Services and Applications for a Healthy Ageing
Servicios y Aplicaciones para Envejecimiento Saludable

SAPIENS

Este proyecto ha sido concedido por la Agencia Estatal de Investigación del Ministerio de Ciencia e Innovación dentro del procedimiento de concesión de ayudas a 'Proyectos de I+D+i' en el marco de los programas estatales de generación de conocimiento y fortalecimiento científico y tecnológico del sistema de I+D+i y de I+D+i orientada a los retos de la sociedad, convocatoria 2020.

MINISTERIO DE CIENCIA E INNOVACIÓN

AGENCIA ESTATAL DE INVESTIGACIÓN

Proyecto de investigación financiado por

MCIN/AEI/10.13039/501100011033

D.3.1: Initial design of conversational agent

WP3: Design and development of an intelligent conversational agent.

UNIVERSIDADE DE VIGO

atlanTTic research center for Telecommunication Technologies

USC UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

1- Introducción 3
2 - Solución adoptada, SKILL 6
2.1. Flujo de trabajo de desarrollo estándar 7
2.2. Aspectos más relevantes a tener en cuenta 8
2.3. Caso de uso 10
2.4. Guía de diseño 12
3 - Resumen 13
4 - Personal participante 14
5 - Referencias 15
6 - Lista de acrónimos 17

1- Introducción

Los altavoces inteligentes son altavoces activados por voz que interactúan con usuarios humanos respondiendo a comandos de voz. Estos dispositivos interactúan con otros elementos del ecosistema del hogar digital y servicios de back-end mediante Bluetooth, NFC, Wi-Fi y otros protocolos inalámbricos. Un altavoz inteligente típico está programado para realizar diferentes tareas en nombre del usuario, como acceder a contenidos de entretenimiento y fuentes de información, enumerar actividades diarias, brindar asistencia con las tareas del hogar, administrar las comunicaciones e interactuar con dispositivos domésticos inteligentes, entre muchas otras. Esta rica cartera de características promovió una demanda creciente sobre otros productos y servicios tecnológicos en el hogar. Los altavoces inteligentes funcionan con un asistente virtual impulsado por inteligencia artificial (IA) con capacidades de reconocimiento de voz, procesamiento del lenguaje natural y síntesis de voz. Como tal, los asistentes virtuales inteligentes pueden simular conversaciones humanas y, por lo tanto, actuar como agentes conversacionales.

Las capacidades informáticas basadas en IA de estos agentes no están alojadas en los propios altavoces inteligentes, sino que las proporcionan sus proveedores en un entorno de computación en la nube según el modelo de software como servicio (SaaS). En otras palabras, cuando el usuario interactúa con un altavoz inteligente mediante comandos de voz o frases, la información de audio capturada se envía a través de la red al back-end de procesamiento y, en algunos casos, se calcula una respuesta adecuada y se envía de vuelta al altavoz para ser pronunciado por él. Además de los altavoces inteligentes, los asistentes virtuales pueden alojarse en la mayoría de los dispositivos inteligentes disponibles en el mercado, como computadoras, teléfonos móviles, televisores, relojes o auriculares.

En la mayoría de los casos, el desarrollo de aplicaciones para dispositivos inteligentes cuenta con el respaldo de sus creadores a través de plataformas de desarrolladores basadas en la nube. Normalmente, a los desarrolladores registrados se les ofrece una colección de herramientas para diseñar, implementar y desplegar sus aplicaciones en un entorno controlado. Este enfoque simplifica y acelera drásticamente el desarrollo de aplicaciones, pero también limita las características y funcionalidades disponibles para quienes administren las herramientas proporcionadas y, en algunos casos, de forma muy restringida. Por ejemplo, no se concede acceso a las señales de voz sin procesar de los usuarios, sino sólo al texto sin formato ya procesado resultante del proceso de conversión de voz a texto realizado en el back-end. Además, registrar datos espontáneos en situaciones del mundo real requiere proteger la privacidad de los involucrados, lo que a su vez implica consideraciones éticas/legales (Wyatt D., Choudhury T.,Bilmes J., 2007).

