

---

# Chatbot para la recuperación de información personal

---



Trabajo de Fin de Grado  
Curso 2021–2022

Autora  
Lucía Latorre Magaz

Directores  
Gonzalo Méndez Pozo  
Pablo Gervás Gómez-Navarro

Grado en Ingeniería Informática  
Facultad de Informática  
Universidad Complutense de Madrid



# Chatbot para la recuperación de información personal

Trabajo de Fin de Grado en Ingeniería Informática  
Departamento de Ingeniería del Software e Inteligencia  
Artificial

**Autora**  
Lucía Latorre Magaz

**Directores**  
Gonzalo Méndez Pozo  
Pablo Gervás Gómez-Navarro

*Dirigida por el Doctor*  
Gonzalo Méndez Pozo  
Pablo Gervás Gómez-Navarro

Grado en Ingeniería Informática  
Facultad de Informática  
Universidad Complutense de Madrid

15 de julio de 2022



# Dedicatoria

Texto de la dedicatoria...



# Agradecimientos

Texto de los agradecimientos





# Resumen

Resumen en español del trabajo

## **Palabras clave**

Máximo 10 palabras clave separadas por comas chatbot, reminiscencia, demencia, recuerdo



# Abstract

Abstract in English.

## **Keywords**

10 keywords max., separated by commas.



# Índice

<b>1. Introduction</b>	<b>1</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Motivación . . . . .	3
1.2. Objetivos . . . . .	3
1.3. Plan de trabajo . . . . .	4
<b>2. Estado de la Cuestión</b>	<b>5</b>
2.1. Demencia . . . . .	5
2.2. Terapia ocupacional basada en reminiscencia . . . . .	5
2.3. Proyecto CANTOR . . . . .	6
2.4. Trabajo previo . . . . .	6
2.5. PLN . . . . .	6
2.5.1. Chatbots . . . . .	6
2.5.2. Análisis de sentimiento . . . . .	6
<b>3. Arquitectura</b>	<b>9</b>
3.1. Prototipo primero de análisis de sentimientos de un texto . . .	9
3.2. Módulo primero de encadenamiento entre preguntas y respuestas	10
3.3. Módulo primero de clasificación de respuestas en etapas de vida y en sentimientos . . . . .	11
<b>4. Pruebas</b>	<b>13</b>
4.1. Módulo primero de encadenamiento entre preguntas y respuestas	13
4.2. Módulo primero de clasificación de respuestas en etapas de vida y en sentimientos . . . . .	15
<b>5. Conclusiones y Trabajo Futuro</b>	<b>17</b>
<b>5. Conclusions and Future Work</b>	<b>19</b>



# Índice de figuras





# Índice de tablas



# Chapter 1

## Introduction

Introduction to the subject area.



# Capítulo 1

## Introducción

*“Frase célebre dicha por alguien inteligente”*

— Autor

### Introducción temporal

Las personas con Alzheimer u otros tipos de demencia pueden beneficiarse del uso de la llamada terapia basada en reminiscencia, que se basa en la construcción de un libro de vida del paciente que recopila recuerdos positivos de su vida que se pueden utilizar posteriormente par ejercitar su memoria y retrasar el deterioro, además de permitir aumentar el bienestar de los pacientes.

En el presente proyecto se propone el desarrollo de un chatbot que permita recopilar y estructurar esta información para ayudar a los terapeutas en la elaboración de los libros de vida.

### 1.1. Motivación

Introducción al tema del TFG.

### 1.2. Objetivos

Este proyecto tiene como objetivo desarrollar un chatbot mediante el cual recabar información personal sobre la vida del paciente con demencia, clasificarla y estructurarla siguiendo un esquema que pueda facilitar la tarea de los terapeutas a la hora de construir un libro de vida.

