

---

# Título del Trabajo. Modificar la portada en Cascaras/cover.tex

---



Trabajo de Fin de Máster  
Curso 2020–2021

Autor  
Nombre Apellido1 Apellido2

Director  
Director 1  
Director 2

Máster en Ingeniería Informática  
Facultad de Informática  
Universidad Complutense de Madrid



# Título del Trabajo. Modificar la portada en Cascaras/cover.tex

**Trabajo de Fin de Máster en Ingeniería Informática**  
**Departamento de XXXXXXXXXXXXXXXX**

**Autor**  
**Nombre Apellido1 Apellido2**

**Director**  
**Director 1**  
**Director 2**

**Convocatoria:** *Febrero/Junio/Septiembre 2021*  
**Calificación:** *Nota*

**Máster en Ingeniería Informática**  
**Facultad de Informática**  
**Universidad Complutense de Madrid**

**28 de noviembre de 2021**



# Autorización de difusión

El abajo firmante, matriculado en el Máster en Ingeniería en Informática de la Facultad de Informática, autoriza a la Universidad Complutense de Madrid (UCM) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor el presente Trabajo Fin de Máster: “TITULO DEL TRABAJO”, realizado durante el curso académico CURSO bajo la dirección de DIRECTORES en el Departamento de XXXXXXXXXXXXXXXXXXXXXXXXXX, y a la Biblioteca de la UCM a depositarlo en el Archivo Institucional E-Prints Complutense con el objeto de incrementar la difusión, uso e impacto del trabajo en Internet y garantizar su preservación y acceso a largo plazo.

Nombre Del Alumno

28 de noviembre de 2021



# 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



# 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
1.1.1. Explicaciones adicionales . . . . .	3
<b>2. Estado de la Cuestión</b>	<b>9</b>
2.1. Alzheimer e historias de vida . . . . .	9
2.2. Generación de lenguaje natural . . . . .	11
2.2.1. Clasificación de sistemas NGL según su entrada . . . . .	12
2.3. Arquitectura tradicional de un sistema NLG . . . . .	14
2.3.1. Macroplanificación . . . . .	15
2.3.2. Microplanificación . . . . .	16
2.3.3. Realización . . . . .	17
2.4. Modelos y herramientas NLG . . . . .	18
2.4.1. Modelos de lenguaje estadísticos . . . . .	18
2.4.2. Seq2seq . . . . .	18
2.4.3. Transformers . . . . .	20
2.4.4. SimpleNLG . . . . .	22
2.5. Proyectos relacionados . . . . .	23
<b>3. Conclusiones y Trabajo Futuro</b>	<b>27</b>
<b>3. Conclusions and Future Work</b>	<b>29</b>
<b>A. Título</b>	<b>31</b>
<b>B. Título</b>	<b>33</b>
<b>Bibliografía</b>	<b>35</b>





# Índice de figuras

2.1. Reducción del cerebro asociada al Alzheimer, Mattson (2004)	10
2.2. Sistema data-to-text FoG . . . . .	13
2.3. Ejemplo de D2T utilizado por Sai et al. (2020) . . . . .	14
2.4. Arquitectura de referencia para sistema NLG, Vicente et al. (2015) . . . . .	15
2.5. Arquitectura de un sistema Seq2Seq . . . . .	19
2.6. Cálculo del estado oculto en el encoder . . . . .	19
2.7. Cálculo del estado oculto en el decoder . . . . .	20
2.8. Arquitectura del modelo Transformer . . . . .	21
2.9. Modelos surgidos a partir de Bert . . . . .	23
2.10. Arquitectura del sistema DICE . . . . .	24
2.11. Entrada y salida del sistema T5nlg . . . . .	24
2.12. Arquitectura modelo conversacional (de Jesús y García) . . .	25



# Índice de tablas



Chapter **1**

Introduction



# Capítulo 1

## Introducción

*“Frase célebre dicha por alguien inteligente”*

— Autor

### 1.1. Motivación

#### 1.1.1. Explicaciones adicionales

Si quieres cambiar el **estilo del título** de los capítulos, abre el fichero `TeXiS\TeXiS_pream.tex` y comenta la línea `\usepackage[Lenny]{fncychap}` para dejar el estilo básico de L<sup>A</sup>T<sub>E</sub>X.

Si no te gusta que no haya **espacios entre párrafos** y quieres dejar un pequeño espacio en blanco, no metas saltos de línea (`\\`) al final de los párrafos. En su lugar, busca el comando `\setlength{\parskip}{0.2ex}` en `TeXiS\TeXiS_pream.tex` y aumenta el valor de `0,2ex` a, por ejemplo, `1ex`.

El siguiente texto se genera con el comando `\lipsum[2-20]` que viene a continuación en el fichero `.tex`. El único propósito es mostrar el aspecto de las páginas usando esta plantilla. Quita este comando y, si quieres, comenta o elimina el paquete *lipsum* al final de `TeXiS\TeXiS_pream.tex`

##### 1.1.1.1. Texto de prueba

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend fau-



cibus, vehicula eu, lacus.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetur at, consectetur sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetur a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Pellentesque sit amet pede ac sem eleifend consectetur. Nullam elementum, urna vel imperdiet sodales, elit ipsum pharetra ligula, ac pretium ante justo a nulla. Curabitur tristique arcu eu metus. Vestibulum lectus. Proin mauris. Proin eu nunc eu urna hendrerit faucibus. Aliquam auctor, pede consequat laoreet varius, eros tellus scelerisque quam, pellentesque hendrerit ipsum dolor sed augue. Nulla nec lacus.