Independientemente de sus limitaciones, los asistentes inteligentes disponibles en el mercado siguen siendo adecuados para implementar interacciones basadas en conversaciones y, más específicamente, conversaciones o diálogos escritos. En su forma más simple, los chatbots de secuencias de comandos reciben respuestas a una colección de palabras clave, que a su vez se extraen de las declaraciones de los usuarios convertidos en

texto (Adamopoulou, E., Moussiades, L. ,2020). Cada vez que el asistente detecta una palabra clave, se articula la respuesta correspondiente y el asistente espera por una nueva interacción del usuario de acuerdo con el guión. Las versiones más avanzadas se basan en técnicas de IA conversacional (Albayrak, N. et al, 2018), para identificar una amplia colección de solicitudes de usuarios sin recibir capacitación explícita sobre todas las variaciones posibles.

Las conversaciones guiadas desempeñan un papel fundamental en muchos campos de aplicación, como los centros de llamadas (Szymanski, M. H. et al, 2020), los servicios al cliente minorista (Sheehan, B.et al, 2020), la creación rápida de prototipos (Choi, Y. et al, 2021**)**, la capacitación basada en roles (Taubman, B. R. et al, 2020), el manejo del autismo (Grosberg, D. et al, 2017) o el manejo de otros trastornos neuropsiquiátricos (Pacheco-Lorenzo, M. et al, 2021). Como enfoque general en estos.

En las aplicaciones, se genera previamente un árbol de conversación que recoge todas las secuencias de conversación posibles, que a su vez se componen de una sucesión de señales detectadas a partir de las expresiones de los usuarios y las respuestas del asistente. Luego, el usuario o el asistente inicia una conversación, que continuará según el guión hasta llegar a un punto final. Debe tenerse en cuenta que los guiones de conversación pueden incluir ciclos para manejar errores, palabras clave faltantes o no identificadas, o como consecuencia de las características específicas del campo de aplicación.

A pesar de ser conceptualmente simple, implementar conversaciones guiadas mediante asistentes inteligentes disponibles en el mercado se plantean desafíos relevantes entre los que se señalan algunos a continuación:

- No acceso a las señales de voz sin procesar
- Tiempos de espera a que se detecte la respuesta
- Retratos de tiempo

La única información disponible es por tanto el texto que se obtiene después de ser procesado el audio por algoritmos de voz a texto.

Este entregable presenta el diseño inicial de un agente conversacional destinado a evaluar el estado cognitivo y estado emocional de un individuo. Es un enfoque prometedor para diseñar una solución de detección para detectar deterioro cognitivo leve y depresión basada en asistentes de voz inteligentes.

Aunque las soluciones adoptadas en este caso particular son aplicables a cualquier plataforma importante de asistente inteligente, se seleccionó la solución de asistente inteligente Alexa de Amazon como nuestro entorno de desarrollo para esta implementación de prueba de concepto, debido a su popularidad y la disponibilidad de herramientas y soporte para desarrolladores.

A continuación, se da paso en detalle a las características y funcionalidades de la solución implementada que se denomina comúnmente SKILL, sobre la plataforma Alexa.

2 - Solución adoptada, SKILL

En esta sección del entregable se entra en detalle en la descripción del flujo de trabajo para el diseño del desarrollo estándar junto con la explicación de las características comunes de las aplicaciones back-end del asistente inteligente. Además, se especifican los aspectos más relevantes de la implementación de una conversación guiada, como una SKILL en Alexa y por último se abordan los desafíos encontrados.

2.1. Flujo de trabajo de desarrollo estándar

Para la implementación de una aplicación de asistente inteligente basada en conversaciones guiadas deben tenerse en cuenta algunas características específicas que deben considerarse a lo largo de las diferentes fases de desarrollo del software.