La clasificación de recuerdos se hará en base a unos criterios predefinidos por expertos en terapia ocupacional del proyecto CANTOR. Se clasificará en función de:

- **Emoción:** Se clasificarán los recuerdos en positivos y negativos, siendo estos últimos para identificar qué recuerdos no deben tratarse en las

terapias por afligir al paciente. Los positivos se puntuarán de 0 a 10 en función de la felicidad que le traen al paciente.

- **Etapas:** Los recuerdos pertenecerán a una de las siguientes etapas: infancia, adolescencia, edad adulta o tercera edad según el periodo temporal en que aconteció.
- **Categorías:** Cada recuerdo entrará dentro de una o varias categorías que recojan una característica del recuerdo. Ejemplos de categorías: guerra civil, bailes, ocio, familia, aficiones...

### 1.3. Plan de trabajo

El plan de trabajo ha consistido de tres etapas:

- Investigación y construcción de prototipo en la que se ha consolidado la idea del TFG, investigado sobre demencia y terapia ocupacional basada en reminiscencia, elegido tecnologías y creado un prototipo de análisis de texto usando spaCy para identificar si un recuerdo es positivo o negativo.
- Programación del Chatbot y desarrollo de la memoria
- Pruebas, revisión de la memoria y entrega

## Capítulo 2

# Estado de la Cuestión

Introducción de lo que voy a hablar en el estado de la cuestión y por qué

### 2.1. Demencia

La demencia (dem) es una condición neurodegenerativa con tendencia a crecer, caracterizada por un deterioro cognitivo que interfiere con la vida cotidiana afectando a la memoria, al pensamiento, al lenguaje, al juicio y al comportamiento. La demencia no es una enfermedad específica aunque la mayor parte de los casos de demencia son provocados por la enfermedad de Alzheimer. Muchas veces se confunde la demencia con una consecuencia más del envejecimiento cuando no tiene por qué ser así.

Hay muchos síntomas asociados a la demencia pero, en este trabajo, nos vamos a centrar en la pérdida de memoria. Trabajar los recuerdos de una persona que sufre demencia ayuda a retrasar los efectos de la demencia. Hablaremos para ello de la terapia ocupacional basada en reminiscencia.

### 2.2. Terapia ocupacional basada en reminiscencia

La terapia ocupacional (ter) se centra en que el paciente sea capaz de participar en las actividades de la vida cotidiana. Es decir, se basa en ayudar al individuo a llevar una vida lo más normal posible adaptando las tareas cotidianas a realizar o el entorno para que pueda llevarlas a cabo.

(Sacar información del seminario cantor) La terapia ocupacional basada en reminiscencia se centra en mejorar la calidad de vida de la persona con demencia. Se trata de una técnica basada en la recuperación de recuerdos dentro de un periodo de tiempo en la vida de la persona con el objetivo de construir la historia de vida del sujeto. La historia de vida surge de la sucesión de acontecimientos que componen la totalidad de la vivencia del

sujeto.

## 2.3. Proyecto CANTOR

(can)

## 2.4. Trabajo previo

Herramienta de ayuda guiada para la reminiscencia (rem) : Generación de historias a partir de una base de conocimiento; recomendación de temas a tratar en la terapia + aplicación web que enlaza situaciones y vivencias mediante grafos y luego permite añadir recursos fotográficos asociado a un tema. (Más como un chatbot que va sugiriendo temas a tratar)

Sistema de asistencia para cuidados de enfermos del Alzheimer (asi) : Página que guarda información sobre pacientes y terapeutas asociados. Información relevante + historia de vida formada por instancias de recuerdos

Extracción de preguntas a partir de imágenes para personas con problemas de memoria mediante técnicas de Deep Learning (pre) : chat desplegado con telegram. De las fotos que tiene archivadas va preguntando al usuario cosas relacionadas con la imagen

Generación de resúmenes de video-entrevistas utilizando redes neuronales (res) : transcripción de video-entrevistas a texto

Extracción de información personal a partir de redes sociales para la creación de un libro de vida (rrs)

## 2.5. PLN

El procesamiento de lenguaje natural

### 2.5.1. Chatbots

### 2.5.2. Análisis de sentimiento

El análisis de sentimiento es un método para identificar las emociones que se esconden tras un mensaje concreto y forma parte del procesamiento de lenguaje natural (PLN). Consiste en analizar las frases para extraer de ellas las opiniones o sentimientos acerca de un tema o producto.

