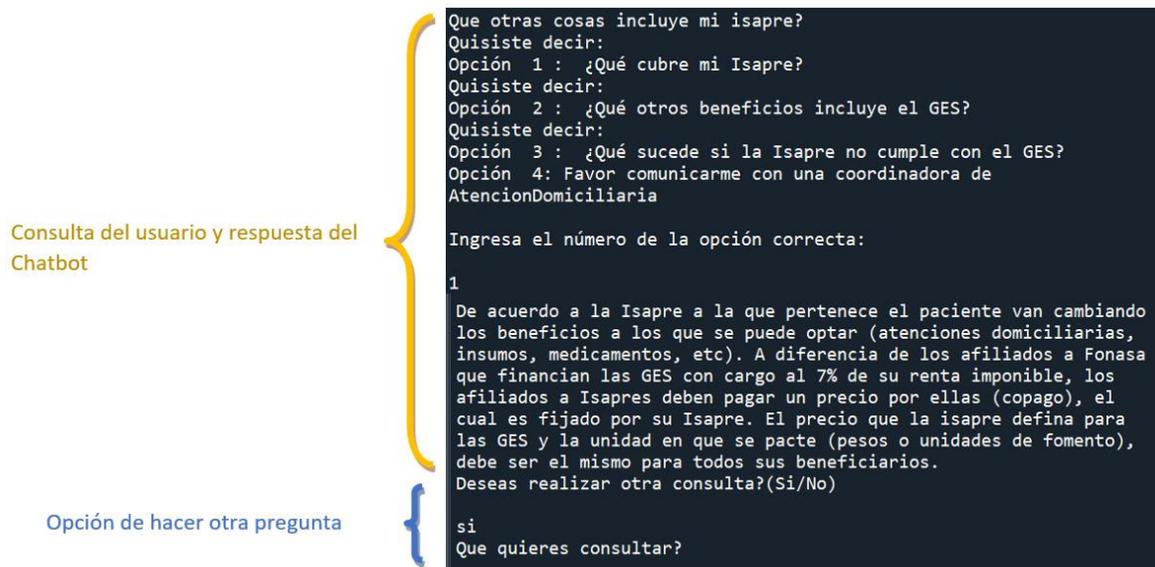


Figura 4.4 Visualización del interfaz de usuario al darle la opción de realizar otra consulta

Como se puede observar en la imagen cada vez que una pregunta es respondida por el chatbot se le da la opción al usuario de realizar otra, en caso de que este quisiera dando respuesta “si/no”, si la respuesta es “si”, permite ingresar otra consulta, y de ser “no” genera que la conversación finalice.

4.4. Conexión con WhatsApp

Para poder implementar el chatbot desarrollado en la plataforma de WhatsApp, se hace uso de una biblioteca llamada *PyAutoGUI*, la cual permite que, a través de sus secuencias de comandos de *Python*, se controle el mouse y el teclado para la automatización de las interacciones con otras aplicaciones [77].

Una vez lograda la automatización del mouse, el siguiente paso a seguir fue tener abierto WhatsApp Desktop y conectado al teléfono celular escogido. El tener abierta esta app permite que se pueda tener un control permanente de esta.

Para poder hacer la conexión entre el chatbot y WhatsApp, se desarrollaron una variedad de funciones las cuales se describirán a continuación:

- *get_message()*: Permite obtener los mensajes que envíen los usuarios.
- *post_response()*: Genera el envío de mensajes a través de WhatsApp.
- *modelo()*: Esta función contiene el modelo de *SBERT* utilizado y genera las opciones a seleccionar por el usuario de acuerdo a lo consultado por él.

- *process_response()*: Permite el procesamiento del chatbot de acuerdo a cómo se va desarrollando la conversación (en caso de que se requiera realizar otra consulta, se desee terminar la conversación, se necesite la derivación a un coordinado del centro de salud, etc.).
- *check_for_new_messages()*: Esta función está constantemente verificando nuevos mensajes con el objetivo de que el chatbot entre en un bucle en el que está todo el tiempo revisando si es que hay nuevos mensajes y respondiendo según corresponda.
- *verificacion_despedida()*: esta función tiene por objetivo que el chatbot identifique mensajes de agradecimiento o despedida (como “Gracias”, “Hasta luego, muchas gracias por la ayuda”), para que no los considere como un mensaje de una nueva conversación o bien que el usuario quiere seguir realizando consultas.
- *contacto_desconocido()*: Tiene como finalidad sacar el mensaje que WhatsApp envía cuando un número desconocido te habla (“El remitente no está en tu lista de contactos. Reportar, bloquear, ok”) y permitir que el chatbot continúe de forma normal la conversación.

4.5. Discusión

Como se pudo observar en este capítulo, la adquisición e instalación de los software, bibliotecas y modelos es necesaria para el desarrollo del trabajo.

Cabe destacar que desarrollar la base de datos de las preguntas del chatbot en un archivo “.xlsx”, permite que sea más fácil ir incorporando nuevas preguntas o bien ir modificando las que actualmente están en este archivo. En cambio, al usar otro tipo de métodos como los archivos JSON, YAML, SQL, entre otros, se dificultaba el acceso al requerir de otros programas o licencias para su correcto funcionamiento.

Una vez que se contó con todos los paquetes necesarios para desarrollar el algoritmo, se desarrollaron dos formas de poder generar las incrustaciones. La primera se realizó haciendo uso del modelo *FastText*, específicamente utilizando un modelo pre-entrenado en español llamado “*Spanish Billion Word Corpus*” [78]. El problema que se encontró a medida que se fue desarrollando el algoritmo con este método, es que se hallaron dificultades como que en el corpus pre-entrenado no se consideraban palabras como “GES” o “Fonasa”, lo que era algo esperable. Para superar esto, se desarrolló un modelo desde cero de *FastText*, y se

entrenó con una serie de documentos sobre el *GES* provenientes de las páginas web de las distintas *Isapres*, con el objetivo de tener un corpus que nos permita obtener incrustaciones adecuadas a lo que se quiere lograr en el proyecto.

Una vez entrenado el modelo con el vocabulario entregado, para abordar la problemática de generar las incrustaciones de las frases, se decidió calcular el promedio de los vectores (palabras incrustadas) que conforman cada oración ingresada. Luego de que se calculan las “Sentences Embeddings” o “Incrustaciones de las frases”, se hizo uso de la similitud coseno para calcular la semejanza entre lo ingresado por usuario y las preguntas de la base de datos, y en este punto es donde se encontraron errores.

Dentro de los errores encontrados estaba que independientemente de lo que se fuera consultando, la similitud coseno arrojaba valores muy altos con todas las preguntas de la base de datos, incluso con aquellas que no tenían ninguna relación con lo consultado. Luego de revisar detalladamente el algoritmo, se llegó a la conclusión de que el error se generaba al momento de calcular la incrustación de cada frase por medio del promedio de los vectores de las palabras que la conformaban. Esto provoca que los embeddings se “distorsionaran” al punto de que luego de calcular el promedio, todos quedaban muy cercanos en el espacio vectorial.

Por lo anterior, se realizó una búsqueda de otros métodos que permitieran la incrustación de la frase completa de forma directa y no de cada palabra que la conformara, para que de este modo se evitara el problema anteriormente mencionado.

Es así como se encontró el modelo *SBERT*, específicamente el modelo pre-entrenado “*distiluse-base-multilingual-cased-v1*”, el cual como se mencionó anteriormente, genera incrustaciones de oraciones en una variedad de idiomas entre ellos el español. Así este dio solución tanto al problema de un corpus en nuestra lengua nativa y la generación de las incrustaciones de las frases.

Una vez que se encontró un modelo capaz de generar las “Sentences Embeddings” de forma correcta, se continuó con el desarrollo del proyecto con los pasos mencionados anteriormente, obteniendo resultados bastante buenos en relación a la semejanza entre lo ingresado por usuario y lo contenido en la base de datos. Estos resultados incluso se replican al momento de ingresar las consultas con faltas ortográficas, haciendo uso de minúsculas y

mayúsculas e incluso escribiendo mal una palabra. Por ello se considera que el proyecto en los casos evaluados da como respuesta la pregunta más cercana entre los primeros resultados.

Luego de tener los elementos básicos para el desarrollado del chatbot (bibliotecas, modelos, funciones, base de datos, etc.), el siguiente paso a seguir es el generar la conexión con WhatsApp. Una vez que se generó una investigación relacionada al uso de la API de WhatsApp, se optó por buscar otra alternativa para generar la conexión. Lo anterior debido a los siguientes puntos considerados: para que WhatsApp permita el uso de su API requiere de un tiempo de 2 a 3 semanas para poder evaluar datos asociados al número que se quiere conectar y a la empresa que va a manejar el número telefónico, y una vez evaluado esto ver si es que da acceso a su API; se requiere pagar una suscripción a alguna página que permita la conexión de la API con *Python* para poder implementar el chatbot mencionado en WhatsApp y por último una vez que se logaran los dos puntos anteriores, se debían realizar pruebas para ver la correcta conexión con la app, además del correcto funcionamiento del chatbot como tal.