Las aplicaciones deben cumplir estándares de disponibilidad, robustez y flexibilidad. Como consecuencia, existen algunas pautas que deben adoptarse desde las primeras etapas de diseño como son:

- Diseño adaptable.
- Definición de diferentes frases o pistas para cada intención.
- Repetir ciclos de ensayo y error para adaptar las interacciones a las respuestas esperadas e inesperadas del usuario.
- Adaptación al reconocimiento de voz.
- Manejo de tiempos de espera activados por silencio.
- Personalización que aporta naturalidad en la conversación.

Así para diferentes perfiles de usuario, se deberán identificar frases específicas para cada perfil (por ejemplo, para distinguir género femenino o masculino). Del mismo modo, las interacciones deben diseñarse teniendo en cuenta la información o restricciones de localización existentes (es el caso de los idiomas, dialectos y variantes lingüísticas).

Las conversaciones escritas pueden ser específicas de un perfil o ubicación determinados, lo que debe tenerse en cuenta en su diseño.

La disponibilidad es otro aspecto clave a la hora de diseñar un agente conversacional. En otras palabras, se debe garantizar que la invocación de la aplicación sea sencilla y pueda lograrse mediante una breve expresión. Además, las tareas deben mantenerse simples y específicas, y deben evitarse interacciones prolongadas y preguntas innecesarias.

Finalmente, un factor crucial es la naturalidad durante la interacción, de forma que la conversación de un asistente inteligente debe percibirse lo más cercana posible a una interacción con una persona real, es decir, debe ser identificable. Para ello, las conversaciones deben redactarse para que suenen claras, concisas y naturales, por ejemplo, introduciendo cierta variedad en las interacciones vocales.

2.2. Aspectos más relevantes a tener en cuenta

Las aplicaciones para asistentes inteligentes comerciales se implementan como servicios remotos basados en la nube. Por lo tanto, crear un agente conversacional que implemente un modelo de interacción de voz específico requiere un punto final accesible a Internet para alojar ese servicio basado en la nube. Como enfoque general, los agentes conversacionales eventualmente se implementarán como servicios web o procedimientos remotos, según el proveedor. Los altavoces inteligentes invocarán una aplicación específica activada por un comando de voz específico de una colección de comandos de voz equivalentes. Los asistentes inteligentes disponibles en el mercado reaccionan a determinadas palabras clave de voz (por ejemplo, Hey Google en el caso de Google Home o Alexa en el caso de Amazon Alexa) seguidas de un comando de invocación de la aplicación. Solo cuando se activa, los datos se envían a la nube para ser procesados por la lógica de la aplicación (ver Fig. 1).

Como opción más sencilla, Google y Amazon ofrecen opciones alojadas en el Asistente de Google o en Alexa para crear, almacenar y alojar aplicaciones de altavoces inteligentes y sus recursos en sus respectivos entornos de nube. Este enfoque simplifica drásticamente el desarrollo de aplicaciones, particularmente en el caso de modelos de interacción simples o populares, a los que se puede acceder como módulos prediseñados en sus respectivos entornos de desarrollo.

En general, estas aplicaciones están controladas por eventos y siguen un modelo sincrónico de llamada-respuesta. En otras palabras, las aplicaciones de asistente inteligente pueden verse como una colección de controladores de eventos, cada uno de los cuales procesa la invocación de un usuario específico representado como una cadena de texto, para generar una respuesta adecuada, también como una cadena de texto.

Las aplicaciones de altavoces inteligentes se pueden desarrollar en una amplia variedad de entornos de desarrollo de software y lenguajes de programación. Por ejemplo, los desarrolladores podrán implementar sus modelos de interacción en Node.js, Java, Python, C# o Go en el caso de Alexa, y Node.js, Go, C++ o Java en el caso de Google Assistant, entre otros.