Con este análisis se pretende determinar quién es el sujeto del sentimiento, sobre qué o quién tiene ese sentimiento y categorizar esos sentimientos como positivos, negativos o neutros.

Tras realizar el análisis del sentimiento se puede averiguar qué se esconde detrás de información subjetiva.



La aplicación de esta tecnología está enfocada al marketing, de forma que permite a las empresas averiguar qué es lo que quieren sus consumidores mediante el escrutinio de opiniones en redes sociales u otros medios.

Estos sistemas tienen limitaciones, ya que no pueden detectar toda la complejidad del lenguaje humano, la carencia de comprensión del contexto o dificultad para entender la ironía o sarcasmo.

Las principales herramientas para el análisis del sentimiento son:

- Lingmotif → Se trata de una herramienta de análisis de sentimiento desarrollada por la universidad de Málaga. Permite obtener valores precisos de las opiniones y sentimientos dentro de un texto.
- Opinion Finder → Sistema desarrollado por investigadores de la Universidad de Pittsburgh, Cornell y Utah. Permite identificar la subjetividad de frases y varios aspectos de la subjetividad dentro de las propias frases mediante el procesamiento de documentos. Funciona en Inglés.
- LIWC → “Linguistic Inquiry and Word Count” es un programa capaz de analizar textos para calcular el uso que hacen las personas de distintas categorías de palabras. Permite saber si los emisores transmiten un mensaje con palabras positivas o negativas entre otras muchas opciones.

<https://gaeapeople.com/marketing-estrategia/sentiment-analysis/> ->informacion de analisis de sentimientos <https://blog.pangeanic.es/funcionamiento-herramientas-analisis-sentimiento-basadas-en-inteligencia-artificial-> ->mas info <https://ltl.uma.es> ->lingmotif <https://mpqa.cs.pitt.edu/opinionfinder/> ->opinion finder <https://www.liwc.app> ->LIWC



# Capítulo 3

## Arquitectura

### 3.1. Prototipo primero de análisis de sentimientos de un texto

Se ha probado una forma de clasificar texto utilizando `TextCategorizer` (`cat`) de la librería `spacy` de python. Se trata de entrenar un modelo para saber identificar si un recuerdo es positivo o negativo. Se ha utilizado el código del artículo `pro` para probar cómo se podría entrenar el modelo para que supiese diferenciar los recuerdos negativos de los positivos y así tener la primera clasificación que pidieron los terapeutas especializados del proyecto CANTOR. Se ha probado usando una batería de recuerdos positivos encontrados entre los archivos del TFG de Laura Castillo (`rem`) y negativos inventados. Antes de probar con recuerdos se probó con un dataset mucho más grande de reseñas de moda que se ponía como ejemplo en `pro`.

El prototipo funciona siguiendo una serie de pasos. En primer lugar, se añade el componente `TextCategorizer` (`textcat`) a un modelo en blanco del idioma español. Un modelo en blanco es un modelo que no tiene ningún componente de `spaCy` definido (como serían NER que explicaré más adelante), es decir, el texto que se almacena en los documentos de `spaCy` no es analizado por ninguna cadena de procesos porque la tubería de componentes está vacía. Si hubiésemos cogido un modelo pre-entrenado como “`es_core_news_sm`”, el clasificador de textos se sumaría al trabajo previo de los procesos que analizan el texto como serían `['tok2vec', 'morphologizer', 'parser', 'attribute_ruler', 'lemmatizer', 'ner']`. Por ejemplo, el componente NER, más reconocido como *Named Entity Recognizer*, reconoce entidades dentro del texto como nombres de personas, fechas o lugares. Al usar el modelo en blanco empezamos con el español de cero, sin analizar. Tras añadir el componente `textcat`, al crear un documento de `spaCy`, automáticamente, pasará por el proceso de clasificación que le indiquemos. Para que `textcat`