Por lo anterior, se decidió hacer uso de la biblioteca *PyAutoGUI*, la cual permite el manejo automático del mouse y teclado de acuerdo a lo que se quiera realizar, además de las funciones mencionadas en el subcapítulo 4.4. El uso de estos dos elementos permite la implementación de un chatbot funcional a través de la plataforma de WhatsApp Desktop, el cual cumple con el objetivo deseado que es el mantener una conversación coherente con el usuario y de acuerdo a los requerimientos funcionales indicados anteriormente.

Capítulo 5. Evaluación Chatbot

5.1. Introducción

En el presente capítulo se presenta la implementación y pruebas realizadas al chatbot para evaluar la funcionalidad del software, así como el punto de vista del especialista en cuidados paliativos sobre el funcionamiento y utilidad de esta herramienta.

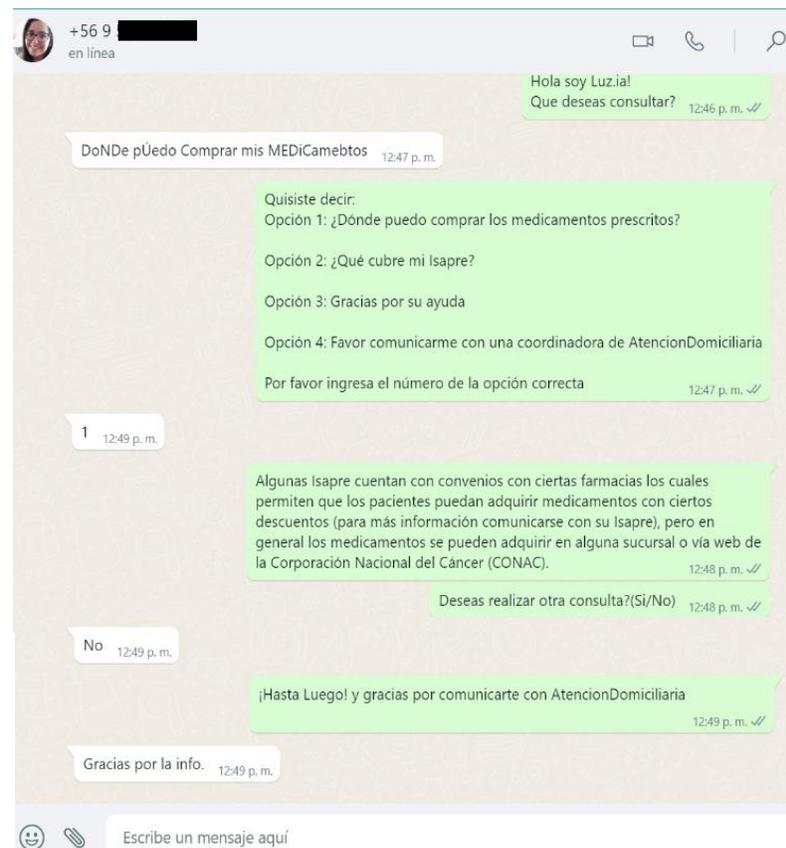
5.2. Resultados Evaluación

A continuación, en las siguientes imágenes se visualizan los resultados del chatbot al generar una misma consulta, pero realizada de 3 formas diferentes a través de la plataforma de WhatsApp (bien escrita (Figura 5.1.c), con faltas de ortografías (Figura 5.1.b) e intercalando mayúsculas y minúsculas en el mensaje (Figura 5.1.a)).

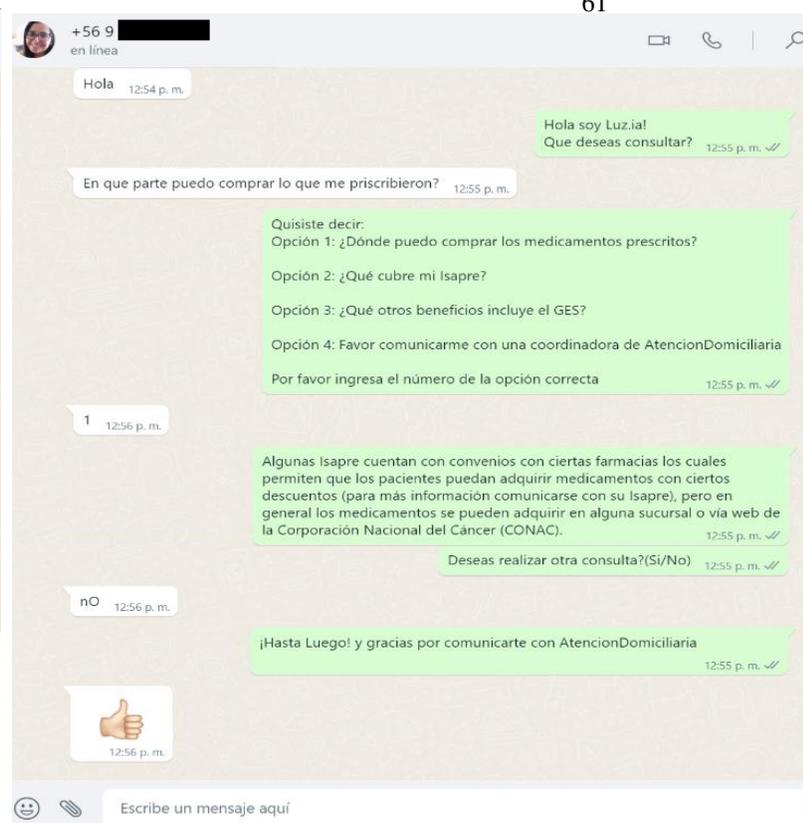
En la Figura 5.1 se puede observar el saludo de bienvenida que da el chatbot una vez que identifica que un usuario quiere realizar una consulta, y además se envía un mensaje para que se ingrese lo que se desea preguntar. Luego que el usuario envía lo consultado, el chatbot responde enviando las 4 opciones relacionadas a lo preguntado (las 3 más similares y la cuarta opción de comunicarse con una coordinadora). Posterior a esto, se solicita que el usuario ingrese la opción correcta. Finalmente, se envía un mensaje con la respuesta correspondiente, y se consulta por si desea realizar otra pregunta, y de acuerdo a lo que el cuidador/paciente responda, el chatbot envía un mensaje final o uno asociado a continuar con la conversación.

En las figuras del Anexo C, se pueden observar todos los casos que se consideraron en el desarrollo del software, tales como:

- Identificación de agradecimiento/despida por parte del usuario (Fig. C. 1)
- Identificación de que se quiere realizar una consulta (Fig. C. 1)
- Derivación a coordinador en caso de que el usuario lo requiera (Fig. C. 2)
- Solicitud por parte del chatbot de enviar una respuesta acorde a lo requerido en caso de que lo ingresado por el cuidador/paciente sea erróneo a lo pedido (Fig. C. 3)
- Conversación fluida de acuerdo a cierta consulta por parte del usuario (Fig. C. 4)



(a)



(b)



(c)

Figura 5.1 Resultados del Chatbot al ingresar la misma pregunta de variadas formas en WhatsApp

Como se puede ver en la imagen al consultar de diferentes maneras “¿Dónde puedo comprar los medicamentos prescritos?”, se observa que el chatbot independientemente de si lo ingresado tiene faltas ortográficas o está escrito con mayúsculas o minúsculas, sigue comprendiendo que en general es la misma consulta, por lo que en los 3 casos da las mismas opciones.

El chatbot desarrollado fue evaluado por un experto en cuidados paliativos de la empresa HealthTracker, María del Pilar Bonatti Escobar. La evaluación realizada por la profesional se hizo en 5 oportunidades distintas, con el objetivo de que se realizarán diferentes preguntas (escritas de variadas formas), se lograra probar la herramienta de una forma más completa y se obtuvieran variadas evaluaciones al ser realizadas en momentos distintos.

La evaluación se realizó en base a la rúbrica indicada en la Tabla D. 1 del Anexo D, la cual se desarrolló en base a la Escala de Likert (Tabla D. 2). Esta escala es un *“método de investigación que se utiliza para conocer el nivel de acuerdo y desacuerdo de las personas sobre cierto tema”* [79].

Para poder determinar la evaluación final del chatbot, lo que se hizo fue sacar un promedio de aquellas preguntas que tenían un puntaje de 1 a 5 (siendo 1 totalmente en desacuerdo y 5 totalmente de acuerdo) (Ver Tabla D. 5). Por lo anterior, solo se tomaron en cuenta 7 de las 8 preguntas, ya que una de ellas va dirigida a conocer opiniones generales del evaluador.