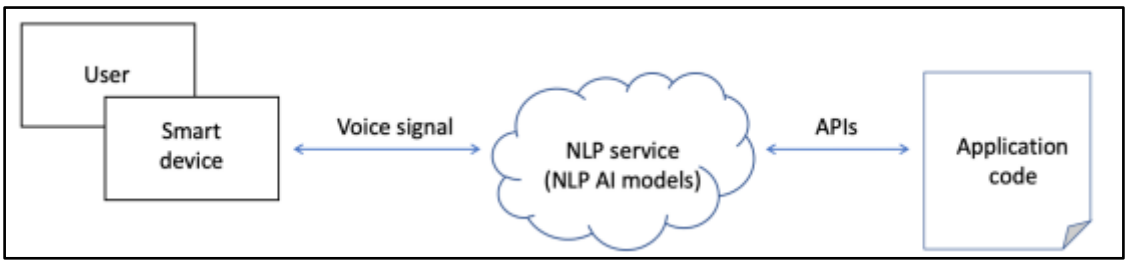


Figura 1. Modelo de implementación de agente conversacional. El dispositivo inteligente se ocupa exclusivamente de las señales de voz. El código de la aplicación interactúa exclusivamente con los servicios en la nube de PNL mediante cadenas de caracteres.

Los asistentes inteligentes y otras soluciones conversacionales habilitadas por agentes simplemente recopilan las expresiones de voz del servidor para enviarlas a servicios

de procesamiento del lenguaje natural (PLN) basados en la nube y reproducir las señales de voz generadas por dichos servicios. Los servicios de PLN son básicamente modelos de IA que aprenden continuamente para mejorar su eficacia. El código de la aplicación recibe solo el texto sin formato generado por los módulos de reconocimiento de voz en el servicio PLN y no tiene acceso a la señal de audio original. De la misma manera, el código de la aplicación puede generar flujos de texto como respuesta a eventos, que a su vez serían convertidos en señales de voz mediante los módulos de síntesis de voz en el servicio PLN.

2.3. Caso de uso

En este apartado del entregable se expone el caso de uso, con el diseño de la implementación como una SKILL (habilidad) de Alexa, es decir, como una aplicación que se puede invocar mediante un altavoz inteligente de Amazon o cualquier dispositivo que implemente la interfaz del asistente inteligente de Alexa. Así, el principal requisito que debe cumplir la implementación ideada es que debe mantener las capacidades de discriminación. En otras palabras, la conversación guiada implementada debería ser indistinguible de la prueba telefónica original.