funcione como categorizador de recuerdos, se le añade a nuestro nuevo componente dos etiquetas, NEGATIVO y POSITIVO, que definirán cuánto de negativo es un recuerdo y cuánto de positivo. Para seguir configurándolo, el prototipo coge el texto que usaremos para entrenar nuestro modelo y lo prepara adecuándolo al formato que entiende el clasificador, siendo éste una lista de tuplas (texto, etiqueta). Además, cogerá un porcentaje de esta batería de recuerdos (% que previamente hemos definido) y lo reservará para la evaluación de las predicciones, es decir, para analizar cómo de bien predice nuestro modelo tras entrenarlo.

Ya tenemos casi todo preparado para comenzar a entrenar el modelo, se dividen los casos de entrenamiento en lotes que se analizarán y evaluarán un número definido de iteraciones para asegurar que el entrenamiento es lo más preciso posible sin pasarnos de vueltas para que sea óptimo. Se trata de un proceso iterativo en el que las predicciones del modelo se comparan con las etiquetas de referencia para estimar el gradiente de la pérdida. El modelo se entrena utilizando una función que lo analiza y actualiza en cada iteración, la función `update()`. También se comprueban las predicciones del modelo en cada vuelta, se comparan con las etiquetas de referencia para estimar la desviación de la pérdida.

Una vez afinado el modelo mediante el entreno previo, ya podemos ponerlo a prueba. El prototipo sacará la probabilidad de que un texto procesado por `spacy` sea un recuerdo positivo y la probabilidad de que sea uno negativo.

### 3.2. Módulo primero de encadenamiento entre preguntas y respuestas

Esta sección consiste en conseguir que las preguntas que se hagan al interrogado tengan sentido con el resto de la conversación previa. Conseguir que sean lo más inteligentes y adecuadas posible.

Lo primero que se ha hecho para encontrar la siguiente pregunta a hacer es coger la lista de todas las posibles preguntas y eliminar aquellas que ya han sido contestadas anteriormente para no repetirlas.

Lo siguiente que se hará es pasar tanto la respuesta más reciente que ha dado el interrogado como todas las posibles preguntas por un proceso de síntesis. Para ello vamos a utilizar el analizador de texto `Spacy` utilizando el código del TFG de Laura Castilla Castellano (Generación de historias a partir de una base de conocimiento), en concreto el código que aparece en el archivo “`analysis.py`” de su código, que contiene funciones para el procesamiento y síntesis de los textos. Los pasos que Laura sigue para el análisis del texto aparecen en el apartado 5.1 (Analizar textos para obtener sugerencias) de su memoria, páginas 30 y 31. Las funciones que se han reutilizado en este trabajo, con algunas modificaciones, son las siguientes:

- Read: Coge el texto a analizar y elimina signos de puntuación
- Synthesis: Elimina las palabras vacías como las preposiciones del texto
- Lemmatize: Transforma cada palabra del texto sus lemas correspondientes
- Categorize: Separa las palabras en sustantivos, verbos y adjetivos. Además, coge los 30 más comunes de cada tipo
- Get\_most\_common\_words: Coge las palabras más comunes de todos los tipos del texto

Estas funciones se han utilizado para conseguir una representación del texto más precisa y reducida. Solo los lemas de las palabras importantes del texto serán usados para este módulo de encadenamiento de preguntas y respuestas.

Por último, dada la respuesta y todas las posibles preguntas ya reducidas a sus lemas, para encontrar la pregunta más adecuada, se compara una a una los lemas de las posibles preguntas con los lemas de la respuesta. Dentro de la lista de posibles preguntas, la que más lemas comunes tenga con la respuesta, gana como siguiente pregunta a formular.