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetur odio sem sed wisi.

Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetur eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

Etiam euismod. Fusce facilisis lacinia dui. Suspendisse potenti. In mi erat, cursus id, nonummy sed, ullamcorper eget, sapien. Praesent pretium, magna in eleifend egestas, pede pede pretium lorem, quis consectetur tortor sapien facilisis magna. Mauris quis magna varius nulla scelerisque imperdiet. Aliquam non quam. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies

ut, mi. Duis nec dui quis leo sagittis commodo.

Aliquam lectus. Vivamus leo. Quisque ornare tellus ullamcorper nulla. Mauris porttitor pharetra tortor. Sed fringilla justo sed mauris. Mauris tellus. Sed non leo. Nullam elementum, magna in cursus sodales, augue est scelerisque sapien, venenatis congue nulla arcu et pede. Ut suscipit enim vel sapien. Donec congue. Maecenas urna mi, suscipit in, placerat ut, vestibulum ut, massa. Fusce ultrices nulla et nisl.

Etiam ac leo a risus tristique nonummy. Donec dignissim tincidunt nulla. Vestibulum rhoncus molestie odio. Sed lobortis, justo et pretium lobortis, mauris turpis condimentum augue, nec ultricies nibh arcu pretium enim. Nunc purus neque, placerat id, imperdiet sed, pellentesque nec, nisl. Vestibulum imperdiet neque non sem accumsan laoreet. In hac habitasse platea dictumst. Etiam condimentum facilisis libero. Suspendisse in elit quis nisl aliquam dapibus. Pellentesque auctor sapien. Sed egestas sapien nec lectus. Pellentesque vel dui vel neque bibendum viverra. Aliquam porttitor nisl nec pede. Proin mattis libero vel turpis. Donec rutrum mauris et libero. Proin euismod porta felis. Nam lobortis, metus quis elementum commodo, nunc lectus elementum mauris, eget vulputate ligula tellus eu neque. Vivamus eu dolor.

Nulla in ipsum. Praesent eros nulla, congue vitae, euismod ut, commodo a, wisi. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Aenean nonummy magna non leo. Sed felis erat, ullamcorper in, dictum non, ultricies ut, lectus. Proin vel arcu a odio lobortis euismod. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Proin ut est. Aliquam odio. Pellentesque massa turpis, cursus eu, euismod nec, tempor congue, nulla. Duis viverra gravida mauris. Cras tincidunt. Curabitur eros ligula, varius ut, pulvinar in, cursus faucibus, augue.

Nulla mattis luctus nulla. Duis commodo velit at leo. Aliquam vulputate magna et leo. Nam vestibulum ullamcorper leo. Vestibulum condimentum rutrum mauris. Donec id mauris. Morbi molestie justo et pede. Vivamus eget turpis sed nisl cursus tempor. Curabitur mollis sapien condimentum nunc. In wisi nisl, malesuada at, dignissim sit amet, lobortis in, odio. Aenean consequat arcu a ante. Pellentesque porta elit sit amet orci. Etiam at turpis nec elit ultricies imperdiet. Nulla facilisi. In hac habitasse platea dictumst. Suspendisse viverra aliquam risus. Nullam pede justo, molestie nonummy, scelerisque eu, facilisis vel, arcu.

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit

amet, consectetur adipiscing elit. Etiam congue neque id dolor.

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.



## Capítulo 2

# Estado de la Cuestión

### 2.1. Alzheimer e historias de vida

La pirámide de población modifica su estructura continuamente debido al progresivo envejecimiento de población generalizado. Según Alzheimer's Association International Conference (27 de Julio de 2021), en el año 2050 las personas mayores de 65 años constituirán el 16 % de la población mundial frente al 8 % del año 2010. Una de las enfermedades más comunes dentro de este rango de población es la enfermedad del Alzheimer, cuya prevalencia a nivel global se espera que supere todo dato conocido hasta ahora, ya que se estima que en 2050 se incrementa el número de casos a 152.8 millones frente a los 57.4 millones de 2019.

La enfermedad de Alzheimer es un trastorno neurológico caracterizado por cambios degenerativos en diferentes sistemas neurotransmisores que abocan finalmente a la muerte de las células nerviosas del cerebro encargadas del almacenamiento y procesamiento de la información. Las regiones del cerebro involucradas en la memoria y procesos de aprendizaje, asociadas a los lóbulos temporal y frontal, reducen su tamaño como resultado de la degeneración de las sinapsis y la muerte de las neuronas (Romano et al., 2007; Mattson, 2004). En la figura 2.1 se puede comprobar de manera visual esta reducción del cerebro.