Se puede mencionar que:

- En la gran mayoría de pruebas realizadas, el chatbot identificó las preguntas que la experta le realizó, obteniendo un puntaje igual a 4.8.
- En general, las respuestas respondidas por la herramienta fueron coherentes a lo consultado, se obtuvo un puntaje promedio igual a 4.8.
- El lenguaje utilizado según la profesional es el adecuado según el objetivo del chatbot, aunque en algunos casos este se podría mejorar. Se obtuvo un puntaje de 4.4.
- Se llegó a la conclusión de que el chatbot es amigable con el cuidador/paciente, debido a que en general utiliza terminologías entendibles por el usuario y da instrucciones claras. Puntaje obtenido 4.6.
- En la pregunta de si la herramienta es de fácil uso e intuitivo, se obtuvo una puntuación de 4.6.
- Al consultar si se considera que las instrucciones dadas por el chatbot son claras se logró un puntaje de 4.6.

- Finalmente, en la consulta relacionada a la evaluación general del chatbot (puntaje de acuerdo a la experiencia del uso general de la herramienta), se obtuvo un puntaje de 4.8.

Todas las evaluaciones entregadas por la experta en cuidados paliativos se pueden encontrar en la Tabla D. 3 y Tabla D. 4 del Anexo D.

En relación a la pregunta que considera la opinión general de la evaluadora, a modo resumen se puede decir que la experta considera que la herramienta es de gran apoyo tanto para el paciente/cuidador como para los profesionales. En algunos casos menciona que cambiaría algunos conceptos o acortaría las respuestas para que estas sean de mejor comprensión para el usuario. Destaca que además, en algunas de las interrogantes como el caso de ¿Cada cuánto tiempo se debe mover a un paciente en cama?, sería bueno el agregar imágenes o videos complementarios que ayuden de mejor forma al usuario. Menciona a futuro la idea de agregar una vez finalizada la conversación con el chatbot, una evaluación por parte del usuario para saber que tanto le sirvió la herramienta, poder aplicar el chatbot al equipo de salud para tratar temas como el manejo de opioides o farmacología en general, temas administrativos, entre otras ideas referidas.

5.3. Discusión

En el presente capítulo se presenta el comportamiento del chatbot ante distintas situaciones y la evaluación realizada por la enfermera especialista en cuidados paliativos Pilar Bonatti, en relación al funcionamiento de la herramienta.

En general, las evaluaciones realizadas obtuvieron buenos resultados, obteniendo en todas las preguntas evaluadas (con un puntaje de 1-5) un puntaje promedio sobre 4.4. Lo anterior se puede traducir a que el chatbot comprende de forma correcta lo que el usuario está consultando, y este envía respuestas coherentes a lo preguntado, permitiendo una fluidez en la conversación con el usuario y cumpliendo con los objetivos generales del proyecto. Además, de acuerdo a las evaluaciones recibidas, se puede decir que la información entregada es de fácil comprensión, clara y permite que el cuidador/paciente reciba respuestas rápidas y oportunas ante las interrogantes más comunes que le pudieran surgir.

Cabe destacar que en aquellos casos en que la evaluadora ingresaba alguna consulta que no estuviera en la base de datos, se le mostró el mecanismo de derivación a un

coordinador del centro de salud, encontrando que este era rápido y eficiente en identificar cuál era el usuario que requería de la ayuda de un profesional. Además, indica que el proceso requiere de poco tiempo para notificar al equipo de Atención Domiciliaria sobre la derivación.

Se puede decir que en general el chatbot se comportó de acuerdo a lo esperado, pudiendo responder sin problemas en todas las pruebas que se le realizaron. Uno de los mayores problemas que se encontró en la herramienta es al momento de que un usuario agradeciera o se despidiera del chatbot con texto, emojis o stickers, debido a que, al enviar un nuevo mensaje una vez finalizada la conversación, la herramienta lo consideraba como una consulta. Para corregir este problema, se consideró agregar una nueva función que identificara este tipo de mensajes, la cual hace que el chatbot quede en espera de una nueva conversación y no responda a lo recibido.

A pesar de que no se pudo utilizar la API de WhatsApp para implementar el chatbot, se considera que la opción escogida fue la correcta, ya que cumple con los objetivos del proyecto y permite el correcto funcionamiento de la herramienta. Se espera que a futuro esta se pueda implementar haciendo uso directo de la API antes mencionada y además se pueda ampliar la base de datos para poder así abarcar una mayor cantidad de interrogantes.

Capítulo 6. Conclusiones

6.1. Sumario y Discusión

En este estudio se evidenciaron una serie de datos estadísticos relacionados al cáncer y cómo año a año va incrementando la cantidad de pacientes que lo padecen a nivel mundial. Esto obliga a que los estados y las distintas instituciones de salud tengan como foco principal generar una serie de cambios y reformas en sus modelos de salud, que tengan como objetivo el batallar con esta enfermedad, ofreciendo una variedad de prestaciones relacionadas al cáncer. Estas se pueden dividir en prevención y promoción, oferta de servicios como el diagnóstico y tratamiento, y servicios más complejos como los tratamientos más intensivos.

Además, se pudo evidenciar la existencia de un tipo de IA llamada Chatbot y los distintos tipos que se pueden encontrar, la cual debido a sus ventajas, permitirá el desarrollo de este proyecto. También se habló sobre el lenguaje de programación a utilizar para el desarrollo del chatbot el cual es *Python*. Este lenguaje tiene ciertas ventajas sobre otros lenguajes de programación, siendo un lenguaje más intuitivo, multiplataforma y apto para trabajar con big data, machine learning, data science e inteligencia artificial.

Igualmente, se dio una pincelada de lo que es el Procesamiento Natural del Lenguaje, las redes neuronales, los distintos modelos para la generación de incrustaciones de palabras, las nuevas tecnologías para el NPL, como los *Transformers* y *SBERT*, entre otros tópicos vistos.

Luego, se repasó el estado del arte, donde se presentan trabajos similares realizados sobre el desarrollo de chatbots, manuales asociados a los cuidados paliativos y herramientas de apoyo al cuidador. Estos permiten conocer el desarrollo actual de estos temas y obtener posibles ideas o aplicaciones que se pudieran utilizar como base para el desarrollo del software.

Luego, se dio a conocer el diseño utilizado para el desarrollo del proyecto y su respectivo progreso, haciendo mención de las bibliotecas, modelos, datos, funciones, entre otros elementos empleados. Posterior a esto, se relata el modo de evaluación utilizado, el cual fue a través de una rúbrica desarrollada en base a la Escala de Likert y evaluada por una experta en cuidados paliativos en un total de 5 evaluaciones.

Las pruebas realizadas cumplieron con los objetivos esperados, permitiendo que se probaran todas las funcionalidades del chatbot, obteniendo en promedio un puntaje por parte de la experta sobre el 4.4 en todas las preguntas, lo cual se considera óptimo para el objetivo del proyecto.

Finalmente, se puede mencionar que debido a la situación sanitaria que se vive a nivel mundial, y teniendo en conocimiento el cuidado que deben tener los pacientes oncológicos paliativos respecto a su salud, sumado a todos los beneficios que conlleva el uso de chatbot, se cree que el desarrollo de este proyecto es fundamental a la hora de ser una herramienta de ayuda para estos pacientes y sus familias.

6.2. Conclusión

Se desarrolló satisfactoriamente un chatbot para el cuidador/familiar de pacientes en cuidados paliativos, el cual permite dar respuesta de forma inmediata a las interrogantes que pudieran tener los respectivos usuarios, en relación a variados tópicos relacionados al cuidado paliativo.

Durante el desarrollo del software, se diseñó el funcionamiento del chatbot y las funcionalidades que tiene, pudiendo definir el camino a seguir por la herramienta en aquellas situaciones en que, por ejemplo, la consulta ingresada no se encontraba en la base de datos, (derivar al usuario al equipo coordinador), que hacer en caso de que el usuario quiera o no continuar con la conversación, como poder identificar cuando un usuario agradece o se despide del chatbot VS cuando quiere realizar una consulta, entre otros detalles abarcados.

Además, se definieron los contenidos que se incorporaron a la herramienta en base a las interrogantes más comunes que se realizaban en los cuidados paliativos y otros tópicos que se consideraron importantes de añadir.

Se implementó el chatbot en WhatsApp Desktop logrando un correcto funcionamiento, y permitiendo dar respuesta a una serie de interrogantes de forma instantánea.

Se evaluó la herramienta a través de una rúbrica desarrollada en base a la Escala de Likert y evaluada por la enfermera experta en cuidados paliativos, Pilar Bonatti, en un total de 5 evaluaciones, obteniendo como evaluación final el promedio de estas. Lo anterior con

el objetivo de obtener respuestas lo más variadas posibles y poder probar todas las funcionalidades del chatbot.

Con lo mencionado anteriormente, se logran los objetivos planteados en la memoria, sobre el desarrollar una chatbot como herramienta de apoyo y educación para el cuidador a cargo de pacientes en etapa GES 4, implementado en la plataforma de WhatsApp.