### **3.3. Módulo primero de clasificación de respuestas en etapas de vida y en sentimientos**

Para categorizar las respuestas según el sentimiento usaremos directamente el prototipo del apartado de análisis de sentimientos de un texto. Se ha añadido un archivo “analyze\_answer” que entrena primero al modelo con textos positivos y negativos y luego evalúa el texto que le llega e identifica si es negativo o positivo. Esto se añade como una categoría que se meterá en la base de datos junto a la respuesta. Este análisis de sentimiento también servirá para identificar temáticas dolorosas para el interrogado que no deberían ser usadas en las terapias.

Otra categoría que se añadirá junto a la respuesta será la de la etapa de la vida a la que corresponde. Primero tenemos que definir esas etapas y para ello me he basado en la clasificación que hace el Ministerio de Salud de Colombia (<https://www.minsalud.gov.co/proteccionsocial/Paginas/cicloVida.aspx>). Le he hecho algunas modificaciones para adecuarle más a lo que quería sacar en claro de la información que se me presentaba. Las etapas serían las siguientes:

- Infancia de 0 a 11 años
- Adolescencia de 12 a 17 años
- Juventud de 18 a 26 años

- Etapa adulta de 27 a 59 años
- Vejez de 60 años o más
- Indeterminado para aquellos textos en los que no se pueda distinguir la etapa

Una vez elegidos los periodos de tiempo en los que se divide la vida de las personas, el objetivo es clasificar recuerdos, dados en forma de respuesta a una pregunta, según la etapa de la vida a la que pertenecía el recuerdo de la persona interrogada. Para ello, también se ha utilizado la misma tecnología que el prototipo primero de análisis de sentimientos. Además de entrenarse en recuerdos positivos y negativos, ahora el modelo se entrena para distinguir etapas vitales en las que ocurren los recuerdos para así, de cara a la elaboración de un libro de vida, que toda la información esté bien estructurada en periodos de tiempo.

# Capítulo 4

## Pruebas

### 4.1. Módulo primero de encadenamiento entre preguntas y respuestas

Veamos algunos ejemplos de cómo funciona el primer acercamiento a un chatbot inteligente que pregunta con algo de sentido. A continuación se muestra la primera pregunta y la respuesta del interrogado:

IA: ¿Quiénes son las personas más importantes de tu vida?

Tú: Mi familia y concretamente mis padres y mi hermano

En la siguiente log se muestra el análisis que hace el módulo de cálculo de la mejor siguiente pregunta:

```
¿Hay algún momento que te gustaría volver a vivir?
```

```
0
```

```
¿Cuál es tu sexo?
```

```
[[momento, gustaría, volver, vivir]]
```

```
{'familia', 'padre', 'hermano'}
```

```
{'volver', 'vivir', 'momento', 'gustar'}
```

```
-----
```

```
¿Dónde viviste cuando eras pequeño?
```

```
0
```

```
¿Cuál es tu sexo?
```

```
[[viviste, pequeño]]
```

```
{'familia', 'padre', 'hermano'}
```

```
{'pequeño', 'vivistar'}
```

```
-----
```

```
¿Has vivido en algún lugar diferente cuando eras pequeño?
```

```
0
```

```

¿Cuál es tu sexo?
[[has, vivido, lugar, pequeño]]
{'familia', 'padre', 'hermano'}
{'lugar', 'pequeño', 'vivir'}
-----
¿Cómo se llaman tus padres?
0
¿Cuál es tu sexo?
[[llaman, padres]]
{'familia', 'padre', 'hermano'}
{'padre', 'llamar'}
-----
¿Si tienes hermanos, Cómo se llaman?
1
¿Cómo se llaman tus padres?
[[tienes, hermanos, llaman]]
{'familia', 'padre', 'hermano'}
{'llamar', 'tener', 'hermano'}
-----
¿Cómo era la casa dónde viviste de pequeño?
1
¿Cómo se llaman tus padres?
[[casa, viviste, pequeño]]
{'familia', 'padre', 'hermano'}
{'pequeño', 'casa', 'vivistar'}