Es una enfermedad difícil de detectar ya que, por lo general, los síntomas iniciales de la enfermedad suelen atribuirse a un olvido puntual o la vejez. Nada más lejos de la realidad. Según avanza la enfermedad, sus síntomas lo hacen con ella, agravándose y aumentando cada vez más hasta que el deterioro cognitivo ocasionado llega a afectar significativamente a las actividades de la vida diaria y finalmente a las necesidades fisiológicas básicas.

La evolución del Alzheimer se puede dividir en tres fases o etapas: Inicial, en la que se observa un deterioro cognitivo leve como puede ser la pérdida paulatina de la memoria episódica. Intermedia, con pérdidas de la memoria

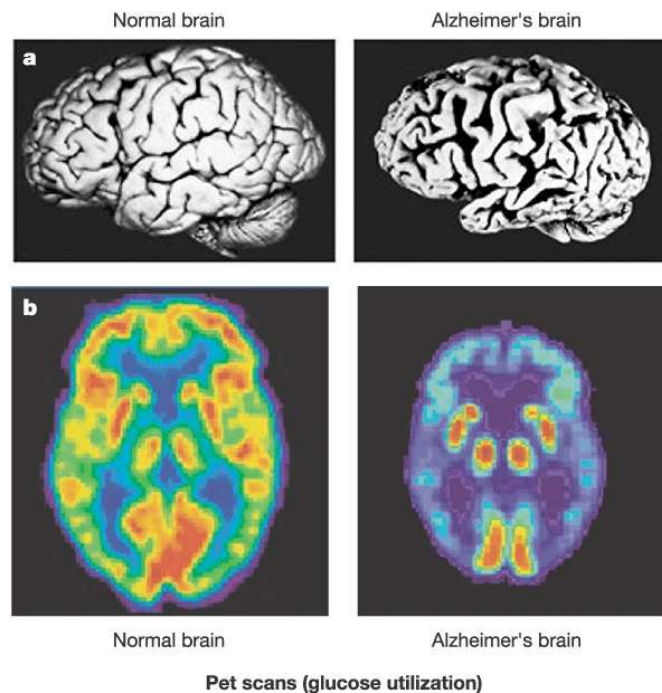


Figura 2.1: Reducción del cerebro asociada al Alzheimer, Mattson (2004)

reciente asociada a un deterioro mayor. Avanzada, en la que se pierde progresivamente la memoria de los acontecimientos más antiguos, acompañando también un gran deterioro físico.

Los síntomas del Alzheimer son muy diversos, ya que no solo provoca problemas de memoria, sino también alteraciones en el estado de ánimo y la conducta, dificultad de toma de decisiones, desorientación, problemas del lenguaje, dificultad para comer y movilidad reducida, entre otros. Estos síntomas dependen de la fase evolutiva de la enfermedad.

En la actualidad, el Alzheimer es una enfermedad irreversible. Sin embargo, existen diversos tratamientos que pretenden ralentizar el avance de la enfermedad y/o mejorar la calidad de vida de los pacientes. Podemos dividir estos tratamientos en dos ramas: tratamientos farmacológicos, que hacen uso de medicamentos, y tratamientos no farmacológicos o psicosociales, que no utilizan sustancias químicas. Ambos tratamientos son eficaces y de la combinación de ambos resulta el procedimiento más recomendado para tratar la enfermedad. (Romano et al., 2007)

Existen una gran variedad de terapias no farmacológicas. Algunas de las más utilizadas son el entrenamiento y estimulación cognitiva, ejercicio físico, musicoterapia, etc. Además, en cada una de estas terapias podemos encontrar una enorme cantidad de técnicas, siendo la reminiscencia la más

utilizada como terapia de estimulación cognitiva.

Según O'Rourke et al. (2013), la reminiscencia es el acto o proceso de recordar sucesos, eventos o información del pasado. Esto puede implicar el recuerdo de episodios particulares o genéricos que pueden o no haber sido olvidados previamente, y que son acompañados por la sensación de que estos episodios son relatos verídicos de las experiencias originales. Esta técnica es utilizada para estimular la memoria episódica autobiográfica mediante el encadenamiento de recuerdos, que se agrupan en categorías y se archivan en el tiempo mediante la elaboración de *la historia de vida*.

La historia de vida es una técnica narrativa que se basa en organizar y estructurar recuerdos de una persona para componer una autobiografía. Según Linde et al. (1993), una historia de vida debe cumplir dos criterios: primero, debe incluir algunos puntos de evaluación que comuniquen los valores morales de la persona; y segundo, los eventos incluidos en la historia de vida deben tener un significado especial y ser de importancia para ella. Estos eventos deben ser aspectos significativos de la vida pasada de la persona, su presente y su futuro.

Para componer la historia de vida de una persona con Alzheimer se recopilan historias a través de familiares u otras personas cercanas. Posteriormente, se documentan en forma de un libro o cuaderno, incluyendo experiencias y logros junto con fotografías y escritos sobre hechos importantes para la vida de la persona, a través de los cuales se muestra quién es esa persona.