6.3. Trabajos Futuros

De acuerdo a lo logrado en este trabajo y a lo conversado con la empresa Health Tracker Analytics, se espera continuar con el desarrollo del chatbot, aumentando la base de datos disponible, considerando otro tipo de funcionalidades para la herramienta como el ser capaz de comprender situaciones fuera de las consideradas en su diseño, poder implementar el chatbot en otras plataformas, como otras redes sociales o en alguna página web propia de la empresa. También se propone permitir la resolución de consultas de forma simultánea, con más de un usuario a la vez, y el implementar el chatbot directamente con los profesionales de salud para abordar temas relacionados a la farmacología, temas administrativos, entre otros.

Finalmente, se espera poder realizar pruebas de la herramienta con usuarios reales, para evaluar de mejor manera la experiencia del cuidador/paciente y el poder identificar las deficiencias del chatbot o tópicos que no se consideraron en su diseño. Además, se espera tener un seguimiento del software a lo largo de los meses, con el fin de determinar posibles mejoras en la herramienta, ver el comportamiento de los usuarios, obtener estadísticas en relación a la utilidad del chatbot, tanto para los pacientes como para Atención Domiciliaria. Además, se espera generar evaluaciones por parte de los usuarios, con el objetivo de conocer su opinión sobre la herramienta, el permitir guardar de forma anónima las conversaciones con previa autorización para el análisis de estas, que permitan dar a conocer si el chatbot es replicable a otras áreas del centro de salud.

Bibliografía

- [1] World Health Organization, “Cáncer.” [Online]. Available: <https://www.who.int/es/newsroom/fact-sheets/detail/cancer>. [Accessed: Mar. 04, 2022]
- [2] Subdepartamento de Demografía, Departamento de Estadísticas Demográficas y Sociales, and Instituto Nacional de Estadísticas (INE), “Síntesis de Resultados Estadísticas Vitales,” 2019.
- [3] Global Cancer Observatory, “Chile.” [Online]. Available: <https://gco.iarc.fr/today/data/factsheets/populations/152-chile-fact-sheets.pdf>. [Accessed: Mar. 01, 2022]
- [4] Parra-Soto S, Petermann-Rocha F., Leiva-Ordeñez A., Martínez-Sanguinetti M., Leiva-Ordeñez A., Troncoso-Pantoja C., Ulloa N., Díaz-Martínez X., Díaz-Martínez X. “Cáncer en Chile y en el mundo,” *Revista médica de Chile*, pp. 1489–1495, 2020 [Online]. Available: https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0034-98872020001001489. [Accessed: Apr. 04, 2022]
- [5] Ministerio de Salud, “Guía Clínica AUGE-Alivio del Dolor por cáncer avanzado y Cuidados Paliativos.” [Online]. Available: <https://www.minsal.cl/portal/url/item/72213ed52c2723d1e04001011f011398.pdf>. [Accessed: Apr. 04, 2022]
- [6] D. R. Chiquete., Díaz-Martínez X., Betancourt, A Mohar, Meneses García, A., Ortiz, González, Betan, Mohar. “Epidemiología del dolor por cáncer,” *Rev. Soc. Esp. Dolor*, vol. 18, pp. 118–134, 2011 [Online]. Available: <https://scielo.isciii.es/pdf/dolor/v18n2/revision1.pdf>. [Accessed: Apr. 24, 2022]
- [7] Y. H. Yun, D. S. Heo, I. G. Lee, H. S. Jeong, H. J. Kim, S. Y. Kim, Y. H. Kim, Y. J. Ro, S. S. Yoon, K. H. Lee. “Multicenter study of pain and its management in patients with advanced cancer in Korea,” *Journal of Pain and Symptom Management*, vol. 25, no. 5, pp. 430–437, May 2003, doi: 10.1016/S0885-3924(03)00103-9. [Online]. Available: [https://www.jpmsjournal.com/article/S0885-3924\(03\)00103-9/fulltext](https://www.jpmsjournal.com/article/S0885-3924(03)00103-9/fulltext). [Accessed: Apr. 24, 2022]
- [8] FONASA, “Boletín Estadístico 2017-2018 del Fondo Nacional de Salud.” [Online]. Available: <https://www.fonasa.cl/sites/fonasa/documentos>. [Accessed: Mar. 05, 2022]
- [9] Gobierno de Chile, “Plan Nacional de Cáncer.” [Online]. Available: <https://www.gob.cl/plannacionaldecancer>. [Accessed: Mar. 04, 2022]
- [10] Instituto Nacional del Cáncer Chile, “AUGE-GES.” [Online]. Available: <https://www.incancer.cl/auge-ges/>. [Accessed: Apr. 05, 2022]
- [11] Biblioteca del Congreso Nacional de Chile / BCN, “Ley 21375-consagra los cuidados paliativos y los derechos de las personas que padecen enfermedades terminales o graves.” [Online]. Available: <https://www.bcn.cl/leychile/navegar?idNorma=1166846&tipoVersion=0>. [Accessed: Mar. 06, 2022]
- [12] Salud Digital-Fundación Carlos Slim, “Chatbots: una apuesta para el beneficio del sector salud.” [Online]. Available: <https://saluddigital.com/big-data/chatbots-una-apuesta-para-el-beneficio-del-sector-salud/>. [Accessed: Apr. 04, 2022]

- [13] Campus Sanofi, “El COVID-19 dispara el uso de Chatbots de salud para ofrecer respuestas ágiles a los ciudadanos.” [Online]. Available: <https://campussanofi.es/e-patient/noticias/chatbots-covid-19/>. [Accessed: Apr. 04, 2022]
- [14] Atlantico, “Latin America Digital Transformation Report 2021,” 2021 [Online]. Available: <https://www.atlantico.vc/latin-america-digital-transformation-report>. [Accessed: Apr. 09, 2022]
- [15] TRIPODORO V., VELOSO V., and LLANOS V., “Sobrecarga del cuidador principal de pacientes en cuidados paliativos,” *Argumentos. Revista de Crítica Social*, 2015 [Online]. Available: <http://repositorioubi.sisbi.uba.ar/gsd/cgi-bin/library.cgi?e=p-10000-00---off-0--00-----0-10-0---0---0direct-10---4-----0-11--10-es-Zz-1---20-home---0--1-00-00--4---0-0-01-00-OutfZz-8-00&a=p&p=about&c=argumentos>. [Accessed: May 14, 2022]
- [16] Superintendencia de Salud, “Garantías Explicitas en Salud (GES).” [Online]. Available: http://www.supersalud.gob.cl/difusion/665/w3-propertyvalue-1962.html#accesos_fichas_ges_6. [Accessed: Mar. 10, 2022]
- [17] Superintendencia de Salud, “Alivio del Dolor y Cuidados Paliativos por Cáncer Avanzado.” [Online]. Available: <http://www.supersalud.gob.cl/difusion/665/w3-article-18738.html>. [Accessed: Mar. 04, 2022]
- [18] Asociación Latinoamericana de Cuidados Paliativos (ALCP), “Atlas de Cuidados Paliativos de Latinoamérica 2020.” [Online]. Available: <https://cuidadospaliativos.org/recursos/publicaciones/atlas-de-cuidados-paliativos-de-latinoamerica/>. [Accessed: Apr. 10, 2022]
- [19] Python Organization, “Python.” [Online]. Available: <https://www.python.org/>. [Accessed: Apr. 24, 2022]
- [20] Santander Becas, “Python: qué es y por qué deberías aprender a utilizarlo.” [Online]. Available: <https://www.becas-santander.com/es/blog/python-que-es.html>. [Accessed: May 14, 2022]
- [21] ORACLE, “¿Qué es la inteligencia artificial—IA?” [Online]. Available: <https://www.oracle.com/cl/artificial-intelligence/what-is-ai/>. [Accessed: Apr. 09, 2022]
- [22] Felipe Ramírez, “¿Qué es y cómo funciona la Inteligencia Artificial?” [Online]. Available: <https://www.uchile.cl/noticias/173079/que-es-y-como-funciona-la-inteligencia-artificial>. [Accessed: Apr. 09, 2022]
- [23] IBM, “¿Qué es machine learning?” [Online]. Available: <https://www.ibm.com/cl-es/analytics/machine-learning>. [Accessed: Apr. 09, 2022]
- [24] TXEMA RODRÍGUEZ, “Machine Learning y Deep Learning: cómo entender las claves del presente y futuro de la inteligencia artificial.” [Online]. Available: <https://www.xataka.com/robotica-e-ia/machine-learning-y-deep-learning-como-entender-las-claves-del-presente-y-futuro-de-la-inteligencia-artificial>. [Accessed: Apr. 09, 2022]
- [25] Paloma Recuero de los Santos, “Tipos de aprendizaje en Machine Learning: supervisado y no supervisado.” [Online]. Available: <https://empresas.blogthinkbig.com/que-algoritmo-elegir-en-ml-aprendizaje/>. [Accessed: Apr. 09, 2022]
- [26] SAP, “¿Qué es machine learning?” [Online]. Available: <https://www.sap.com/latinamerica/insights/what-is-machine-learning.html>. [Accessed: Apr. 09, 2022]