```

En la imagen podemos observar el análisis de varias posibles preguntas. La separación entre preguntas se marca con una fila de guiones. En cada apartado nos encontramos con la posible pregunta, después el número de lemas que han coincidido en preguntas anteriores, la pregunta que hasta el momento va ganando como candidata a ser preguntada, las palabras más relevantes de la posible pregunta entre corchetes, los lemas de la respuesta entre llaves y los lemas de la posible pregunta entre llaves. Las tres primeras preguntas no serán elegidas como candidatas a ser preguntadas porque no coincide ningún lema de la respuesta con los lemas de la posible pregunta, es decir, no mejora el número de coincidencias. Cuando llega a la cuarta pregunta, “¿Cómo se llaman tus padres?” va a encontrar una coincidencia de tal forma:

Lemas de la respuesta:	{hermano, familia, padre}
Lemas de la posible pregunta:	{llamar, padre}

Como podemos observar el lema “padre” coincide en ambas, es decir, en esta posible pregunta encontramos una coincidencia. Como el número de



coincidencias es mejor que 0, la pregunta candidata “¿Cuál ha sido el momento más feliz de tu vida?” se sustituye por la pregunta “¿Cómo se llaman tus padres?”. Es por esto que cuando analizamos el resto de posibles preguntas, no encontramos ninguna que sea mejor porque no superan el número de coincidencias en lemas. En conclusión, esta será la siguiente pregunta que se le plantee al usuario, la cual tiene relación con la que se había hecho anteriormente.

#### **4.2. Módulo primero de clasificación de respuestas en etapas de vida y en sentimientos**



# Capítulo 5

## Conclusiones y Trabajo Futuro

Conclusiones del trabajo y líneas de trabajo futuro.



# Chapter 5

## Conclusions and Future Work

Conclusions and future lines of work.



# Bibliografía

*Y así, del mucho leer y del poco dormir,  
se le secó el cerebro de manera que vino  
a perder el juicio.*

Miguel de Cervantes Saavedra

*Concesión del Proyecto CANTOR.* Versión electrónica, Disponible en <https://www.ucm.es/itc/noticias/40943>.

*Extracción de información personal a partir de redes sociales para la creación de un libro de vida.* Versión electrónica, Disponible en <https://eprints.ucm.es/id/eprint/68328/>.

*Extracción de preguntas a partir de imágenes para personas con problemas de memoria mediante técnicas de Deep Learning.* Versión electrónica, Disponible en <https://eprints.ucm.es/id/eprint/66857/>.

*Generación de resúmenes de video-entrevistas utilizando redes neuronales.* Versión electrónica, Disponible en <https://eprints.ucm.es/id/eprint/68333/>.

*Herramienta de ayuda guiada para la reminiscencia.* Versión electrónica, Disponible en <https://eprints.ucm.es/id/eprint/68332/>.

*How to Train Text Classification Model in spaCy?*. Versión electrónica, Disponible en <https://www.machinelearningplus.com/nlp/custom-text-classification-spacy/>.

*¿Qué es la demencia?*. Versión electrónica, Disponible en <https://www.alz.org/alzheimer-demencia/que-es-la-demencia?lang=es-MX>.

*¿Qué es la terapia ocupacional.* Versión electrónica, Disponible en <https://aptoca.org/terapia-ocupacional/que-es-la-terapia-ocupacional-2/>.

*Sistema de asistencia para cuidados de enfermos del Alzheimer.* Versión electrónica, Disponible en <https://eprints.ucm.es/id/eprint/67069/>.

*TextCategorizer.* Versión electrónica, Disponible en <https://spacy.io/api/textcategorizer>.