Las historias de vida ayudan a las personas con Alzheimer a conectar con su identidad recordando épocas felices. El miedo y la frustración provocados por el olvido de las tareas de la vida cotidiana, nombres y rostros, se mitigan recordando quiénes eran a través de estas historias. Les ayuda a ser conscientes de los momentos especiales que han marcado su vida, las personas que han conocido en su infancia o trabajo. También pueden ser utilizados por los cuidadores para saber más de ellos y ayudarles en la reminiscencia de recuerdos. (Karlsson et al., 2014)

## 2.2. Generación de lenguaje natural

La Generación de Lenguaje Natural (GLN) se define como el “subcampo de la inteligencia artificial y lingüística computacional que se ocupa de la construcción de sistemas informáticos que pueden producir textos comprensibles en inglés u otros lenguajes humanos a partir de alguna representación no lingüística subyacente de la información” (Reiter y Dale, 1997). Si bien esta definición estuvo generalmente aceptada como la más conveniente al hablar de generación de lenguaje natural durante muchos años, Gatt y Krahmer (2018) puntualizan que es una afirmación que solo engloba una parte de la generación de textos, ya que se refiere únicamente a aquellos sistemas cuya entrada es una ‘representación no lingüística [...] de la información’ o datos,

como veremos más adelante en el apartado 2.2.1.

Desde hace muchos años, el GNL es empleado en numerosos proyectos de distinta naturaleza como la traducción de textos (Cho et al., 2014), realización de resúmenes y fusión de documentos (Clarke y Lapata, 2010), corrección automática de ortografía y gramática (Kukich, 1992), redacción de noticias (Molina et al., 2011), informes meteorológicos (Goldberg et al., 1994), informes financieros (Kukich, 1983), generación de resúmenes sobre la información de recién nacidos en un contexto clínico (Gatt et al., 2009)... Todos estos sistemas construidos tienen en común la generación de un texto (normalmente de una alta calidad) a partir de muy diferentes fuentes de información.

### **2.2.1. Clasificación de sistemas NGL según su entrada**

En los ejemplos de proyectos listados con anterioridad en los que se empleó generación de lenguaje natural para redactar distintos textos, podemos percatarnos de que los datos utilizados como fuente de información son muy dispares, no solo en su contenido sino también en el tipo de dato. Así, si para la traducción de textos se utiliza texto ya existente como entrada, en otros sistemas como en la generación de informes meteorológicos se emplean datos no lingüísticos. De esta manera, se consideran dos posibles enfoques en los sistemas NGL dependiendo del tipo de entrada: text-to-text y data-to-text.

#### **2.2.1.1. Generación text-to-text (T2T)**

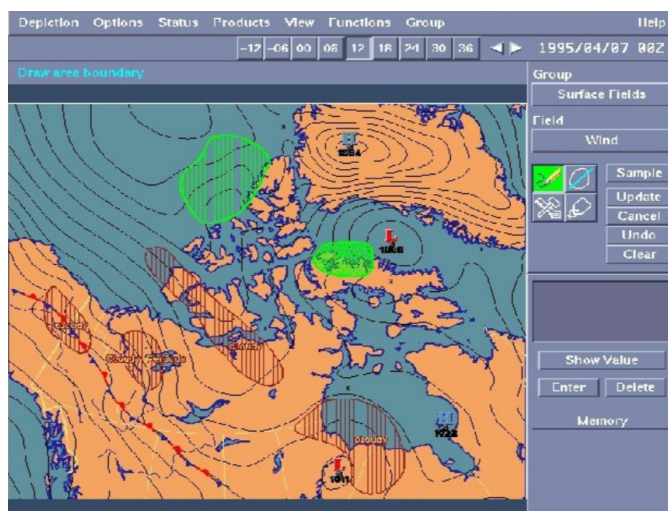
Estos tipos de sistemas de generación toman textos existentes como entrada y produce un texto nuevo y coherente como salida. La entrada de estos sistemas puede abarcar desde pequeñas oraciones a extensos textos. Existen muchas aplicaciones en los sistemas NLG que utilizan T2T como pueden ser generación de resúmenes, fusión, simplificación de textos, corrección gramatical, entre otros.

Si consideramos un ejemplo concreto, cualquier traductor automático es de tipo text-to-text ya que utilizan una entrada textual correspondiente a un escrito en un idioma y genera un texto en otro. La traducción automática es un proceso muy complejo puesto que no solamente hay que tener en cuenta el significado del escrito, sino que también hace falta interpretar y analizar de manera correcta todos los elementos del texto y saber cómo influyen unas palabras en otras para generar un texto fluido y coherente.

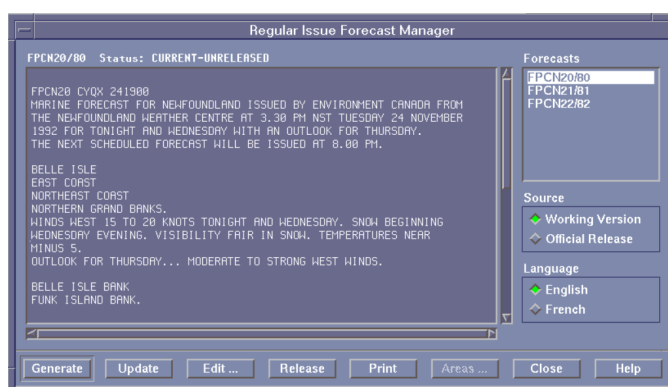
#### **2.2.1.2. Generación data-to-text (D2T)**

Estos tipos de sistemas permiten la generación de texto como salida a partir de entradas no textuales. Además, el formato de los datos que puede tomar como entrada son muy diversos. Aunque es muy común encontrar





(a) Entrada del sistema FoG



(b) Salida del sistema FoG

Figura 2.2: Sistema data-to-text FoG

sistemas que parten de datos numéricos como hojas de cálculo, hay que considerar otros orígenes de datos de tipo estructurado tales como bases de datos, simulaciones de sistemas físicos, hojas de cálculo o grafos de conocimientos.