- [27] Ligdi Gonzalez, “Todo sobre aprendizaje supervisado en Machine Learning,” Aug. 03, 2018. [Online]. Available: <https://aprendeia.com/todo-sobre-aprendizaje-supervisado-en-machine-learning/>. [Accessed: May 16, 2022]
- [28] Ligdi Gonzalez, “Todo sobre aprendizaje no supervisado en Machine Learning,” Aug. 10, 2018. [Online]. Available: <https://aprendeia.com/todo-sobre-aprendizaje-no-supervisado-en-machine-learning/>. [Accessed: May 16, 2022]
- [29] IBM, “Redes neuronales,” Aug. 17, 2020. [Online]. Available: <https://www.ibm.com/cl-es/cloud/learn/neural-networks>. [Accessed: Apr. 16, 2022]
- [30] Digital Guide - IONOS, “Neural networks: ¿de qué son capaces las redes neuronales artificiales?,” Mar. 10, 2020. [Online]. Available: <https://www.ionos.es/digitalguide/online-marketing/marketing-para-motores-de-busqueda/que-es-una-neural-network/>. [Accessed: Apr. 16, 2022]
- [31] D. Moreira, I. Cruz, K. Gonzalez, A. Quirumbay, C. Magallan, T. Guarda, A. Andrade, C. Castillo. “Análisis del Estado Actual de Procesamiento de Lenguaje Natural Analysis of the Current State of Natural Language Processing,” *Iberian Journal of Information Systems and Technologies*, pp. 126–136, 2021 [Online]. Available: <https://www.proquest.com/openview/a44d67c88cfaada2563dc16f94ccd3c8/1?pq-origsite=gscholar&cbl=1006393>. [Accessed: May 04, 2022]
- [32] Instituto de Ingeniería del Conocimiento, “Procesamiento del Lenguaje Natural ¿qué es?” [Online]. Available: <https://www.iic.uam.es/inteligencia/que-es-procesamiento-del-lenguaje-natural/>. [Accessed: Apr. 04, 2022]
- [33] Wikipedia, “Lematización.” [Online]. Available: <https://es.wikipedia.org/wiki/Lematizaci%C3%B3n>. [Accessed: Mar. 19, 2022]
- [34] NLTK Organization, “Natural Language Toolkit.” [Online]. Available: <https://www.nltk.org/>. [Accessed: Apr. 08, 2022]
- [35] S. Raj, *Building Chatbots with Python*. Berkeley, CA: Apress, 2019.
- [36] IBM, “Natural Language Processing (NLP).” [Online]. Available: <https://www.ibm.com/cloud/learn/natural-language-processing>. [Accessed: May 14, 2022]
- [37] Bird S., Edward L., and Ewan K., *Natural Language Processing with Python*. 2009.
- [38] spaCy, “Industrial-Strength Natural Language Processing.” [Online]. Available: <https://spacy.io/>. [Accessed: Mar. 09, 2022]
- [39] Unipython, “EL MODELO EMBEDDINGS (INCRUSTACIONES) DE PALABRAS.” [Online]. Available: <https://unipython.com/el-modelo-embeddings-incrustaciones-de-palabras/>. [Accessed: Apr. 15, 2022]
- [40] Akash Deep, “Word2Vec, GLOVE, FastText and Baseline Word Embeddings step by step,” Aug. 22, 2020. [Online]. Available: <https://medium.com/analytics-vidhya/word2vec-glove-fasttext-and-baseline-word-embeddings-step-by-step-d0489c15d10b>. [Accessed: Apr. 15, 2022]
- [41] Prateek Joshi, “How do Transformers Work in NLP? A Guide to the Latest State-of-the-Art Models,” Jun. 19, 2019. [Online]. Available: https://www.analyticsvidhya.com/blog/2019/06/understanding-transformers-nlp-state-of-the-art-models/?utm_source=blog&utm_medium=demystifying-bert-groundbreaking-nlp-framework. [Accessed: Apr. 15, 2022]
- [42] Wikipedia.org, “Transformer (machine learning model).” [Online]. Available: [https://en.wikipedia.org/wiki/Transformer_\(machine_learning_model\)](https://en.wikipedia.org/wiki/Transformer_(machine_learning_model)). [Accessed: Apr. 24, 2022]

- [43] Nvidia, “What Is a Transformer Model?,” Mar. 25, 2022. [Online]. Available: <https://blogs.nvidia.com/blog/2022/03/25/what-is-a-transformer-model/>. [Accessed: Apr. 15, 2022]
- [44] Medium, “What is a Transformer?” [Online]. Available: <https://medium.com/inside-machine-learning/what-is-a-transformer-d07dd1fbec04>. [Accessed: Apr. 24, 2022]
- [45] Susana Meijomil, “Guía avanzada de Google BERT: qué es, cómo funciona y en qué te afecta,” Nov. 21, 2019. [Online]. Available: <https://www.inboundcycle.com/blog-de-inbound-marketing/google-bert-que-es-como-funciona>. [Accessed: Apr. 24, 2022]
- [46] DX MEDIA, “¿Qué es BERT y cómo funciona?” [Online]. Available: <https://dxmedia.net/algorithm-bert-google/>. [Accessed: Apr. 16, 2022]
- [47] Analytics Vidhya, “Demystifying BERT: A Comprehensive Guide to the Groundbreaking NLP Framework,” Sep. 25, 2019. [Online]. Available: <https://www.analyticsvidhya.com/blog/2019/09/demystifying-bert-groundbreaking-nlp-framework/>. [Accessed: Apr. 16, 2022]
- [48] Laia Cardona, “Google BERT: qué es, cómo funciona y cómo te afecta (2022).” [Online]. Available: <https://www.cyberclick.es/numerical-blog/google-bert-que-es-como-funciona-y-como-te-afecta>. [Accessed: Apr. 24, 2022]
- [49] Krypton Solid, “¿Qué es BERT (modelo de lenguaje) y cómo funciona?” [Online]. Available: <https://kryptonsolid.com/que-es-bert-modelo-de-lenguaje-y-como-funciona/>. [Accessed: Apr. 16, 2022]
- [50] SBERT.net, “SentenceTransformers.” [Online]. Available: <https://www.sbert.net/index.html>. [Accessed: Apr. 16, 2022]
- [51] Oracle, “¿Qué es un Chatbot?” [Online]. Available: <https://www.oracle.com/cl/chatbots/what-is-a-chatbot/>. [Accessed: May 04, 2022]
- [52] Google Trends, “Chatbot.” [Online]. Available: <https://trends.google.com/trends/explore?date=2010-01-01%202021-11-05&q=chatbot&hl=en&tz=-120>. [Accessed: Nov. 04, 2021]
- [53] R. Tamrakar, S. Vallabhbai, and N. Wani, “Design and Development of CHATBOT: A Review Dynamics of cracked rotors View project Design and Development of CHATBOT: A Review,” in *International Conference on “Latest Trends in Civil, Mechanical and Electrical Engineering”*, 2021 [Online]. Available: https://www.researchgate.net/publication/351228837_Design_and_Development_of_CHATBOT_A_Review. [Accessed: Apr. 04, 2022]
- [54] Artificial Solutions, “Chatbots: La Guía definitiva (2020).” [Online]. Available: <https://www.artificial-solutions.com/chatbots/>. [Accessed: May 04, 2022]
- [55] Chatterbot, “About Chatterbot.” [Online]. Available: <https://chatterbot.readthedocs.io/en/stable/#>. [Accessed: Mar. 12, 2022]
- [56] Landbot, “Landbot.” [Online]. Available: <https://landbot.io/>. [Accessed: Mar. 12, 2022]
- [57] A. O. Priadko, K. P. Osadcha, and B. Khmelnytsky, “Development of a chatbot for informing students of the schedule,” in *CS&SE@SW 2019 2nd Student Workshop on Computer Science & Software Engineering*, 2019 [Online]. Available: https://www.researchgate.net/publication/340685740_Development_of_a_chatbot_for_informing_students_of_the_schedule. [Accessed: May 04, 2022]
- [58] Kavitha B. R. and Chethana R. Murthy, “Chatbot for healthcare system using Artificial Intelligence,” *International Journal of Advance Research, Ideas and Innovations in Technology*, 2019 [Online]. Available:

- https://www.academia.edu/39794526/Chatbot_for_healthcare_system_using_Artificial_Intelligence?from=cover_page. [Accessed: Apr. 02, 2022]
- [59] O. Chinedu, C. W. Okonkwo, and A. Ade-Ibijola, “Python-Bot: A Chatbot for Teaching Python Programming,” *Engineering Letters*, pp. 25–34, 2021 [Online]. Available: https://www.researchgate.net/publication/349574948_PythonBot_A_Chatbot_for_Teaching_Python_Programming. [Accessed: May 01, 2022]
- [60] Tode V., Gadge H., Madane S., Kachare P., and Deokar, “A Chatbot for Medical Purpose using Deep Learning,” *INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT)*, 2021 [Online]. Available: <https://www.ijert.org/a-chatbot-for-medical-purpose-using-deep-learning>. [Accessed: May 11, 2022]
- [61] Unidad de Oncología Hospital Regional de Talca (HRT) y Universidad Católica del Maule, “Manual de Cuidados Domiciliarios para Paciente Oncológico y Familia” [Online]. Available: <https://issuu.com/ucdelmaule/docs/manual-cuidados-domiciliarios-pacie>. [Accessed: Apr. 11, 2022]
- [62] de la Cabeza M. and CaleroM., “TIC para cuidadores informales de pacientes paliativos,” *European Journal of Health Research*, vol. 3, pp. 27–39, 2017 [Online]. Available: <https://revistas.uautonoma.cl/index.php/ejhr/article/view/1179>. [Accessed: Apr. 12, 2022]
- [63] Ministerio de Salud Argentina, “Manual de Cuidados Paliativos para Cuidadores.” [Online]. Available: <https://bancos.salud.gob.ar/recurso/manual-de-cuidados-paliativos-para-cuidadores>. [Accessed: May 04, 2022]
- [64] Escuela de Enfermería Universidad de Valparaíso, “Curso para cuidadores informales - Escuela de Enfermería UV.” [Online]. Available: <https://geropolis.uv.cl/noticias/163-escuela-de-enfermeria-uv-impartio-curso-para-cuidadores-informales>. [Accessed: Apr. 04, 2022]
- [65] Pontificia Universidad Católica de Chile, “Programa Cuidador Primario.” [Online]. Available: <https://educacioncontinua.uc.cl/27327-ficha-programa-cuidador-primario>. [Accessed: Apr. 07, 2022]
- [66] Fundación General Universidad de Alcalá, “APP MÓVIL CUIDAR Y CUIDARSE.” [Online]. Available: <https://www.fgua.es/proyectos-investigacion/ciencia-salud/app-movil-cuidar-cuidarse/>. [Accessed: Apr. 07, 2022]
- [67] María José Gidi Lantadilla, “acude: Aplicación móvil para cuidadores de adultos mayores con demencia,” Pontificia Universidad Católica de Chile, 2017 [Online]. Available: https://diseno.uc.cl/memorias/pdf/memoria_dno_uc_2017_2_GIDI_LANTADILLA_M.pdf. [Accessed: Apr. 07, 2022]
- [68] Gobierno de México, “Promoverá IMSS mayor uso del chatbot para agilizar la comunicación con familiares de pacientes pediátricos oncológicos.” [Online]. Available: <http://www.imss.gob.mx/prensa/archivo/202107/319>. [Accessed: Apr. 07, 2022]
- [69] Anaconda, “Anaconda.” [Online]. Available: <https://www.anaconda.com/>. [Accessed: Jan. 31, 2022]
- [70] Wikipedia Organization, “Spyder.” [Online]. Available: [https://en.wikipedia.org/wiki/Spyder_\(software\)](https://en.wikipedia.org/wiki/Spyder_(software)). [Accessed: Jan. 31, 2022]

- [71] Wikipedia.org, “NumPy.” [Online]. Available: <https://es.wikipedia.org/wiki/NumPy>. [Accessed: Apr. 24, 2022]
- [72] Healthtracker, “Healthtracker.” [Online]. Available: <https://healthtracker.ai/>. [Accessed: Jul. 08, 2022]
- [73] Sociedad Española de Cuidados Paliativos, “Guía de Cuidados Paliativos.” [Online]. Available: https://www.secpal.com/biblioteca_guia-cuidados-paliativos-1. [Accessed: Apr. 18, 2022]
- [74] Asociación Latinoamericana de Cuidados Paliativos (ALCP), “Asociación Latinoamericana de Cuidados Paliativos.” [Online]. Available: <https://www.cuidadospaliativos.org/>. [Accessed: Apr. 18, 2022]
- [75] SBERT.net, “Pretrained Models.” [Online]. Available: https://www.sbert.net/docs/pretrained_models.html. [Accessed: Apr. 18, 2022]
- [76] Wikipedia.org, “Similitud coseno.” [Online]. Available: https://es.wikipedia.org/wiki/Similitud_coseno. [Accessed: Apr. 18, 2022]
- [77] PyAutoGUI, “PyAutoGUI.” [Online]. Available: <https://pyautogui.readthedocs.io/en/latest/>. [Accessed: Jun. 27, 2022]
- [78] Jorge Perez, Jose Cañete, and Duy Nguyen, “Spanish Word Embeddings,” *GitHub*. [Online]. Available: <https://github.com/dccuchile/spanish-word-embeddings>. [Accessed: Apr. 19, 2022]
- [79] Blog de Zendesk, “¿Qué es escala de Likert y cómo aplicarla?” [Online]. Available: <https://www.zendesk.com.mx/blog/que-es-escala-de-likert/>. [Accessed: Jun. 30, 2022]
- [80] Mathias Etcheverry, “Representaciones Distribuidas de las Palabras y Redes Neuronales Artificiales para el PLN,” in *3 er Encuentro de Ciudades Inteligentes*, 2017 [Online]. Available: <https://montevideo.gub.uy/sites/default/files/biblioteca/14001415etcheverryplnne mb3rdocidciudadesinteligentes.pdf>. [Accessed: Apr. 15, 2022]
- [81] Alexis Perrier, “Discover The Power of Word Embeddings,” Mar. 04, 2022. [Online]. Available: <https://openclassrooms.com/en/courses/6532301-introduction-to-natural-language-processing/6980911-discover-the-power-of-word-embeddings>. [Accessed: Apr. 15, 2022]
- [82] Daniel Johnson, “Cómo descargar e instalar NLTK en Windows / Mac.” [Online]. Available: <https://www.guru99.com/download-install-nltk.html>. [Accessed: Apr. 10, 2022]

Anexo A. Requerimientos del Chatbot

Tabla A. 1 Requerimientos funcionales (RF) y Requerimientos No Funcionales (RNF) del Chatbot

Requerimiento	Descripción del requerimiento
RF1	El chatbot permite la generación de una conversación dando respuesta a cierta consulta ingresada por el usuario
RF2	Permite una conversación fluida dando respuestas coherentes a lo preguntado
RF3	En caso que el usuario requiera contactarse con un profesional del área, el chatbot genera la derivación correspondiente
RF4	La herramienta es capaz de dar respuesta a aquellas preguntas indicadas en la base de datos
RNF1	El chatbot es fácil de usar e intuitivo
RNF2	El chatbot requiere de respuestas por parte del usuario para generar la conversación una vez iniciada esta
RNF3	El diseño del chatbot debe guiar al usuario en su uso
RNF4	El chatbot esta diseñado para implementarse en la plataforma de WhatsApp

Anexo B. Paso a paso para la descarga de los paquetes de NLTK

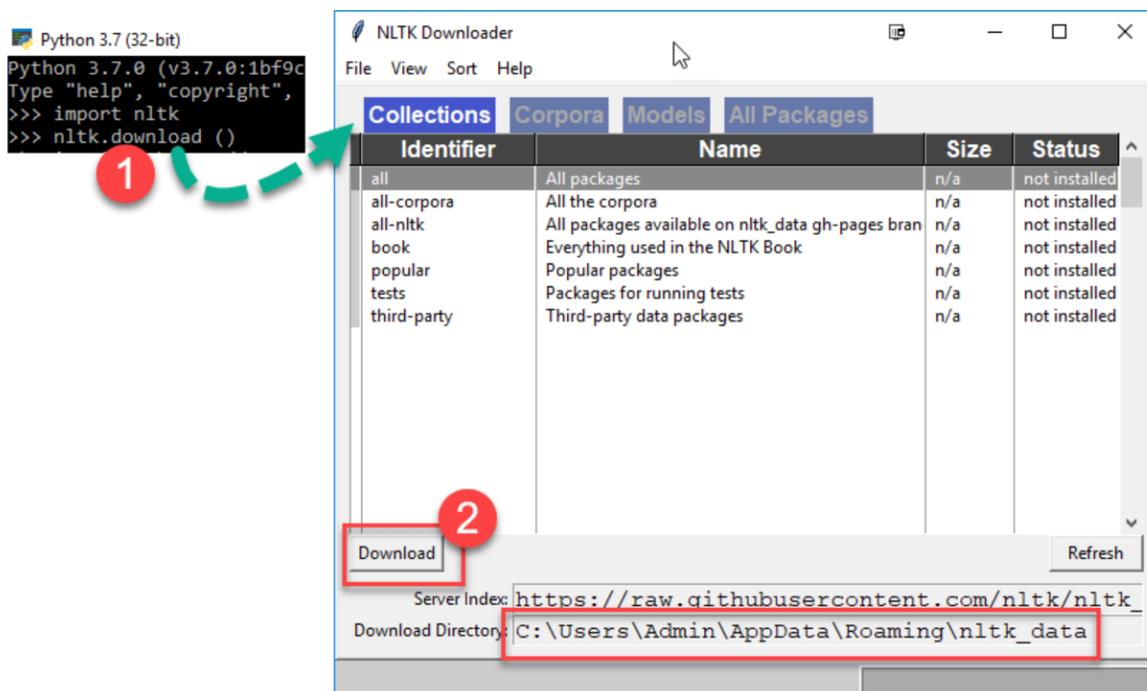


Fig. B. 1 Pasos a seguir para la instalación de todos los paquetes de la biblioteca NLTK