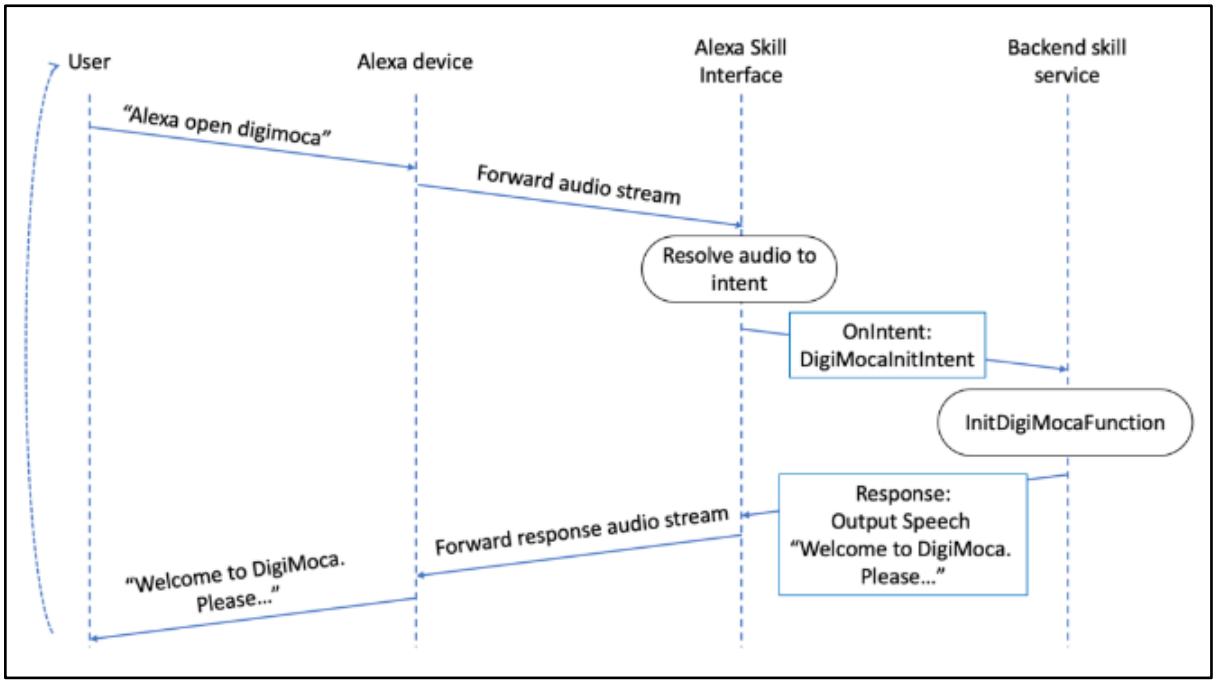


Figura 2. Flujo de trabajo interactivo de la SKILL Alexa.

Para la implementación se han de tener en cuenta los siguientes pasos:

Primero, se define la invocación de la SKILL que implementa.

Segundo, la identificación de todas las intenciones relevantes y su mapeo en las expresiones habladas.

Una intención representa un comportamiento que satisface la solicitud expresada por un usuario. Los comportamientos pueden incluir, por ejemplo, interactuar con otros módulos de software, actualizar elementos de datos locales o remotos o sintetizar una respuesta expresada. Por lo tanto, en esta fase, se especifica un conjunto de expresiones de muestra que se asignan a intenciones específicas. Como se muestra en la (ver Fig. 2), cada vez que el usuario produce una de las expresiones definidas, las palabras habladas se resuelven con una intención específica, que a su vez es implementada y manejada por el servicio en la nube de Alexa.

En nuestro caso particular, para cada pregunta u oración en la conversación escrita que producirá el agente conversacional, tuvieron que asignar e identificar todas las

respuestas probables del usuario. Estos corresponden a las expresiones de muestra analizadas anteriormente que serán manejadas por el servicio de nube de back-end. Por ejemplo, para cada ítem, se deben identificar de antemano todas las respuestas posibles del sujeto al que se le realiza la prueba de deterioro cognitivo.

Tercero, la implementación real de la SKILL. En este caso, se seleccionó Python como lenguaje de programación.

El código de SKILL estará alojado en la nube de Amazon y ejecutado por el servicio AWS Lambda, el servicio de computación en la nube sin servidor de Amazon. El concepto de computación sin servidor (Baldini, I. et al, 2017), se refiere a un modelo de computación en la nube que no requiere mantener servidores específicos para ejecutar un servicio. AWS Lambda es un servicio totalmente administrado que se ocupa de todas las necesidades de infraestructura.

2.4. Guía de diseño

En este apartado del entregable se describen los aspectos más relevantes que se deben tener en cuenta en el diseño del agente para dar soporte a conversaciones guiadas.

A. Definir el nombre de invocación de la SKILL.

   Esta es la palabra o frase que el usuario debe pronunciar para que Alexa lance la SKILL. Debe definirse como una cadena de caracteres en minúscula y no puede contener ninguna de las frases de inicio estándar de Alexa Skill (por ejemplo, iniciar, abrir, cargar, comenzar, etc.), ninguna de las palabras de activación predefinidas (como Alexa, Amazon, etc.) o preposiciones.