Algunos autores prefieren utilizar el término *concepto* en lugar de *data*, motivo por el que algunos se refieren a este enfoque como generación concepto-to-text (c2t), (Vicente et al., 2015).

Uno de los ejemplos más visuales que nos permite comprender este tipo de sistema sería el Forecast Generator, sistema que forma parte del Forecaster's Production Assistan, entorno desarrollado por CoGenTex en 1992 para Environment Canada con el fin ayudar a los meteorólogos a aumentar su productividad al redactar por ellos un informe meteorológico textual en inglés y en francés (Goldberg et al., 1994). En la figura 2.2a se muestra el entorno sobre el que los meteorólogos modifican valores como la presión at-

<b>Data:</b> (John E Blaha <i>birthdate</i> 1942 08 26) (John E Blaha <i>birthplace</i> San Antonio) (John E Blaha <i>occupation</i> Fighter Pilot) <b>Reference Text:</b> John E Blaha, born in San Antonio on 1942-08-26, worked as a fighter pilot
1. John E Blaha who worked as a fighter pilot was born on 26.08.1942. 2. Fighter pilot John E Blaha was born in San Antonio on the 26th July 1942. 3. John E Blaha, born on the 26th of August 1942 in San Antonio, served as a fighter pilot.

Figura 2.3: Ejemplo de D2T utilizado por Sai et al. (2020)

mosférica, situación de frentes y otros datos (datos no textuales). Una vez se pulsa sobre 'Generar', el sistema muestra el texto correspondiente al informe (figura 2.2b).

En la figura 2.3, explicada con más detalle en Sai et al. (2020), se muestran los datos de entrada y de salida de un sistema NLG D2T acercándonos a la generación de lenguaje desde una perspectiva distinta al ejemplo explicado anteriormente. La fila superior corresponde a la entrada. En este caso a partir de un texto de referencia se obtienen unos datos que se introducen en el sistema, ya sea en forma de grafo u otro tipo de datos semiestructurados como tablas (conjunto de tuplas del tipo [entidad, atributo, valor]). En la fila inferior, se muestran diferentes posibles soluciones como salida del sistema. Además, el autor introduce la necesidad de métodos de evaluación de la calidad del texto redactado ya que de las diferentes salidas, solo la tercera opción cubre toda la información de entrada y resulta ser fluida.

### 2.3. Arquitectura tradicional de un sistema NLG

El objetivo final de un sistema de generación de lenguaje natural es mapear unos datos de entrada a un texto de salida. (Reiter y Dale, 1997). Sin embargo, este proceso, aunque pueda parecer sencillo de entender, resulta complicado de llevar a cabo. Al principio del desarrollo de sistemas NLG, no había un consenso entre autores a la hora de establecer un proceso para construir este sistema. Finalmente, Reiter y Dale (1997) propusieron una arquitectura asociada a una lista de tareas que se deben recomendablemente realizar a la hora de llevar a cabo dicha construcción. Esta arquitectura surgió de la observación de los diferentes sistemas que se habían llevado a cabo hasta la fecha. Actualmente, es la solución más extendida y reconocida.

La arquitectura presentada por Reiter y Dale (1997), como se puede observar en la figura 2.4, se divide en 3 módulos: macroplanificación, mi-

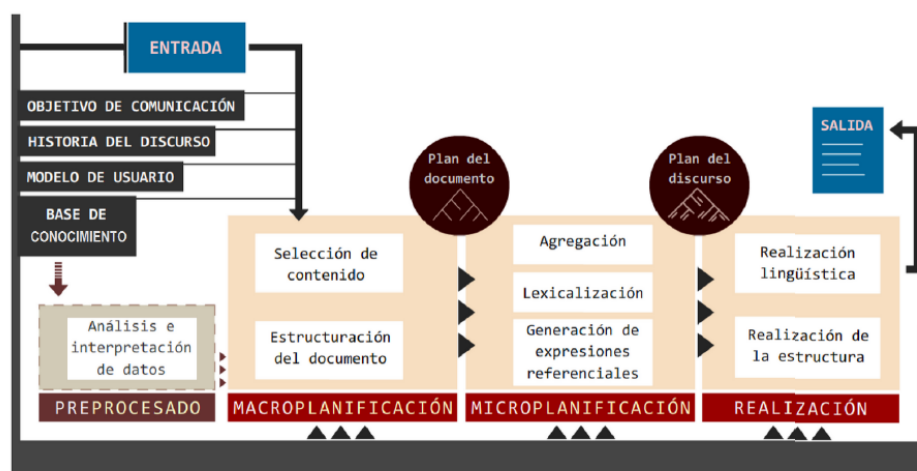


Figura 2.4: Arquitectura de referencia para sistema NLG, Vicente et al. (2015)

croplanificación y realización. Además, cada módulo contiene una lista de tareas. Esta asignación tareas-módulo no es inamovible, una tarea asociada a un módulo se puede realizar en otro si así se considera, incluso implementar su desarrollo a lo largo de varios módulos. Los módulos que se corresponden con las tareas iniciales suelen estar relacionados con adaptar datos o estructura a nuestro sistema, mientras que los módulos finales corresponden a la transformación de los resultados intermedios en en texto final.