Como se puede observar en la imagen, en el paso 1 se indican los comandos para descargar los paquetes de NLTK, y en el paso 2 se dan a elegir los paquetes que se desean descargar e instalar, se recomienda instalarlos todos si es que no se tiene claridad de cuales se usarán [82]

Anexo C. Funcionalidades del Chatbot

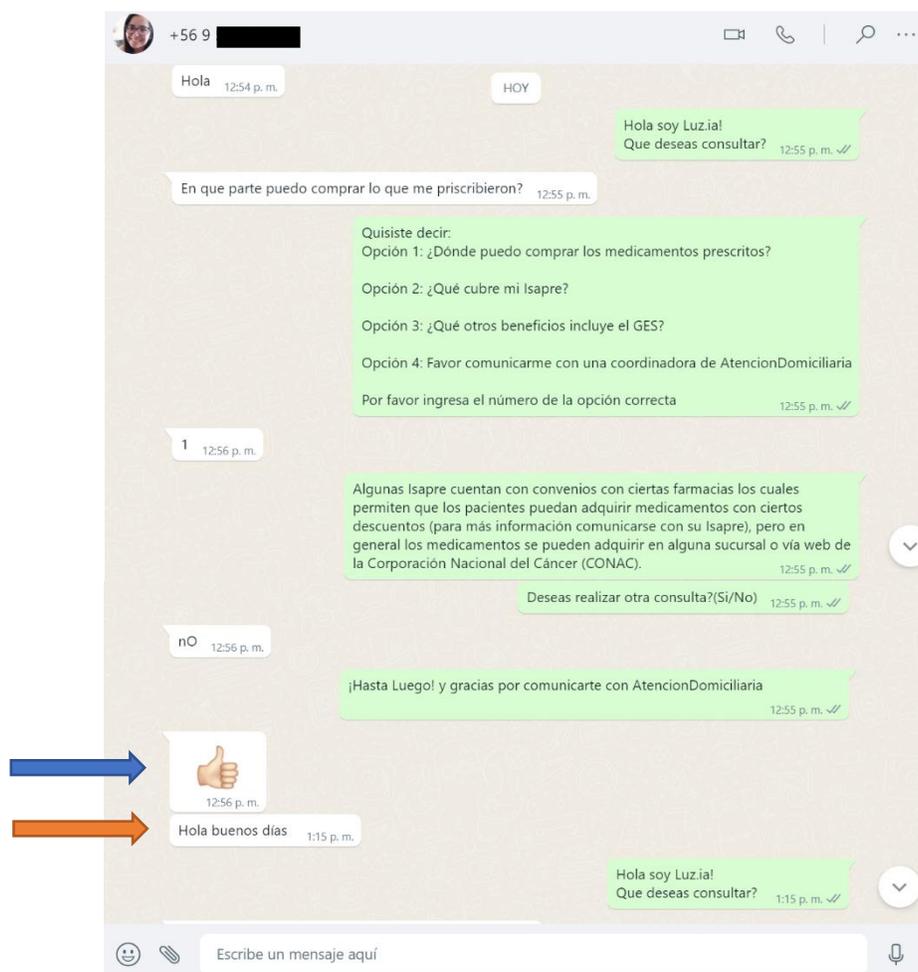


Fig. C. 1 Vista del Chatbot en caso de identificación de despedida/agradecimiento y cuando usuario quiere realizar una consulta

Se puede observar que cuando el usuario envía un emoji como mensaje de despedida el chatbot no responde nada más, ya que lo identifica como despedida/agradecimiento (flecha azul), mientras que cuando el usuario escribe "Hola buenos días" ahí si lo entiende como que quiere realizar una consulta (flecha anaranjada) y envía el saludo de bienvenida



Fig. C. 2 Caso de derivación a coordinador de AtencionDomiciliaria en caso que paciente lo solicite

En la imagen se puede visualizar que el usuario al identificar que lo consultado no corresponde con ninguna opción enviada por el chatbot, solicita la derivación al equipo de coordinación (imagen de la izquierda). Por su parte el chatbot envía un mensaje al equipo coordinador indicando que cierto número esta a la espera para realizar una consulta (imagen de la derecha).

Se identifica un error en el formato solicitado y se solicita ingresarlo de forma correcta

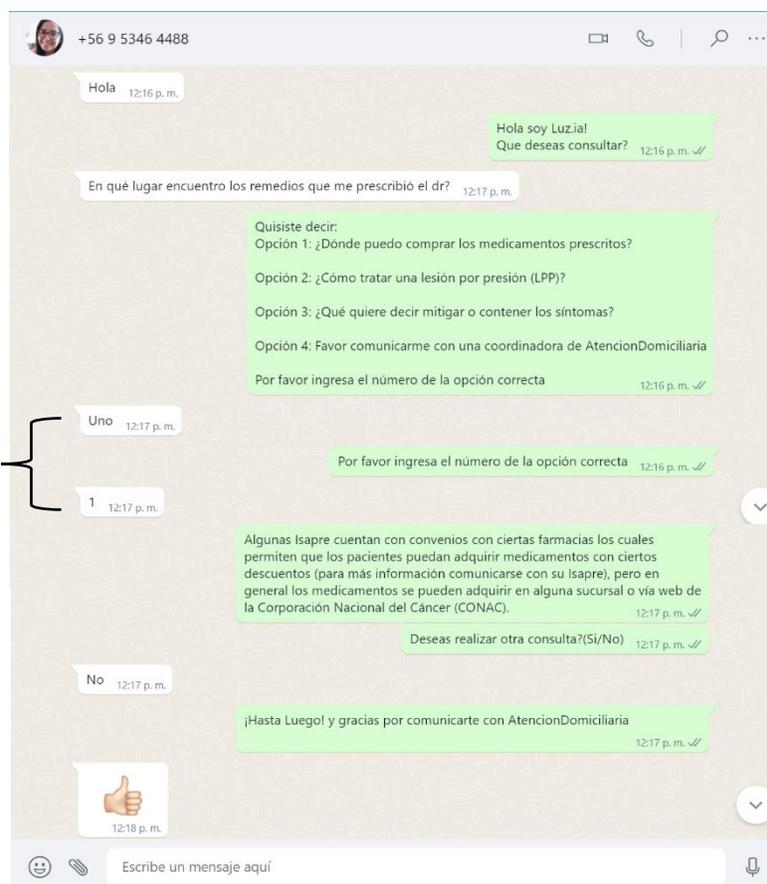


Fig. C. 3 Caso de respuesta mal ingresada por parte del usuario ante solicitud del chatbot

Se puede visualizar que cuando el Chatbot le pide al cuidador/paciente ingresar la opción correcta de acuerdo a lo consultado, y este responde en un formato que no corresponde, se le solicita que lo ingrese nuevamente de forma correcta, y una vez que se responda bien el chatbot envía la respuesta a lo preguntado.