B. Identificar intenciones relevantes.

   Además de las intenciones específicas de una aplicación, todas las SKILLs de Alexa están equipadas con cuatro intenciones básicas: Cancelar, Ayuda, Detener y Navegar a casa. Los primeros tres brindan instrucciones de uso y soporte para abandonar una SKILL, mientras que Navegar a casa, solo se usa cuando se desarrollan SKILLs complejas que involucran múltiples pasos. Los desarrolladores también pueden agregar intenciones integradas (es decir, predefinidas) y personalizadas. Las intenciones se recopilan en un archivo de formato de intercambio de datos abierto de notación de objetos JavaScript (JSON).

C. Implementar la SKILL.

   El kit de SKILLs de Alexa, es un marco de desarrollo de software que admite la implementación de SKILLs en Node.js, Java y Python, se utiliza para construir SKILLs de Alexa. Las SKILLs a menudo dependen de la conectividad a Internet, y las interrupciones en la conexión pueden afectar negativamente tanto a la experiencia de usuario como a la depuración y mantenimiento del código.

3 - Resumen

Este entregable analiza el diseño de la implementación de agentes conversacionales inteligentes que utilizan dispositivos inteligentes disponibles en el mercado. Este enfoque se ilustra con la implementación de una prueba estandarizada basada en conversaciones para la detección de trastornos cognitivos, que normalmente se diagnostican mediante pruebas de lápiz y papel.

Desde el punto de vista de la ingeniería de software, los asistentes inteligentes son básicamente interfaces basadas en voz para aplicaciones de software basadas en eventos. Los servicios de PNL basados en la nube convierten las interacciones de voz de los usuarios en texto y viceversa. En el back-end, las aplicaciones de software basadas en eventos reaccionan a señales específicas o desencadenan eventos realizando actividades definidas por la lógica de negocios, como reproducir música, actualizar un calendario, iniciar otras aplicaciones o realizar alguna tarea en nombre del usuario.

La lógica de manejo de eventos en el back-end puede ser tan simple como registrar la intervención expresada por el usuario, pero también puede implicar una toma de decisiones compleja para guiar la conversación programada para que evolucione de diferentes maneras. Al utilizar el almacenamiento de back-end, se puede utilizar información sobre interacciones anteriores para impulsar la conversación por diferentes caminos y también para proporcionar persistencia para permitir que se reanuden las conversaciones interrumpidas.

Por otro lado, desde un punto de vista neuropsicológico, este entregable sienta las bases para el diseño a la hora de digitalizar y adaptar a un agente conversacional pruebas clásicas de lápiz y papel afectadas por variables de confusión.

Además de la evaluación cognitiva, el uso de cuestionarios como herramienta de evaluación, cribado o diagnóstico es una práctica habitual en otros campos como la neuropsiquiatría, las ciencias sociales o la psicología. En consecuencia, una aproximación exitosa a la introducción de agentes conversacionales en estos campos puede suponer un avance relevante en un amplio abanico de escenarios.

4 - Personal participante

- Investigadores Principales (IPs)
  - Luis Eulogio Anido Rifón (Uvigo)
  - Luis Modesto Álvarez Sabucedo (Uvigo)

- Equipo de Investigación (EI)
  - Manuel Fernández Iglesias (Uvigo)
  - Martín Llamas Nistal (Uvigo)
  - Manuel Gandoy Crego (USC)
  - Marcos Pazos Couselo (USC)
  - Manuel Caeiro Rodríguez (Uvigo)
  - Juan Manuel Santos Gago (Uvigo)
  - Fernando Ariel Mikic Fonte (Uvigo)

- Equipo de Trabajo (ET)
  - Alba Elena Martínez Santos (USC)
  - César Bugallo Carrera (USC)
  - Sonia María Valladares Rodríguez (USC)
  - Martín Liz Domínguez (Uvigo)
  - Moises Ruben Pacheco Lorenzo (Uvigo)