### 2.3.1. Macroplanificación

Este es el primer módulo del sistema. Debe determinar qué decir, seleccionando para ello la información de entrada necesaria y organizarlo en una estructura coherente, resultando de este proceso el plan del documento. Las tareas que intervienen se describen en los apartados siguientes:

#### 2.3.1.1. Selección del contenido

La determinación del contenido puede definirse como el proceso de decidir que información debe ser incluida en el texto generado y cual no. Por lo general, la información de la que partimos contendrá más información de la que nos interesa, así debemos decidir que información resulta innecesaria y por tanto tenemos que eliminar para la generación del texto final. También hay que tener en cuenta el público al que está dirigido el texto generado, ya que dependiendo de este podremos incluir cierta información de los datos entrantes o no.

Este proceso de selección de la información correspondiente lleva a cabo la filtración y resumen de esta en un conjunto de *mensajes*. Cada uno de estos mensajes corresponde al significado de una palabra u oración y se le asigna una entidad, concepto o relación dominante.

### 2.3.1.2. Estructuración del documento

Definiendo el concepto *texto* como 'unidad de comunicación completa, formada habitualmente por una sucesión ordenada de enunciados que transmiten un mensaje con las siguientes propiedades: adecuación, coherencia y cohesión', podemos advertir que un texto no es un conjunto aleatorio de oraciones, sino que es necesario la existencia de un orden en la presentación del texto final.

Dependiendo de la información que se comunique, este orden puede verse modificado u alterado, es por ello que no hay una estructura fija, sino que hay que adecuarla al tipo de documento.

Una vez realizada la estructuración del texto, se obtiene un plan de discurso que corresponde a una representación estructurada y ordenada de los mensajes obtenidos en la tarea anterior.

### 2.3.2. Microplanificación

Segundo módulo de la arquitectura que parte del plan del documento. Se generan las oraciones evitando información redundante e innecesaria en un discurso. El resultado de este módulo es el plan de discurso. Lo hace mediante tres tareas:

#### 2.3.2.1. Agregación de oraciones

La generación de una oración por cada uno de los mensajes puede resultar en la generación de un texto redundante y excesivamente estructurado. Una tarea en el proceso de construcción de un sistema NLG es la agregación de oraciones que pretenden paliar este problema mediante la unión o agregación de contenidos de distintos mensajes en una sola oración. De esta manera los mensajes se combinan para obtener oraciones más largas y complejas, resultando en conjunto un texto con una mayor simplicidad.

#### 2.3.2.2. Lexicalización

En esta fase del proceso se empieza a generar el texto en lenguaje natural como tal, a partir de las estructuras sintácticas y palabras específicas de las etapas anteriores. La dificultad de la generación en esta etapa reside en la gran cantidad de alternativas que encontramos a una sola palabra u oración para expresar los bloques de mensajes, además debemos tener en cuenta

un número mayor de posibilidades ya que debemos además considerar las necesidades o conocimiento de los usuarios, si el objetivo es generar textos con variaciones sintácticas o semánticas a lo largo del mismo, selección de palabras sinónimas, selección de adjetivos relacionados estrechamente con sustantivos, consideración de magnitudes...

### **2.3.2.3. Generación de expresiones de referencia**

La diferenciación de unas entidades de otras para poder generar expresiones que se refieran a ellas es tratada en esta tarea con el objetivo de evitar la ambigüedad. Para realizar esta tarea se debe conseguir encontrar características particulares que contribuyan a diferenciar a una entidad del resto de entidades. Esta etapa está bastante consensuada en el campo NLG.

La generación de expresiones de referencia (REG, por sus siglas en inglés) debe llevarse a cabo una vez que el plan del documento se haya generado y depende de este, esto implica que esta fase debe llevarse a cabo desde el primer momento después de que se hayan analizado los datos. Debemos adaptar el plan de documento del primer módulo a lo que necesita REG, es por ello que debemos tener conocimiento de ello desde el comienzo.

Un caso especialmente estudiado que aplica esta técnica es la descripción de imágenes, ya que debe tener en cuenta si un elemento se encuentra a la derecha de otro, detrás de otro, etc, para poder enriquecer el texto. Para ello es necesario reconocer y distinguir los elementos en escena unos de otros y así, obtener una descripción lo más fidedigna posible a la imagen real.

### **2.3.3. Realización**

La realización constituye el último módulo de la arquitectura de un sistema NLG. El objetivo final corresponde en generar oraciones gramaticalmente correctas para comunicar mensajes. En este módulo deberán tenerse en cuenta reglas a cerca de la formación de verbos (elección del tiempo verbal adecuado y por tanto generación de las palabras correspondientes), reglas sobre concordancia de género y número entre palabras (Reiter y Dale (1997) no tiene en cuenta el género de las palabras ya que focaliza la generación del lenguaje al inglés), generación de pronombres...