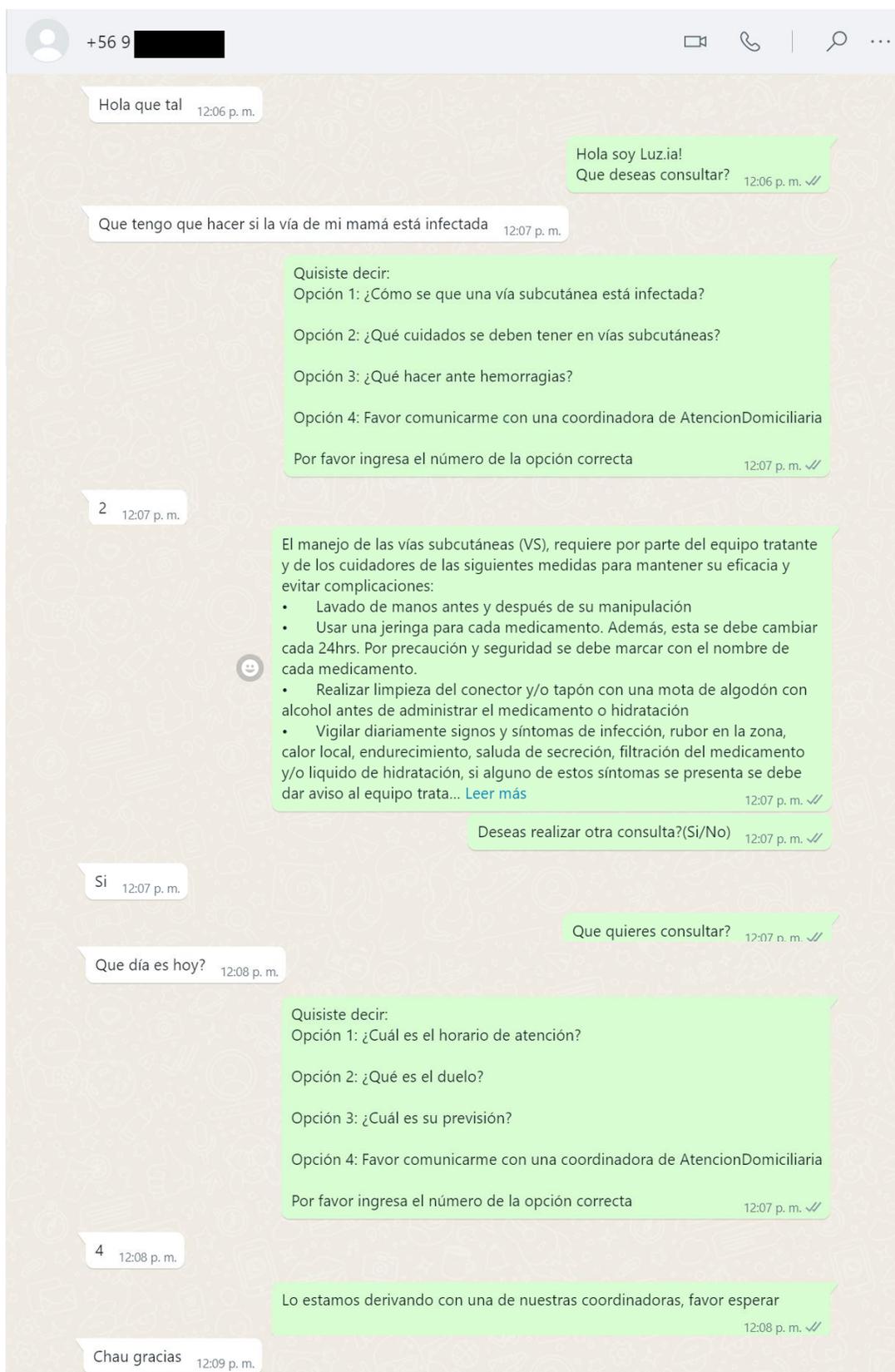


Fig. C. 4 Ejemplo de conversación entre usuario y el chatbot

Anexo D. Tabla de Evaluación Chatbot y Resultados evaluación

Tabla D. 1 Rúbrica de evaluación Chatbot

Pregunta	Modo evaluación
1.- Identifica correctamente la pregunta	1-5
2.- Experiencia de la conversación (las respuestas del chatbot fueron coherentes a lo consultado)	1-5
3.- ¿Le parece correcto el lenguaje utilizado?	1-5
4.- El chatbot es amigable con el paciente/cuidador? (Se da a entender bien, usa terminologías que sean de fácil comprensión, etc.)	1-5
5.- La herramienta es de un fácil uso e intuitivo	1-5
6.- Las instrucciones dadas son claras	1-5
7.- Opiniones Generales (posibles mejoras, que otro tipo de preguntas se pueden incluir, es una herramienta útil, que cosas cambiarías, etc.)	Texto Libre
8.- Evaluación General (Dar un puntaje de acuerdo a su experiencia con el chatbot)	1-5

Tabla D. 2 Valores de la escala utilizada en la rúbrica

Puntuación	Valores de la Escala
1	Totalmente desacuerdo
2	En desacuerdo
3	Neutral
4	De acuerdo
5	Totalmente de acuerdo

Tabla D. 3 Resultados Evaluación Chatbot preguntas con puntuación de 1-5

N° Pregunta	Evaluación 1	Evaluación 2	Evaluación 3	Evaluación 4	Evaluación 5
1	5	4	5	5	5
2	5	4	5	5	5
3	5	4	5	4	4
4	5	4	5	4	5
5	4	4	5	5	5
6	5	4	5	4	5
8	5	4	5	5	5

Tabla D. 4 Resultados Evaluación Chatbot preguntas texto libre

Evaluación 1	Evaluación 2	Evaluación 3	Evaluación 4	Evaluación 5
<p>Luego de hacer el ejercicio de probar el chatbot, encuentro que la herramienta es totalmente amigable con el cuidador ya que utiliza un lenguaje no técnico dándose a entender claramente. En general la persona al otro lado del bot (usuario), tiende a pensar que quien responde es otra persona y no un robot por lo que causaría en cierta medida algo de frustración al estar en un caso de “urgencia”, ya que no contará con la tranquilidad de leer cada una de las opciones e ir recibiendo respuestas tan automáticas por parte del bot. Sin embargo, es una herramienta altamente necesaria para el día a día del cuidador y del equipo de profesionales.</p>	<p>Me parece una buena herramienta que ayuda a mejorar la relación con los pacientes. Es importante como fuente de consulta que permite mejorar los cuidados del enfermo y las dudas que pudieran tener los cuidadores de este.</p>	<p>El chatbot permite de una forma clara y sencilla resolver preguntas repetitivas realizadas por los cuidadores o pacientes. Encuentro que tiene una gran utilidad y aplicabilidad al área de cuidados paliativos. En general además tiene una buena proyección para aplicarlo a otras áreas. Buena herramienta.</p>	<p>Creo que el lenguaje debería ser un poco más amigable. Invitar a seguir conversando, encuentro que “¿Qué quieres preguntar? Es un poco cortante. Y al momento de despedirse, podría identificarse como Luz.ia, aprovechar de solicitar una evaluación y solicitar guardar la conversación de forma anónima para seguir mejorando el chatbot.</p>	<p>El chatbot me parece una muy buena herramienta para los pacientes, responde temas tanto de cuidado como administrativos dando una respuesta a las inquietudes de familiares y pacientes, permite además el contacto con una persona en caso de dificultades mayores, por lo que puede convertirse en apoyo 24 hrs abriendo una oportunidad de asistencia telefónica. Se podría abrir a profesionales dando respuesta a manejo y rotación de opioides, farmacología en general, temas administrativos, manejo de heridas p/e. Acortaría algunos de los textos, dejando 2 a 3 ideas importantes. Las respuestas de procedimientos las apoyaría con imágenes o video. Incluiría preguntas en cuanto a la obtención del certificado de defunción realizaría una respuesta de calificación del usuario al final del contacto (¿Cuánto</p>

le sirvió el
contenido?).

Tabla D. 5 Resultados Promedio de la evaluación del Chatbot

Pregunta	Evaluación Promedio
1.- Identifica correctamente la pregunta	4.8
2.- Experiencia de la conversación (las respuestas del chatbot fueron coherentes a lo consultado)	4.8
3.- ¿Le parece correcto el lenguaje utilizado?	4.4
4.- El chatbot es amigable con el paciente/cuidador? (Se da a entender bien, usa terminologías que sean de fácil comprensión, etc.)	4.6
5.- La herramienta es de un fácil uso e intuitivo	4.6
6.- Las instrucciones dadas son claras	4.6
7.- Opiniones Generales (posibles mejoras, que otro tipo de preguntas se pueden incluir, es una herramienta útil, que cosas cambiarías, etc.)	No aplica
8.- Evaluación General (Dar un puntaje de acuerdo a su experiencia con el chatbot)	4.8

**UNIVERSIDAD DE CONCEPCION – FACULTAD DE INGENIERIA
RESUMEN DE MEMORIA DE TITULO**

Departamento : Departamento de Ingeniería Eléctrica
Carrera : Ingeniería Civil Biomédica
Nombre del memorista : Antonia Belén Baeza Acuña
Título de la memoria : CHATBOT PARA CUIDADOR/FAMILIAR DE
 PACIENTE EN CUIDADOS PALIATIVOS (GES 4)

Fecha de la presentación oral : 25-08-2022

Profesor(es) Guía : Pamela Guevara
Profesor(es) Revisor(es) : Eduardo Wiechmann
Concepto :
Calificación :

Resumen (máximo 200 palabras)

Este trabajo tiene por objetivo el desarrollo e implementación de un chatbot dirigido al cuidador informal de pacientes en GES 4, enfocado en el cuidador. El chatbot permite una comunicación continua, directa y las 24hrs del día entre el centro de salud y el paciente. A lo largo del trabajo se revisó bibliografía relacionada al chatbot, proyectos asociados a los cuidadores informales, además de herramientas que permitieran el desarrollo del proyecto, como modelos de Word-Embeddings o técnicas asociadas al NLP. Dentro de todas las opciones para desarrollar el chatbot, se determinó que *SBERT* era la mejor opción para realizar la incrustación de frases. Después, se utilizan estas incrustaciones para permitir que la maquina entienda las consultas ingresadas, pasando de un lenguaje natural a uno entendido por este. Luego en conjunto a un experto en CP de HealthTracker, se creó una base de datos de preguntas a utilizar, la cual fue integrada al algoritmo. Finalmente se realizaron pruebas por parte del evaluador para corroborar el correcto funcionamiento del chatbot, realizando un total de 5 evaluaciones para probar preguntas escritas correctamente, con errores ortográficos y utilizando palabras inexistentes, en todos los casos obteniendo respuestas coherentes a las preguntas realizadas por el usuario.