- Personal Contratado (PC)
  - Iván Otero González

5 - Referencias

Adamopoulou E, Moussiades L. An Overview of Chatbot Technology. In: Maglogiannis I, Iliadis L, Pimenidis E. , eds. Artificial Intelligence Applications and Innovations. IFIP WG 12.5. Springer; 2020: 373–383 https://link.springer.com/chapter/10.1007/978-3-030-49186-4_31?ref=blog.min.io

Albayrak N, Özdemir A, Zeydan E. An overview of artificial intelligence based chatbots and an example chatbot applica-tion. In: . ., ed. Procs. of the 26th Signal Processing and Communications Applications Conference (SIU). IEEE. Curran Associates Proceedings; 2018: 1130-1134 https://ieeexplore.ieee.org/abstract/document/8404430

Baldini I, Castro P, Chang K, et al. Serverless Computing: Current Trends and Open Problems. In: Chaudhary S, Somani G, Buyya R., eds. Research Advances in Cloud Computing. Singapore: Springer Singapore. 2017 (pp. 1–20) https://link.springer.com/chapter/10.1007/978-981-10-5026-8_1

Choi Y, Monserrat TJKP, Park J, Shin H, Lee N, Kim J. ProtoChat: Supporting the Conversation Design Process with Crowd Feedback. Proc. ACM Hum.-Comput. Interact. 2021; 4(CSCW3). doi: 10.1145/3432924 https://dl.acm.org/doi/abs/10.1145/3432924

Daubman BR, Bernacki R, Stoltenberg M, Wilson E, Jacobsen J. Best Practices for Teaching Clinicians to Use a Serious Illness Conversation Guide. Palliative Medicine Reports 2020; 1(1): 135-142. doi: 10.1089/pmr.2020.0066 https://www.liebertpub.com/doi/full/10.1089/pmr.2020.0066

Grosberg D, Charlop MH. Teaching conversational speech to children with autism spectrum disorder using text-message prompting. Journal of applied behavior analysis 2017; 50(4): 789—804. doi: 10.1002/jaba.403 https://onlinelibrary.wiley.com/doi/abs/10.1002/jaba.403

Pacheco-Lorenzo M, Fernández-Iglesias MJ, Valladares-Rodriguez S, Anido-Rifón LE. Implementing scripted conversations by means of smart assistants. Softw Pract Exper. 2023; 53(5): 1271–1283. doi:10.1002/spe.3182. https://doi.org/10.1002/spe.3182

Pacheco-Lorenzo MR, Valladares-Rodríguez SM, Anido-Rifón LE, Fernández-Iglesias MJ. Smart conversational agents for the detection of neuropsychiatric disorders: A systematic review. Journal of Biomedical Informatics 2021; 113: 103632. doi: https://doi.org/10.1016/j.jbi.2020.103632

Sheehan B, Jin HS, Gottlieb U. Customer service chatbots: Anthropomorphism and adoption. Journal of Business Research 2020; 115: 14-24. doi: https://doi.org/10.1016/j.jbusres.2020.04.030

Szymanski MH, Wall P, Watts-Englert J. Creating interactional alignment in call center customer care. In: Waal Malefyt dT, McCabe M., eds. Women, Consumption and Paradox. Routledge. 2020 (pp. 78-102)

Wyatt D, Choudhury T, Bilmes J. Conversation Detection and Speaker Segmentation in Privacy Sensitive Situated Speech Data. In: Hamme vH, Son vR., eds. Proceedings of Interspeech 2007. ISCA. 2007: 586–589. https://ai.cs.washington.edu/www/media/papers/tmpNDHZvc.pdf

6 - Lista de acrónimos

AWS	Amazon Web Services
IA	Inteligencia Artificial
JSON	JavaScript Object Notation
NFC	Near Field Communication
PLN	Procesamiento del Lenguaje Natural
SaaS	Software as a Service
Wi-Fi	Wireless Fidelity