La entrada sobre la que se trabaja es el plan de discurso que contiene información sobre las oraciones generadas y la estructura utilizada en el texto final. En esta fase se traduce esta entrada en la salida que el usuario final recibirá.

Algunos autores consideran una única tarea de realización que engloba el convertir las especificaciones en oraciones y el dar un formato final al texto. Otros prefieren separar estas etapas para diferenciarlas y que sea más sencillo su estudio.

### 2.3.3.1. Realización lingüística

Con el objetivo de transformar las especificaciones de oraciones en las oraciones finales, en esta fase se ordenan los diferentes elementos constitutivos de una oración y se les asigna un formato correcto. Para elegir la forma morfológica correcta de una palabra se debe conjugar verbos, establecer concordancias de palabras, añadir formas pronominales en los lugares adecuados de las oraciones y establecer los signos de puntuación adecuados.

### 2.3.3.2. Realización de la estructura

Esta etapa no está considerada por algunos autores como tal aunque aquí se muestra ya que puede ser relevante en ciertos contextos. En algunos documentos, es necesario añadir o modificar algunas líneas del texto para darle una estructura al documento. Es por eso, que si queremos generar un texto en formato html se debe añadir las etiquetas a lo largo del texto generado. Igualmente si el documento final utiliza Latex.

## 2.4. Modelos y herramientas NLG

En esta sección se identifican los distintos tipos de modelos que han surgido durante los últimos años. Por otra parte, se describen los Modelos de Lenguaje (LM) actualmente más relevantes para la generación de lenguaje natural junto con las herramientas que nos permitan utilizarlos.

### 2.4.1. Modelos de lenguaje estadísticos

Estos tipos de modelos utilizan técnicas estadísticas y reglas lingüísticas para aprender la distribución de probabilidad de las palabras y, de esta manera generar lenguaje. Entre las técnicas más utilizadas en este ámbito encontramos el modelo N-Gram y Markov.

Un modelo N-Gram nos permite realizar una predicción estadística de un n-gram dentro de cualquier secuencia de palabras dado el historial de las N-1 palabras anteriores. Sin embargo, esta dependencia de generación de la palabra siguiente a una secuencia dada con respecto al conjunto de palabras generadas inmediatamente antes puede llevar a generaciones erróneas que no tengan en cuenta otros contextos anteriores.

### 2.4.2. Seq2seq

El modelo Sequence-to-Sequence (Seq2Seq), que sigue una clase especial de arquitectura de Red Neuronal Recurrente (RNN), ha alcanzado mucho éxito a la hora de resolver problemas complejos de Procesamiento de Lenguaje Natural, incluso llegando a superar a los modelos estadísticos de lenguaje

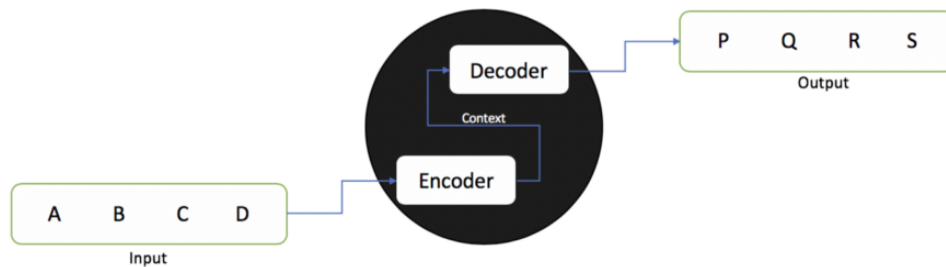


Figura 2.5: Arquitectura de un sistema Seq2Seq

$$h_t = f(W^{(hh)}h_{t-1} + W^{(hx)}x_t)$$

Figura 2.6: Cálculo del estado oculto en el encoder

en su efectividad (Joshi, 2020).

Considerando una aproximación lo más sencilla posible, podríamos representarlo como un sistema que toma una secuencia de elementos como entrada (input) y genera otra secuencia de elementos de salida (output). Como se muestra en la figura 2.5, este sistema está compuesto internamente por un *encoder* y un *decoder* que tradicionalmente implementan modelos basados en redes neuronales como LSTM (o GRU en menor número de casos).

La tarea del *encoder* consiste en resumir la información de la secuencia que se introdujo como entrada en forma de un vector de estado oculto o *context* y enviar los datos resultantes al *decoder*. El objetivo principal de este vector es encapsular la información de todos los elementos de entrada para ayudar al *decoder* a realizar predicciones precisas. Para calcular el estado oculto  $t$ -ésimo de la secuencia se utiliza la fórmula representada en la figura 2.6, donde ' $t$ ' corresponde al tamaño de la secuencia de entrada ' $x$ ' y ' $W$ ' representa a todas y cada una de las palabras del dataset. Para cada una de las celdas del *encoder* se calcula su vector de estado oculto, generando la última celda el vector de estados finales.

Por su parte, el *decoder* utiliza como estado inicial la salida del *encoder* correspondiente al vector de estados finales, calculando cada celda su estado oculto con la fórmula de la figura 2.7. Una vez que se obtiene el estado oculto  $h_t$ , puede generarse la secuencia de palabras final aplicando al dataset de palabras junto con  $h_t$  la función *softmax*.

$$h_t = f(W^{(hh)}h_{t-1})$$

Figura 2.7: Cálculo del estado oculto en el decoder

### 2.4.3. Transformers

Los Transformers surgen como una evolución de los modelos Seq2Seq. Es por esto que aún podemos encontrar características de estos últimos sistemas en la arquitectura externa de los Transformes como puede ser la existencia de dos componentes clave: el *encoder* y el *decoder*, cuyos objetivos coinciden en ambos.

A diferencia de los sistemas Seq2Seq, ya no se van a utilizar RNN para implementar el *encoder* y el *decoder*, que de hecho van a verse divididos en dos subcapas (*Feed Forward* y *Multi-Head Attention*) como se muestra en la figura 2.8 en la que se representa la arquitectura final de este tipo de sistemas.

Además, la diferencia de potencia de procesamiento entre ambos modelos es enorme. Si en Seq2Seq el *decoder* procesaba palabra a palabra la secuencia que le llegaba desde el *encoder*, los Transformers permiten alimentar al *decoder* con toda la secuencia de manera simultanea resultando en un mayor poder por parte de este tipo de modelos.

Hay que destacar la existencia de una herramienta en Python denominada 'Transformers' que proporciona una serie de sistemas de propósito general para Natural Language Understanding (NLU) y Natural Language Generation (NLG). Ofrece más de 32 modelos preentrenados en más de 100 idiomas, entre los que se encuentra el español. Entre los modelos más utilizados encontramos los famosos GPT-2 y BERT junto con un gran número de variaciones de ellos dependiendo de los datos utilizados para su entrenamiento.

#### 2.4.3.1. GPT-2

GPT-2 (Generative Pretrained Transformer) es un modelo NLG presentado por OpenAI en el año 2019 basado en redes neuronales para secuencias, basadas en la autoatención, y que ha sido construido sobre una arquitectura Transformer. El objetivo de este sistema es construir una distribución de probabilidad en la que para cada palabra posible a generar se le asigna una probabilidad en función del contexto anterior. Se trata de un modelo que ha sido preentrenado con un conjunto de datos correspondiente a las 8 millones de páginas web mejor valoradas en Reddit, lo que resulta en una gran base de conocimiento para generar textos automáticamente de manera muy correcta.



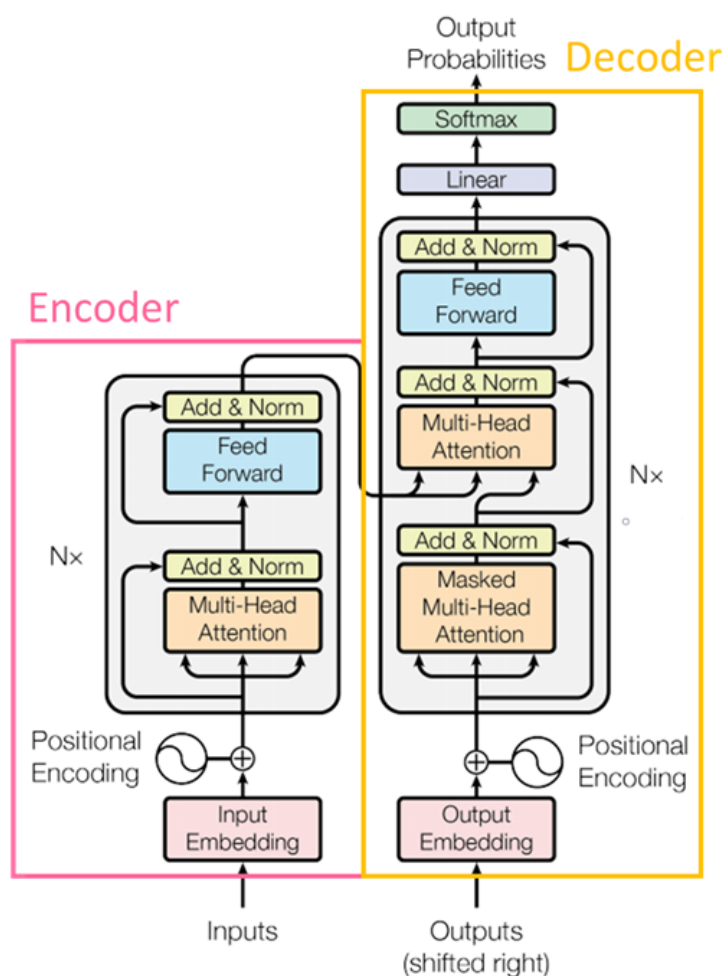


Figura 2.8: Arquitectura del modelo Transformer

La potencia de este modelo es tal que sus creadores no quisieron en un primer momento publicar la versión completa por miedo de que se pudiera utilizar de manera ilícita. Según fueron pasando los años, se fueron liberando progresivamente diferentes versiones del modelo original ya que comenzaban a surgir otros proyectos con potencias igualmente competitivas. Estas diferentes versiones se diferenciaban en el número de parámetros que admitía la arquitectura, de esta manera se conseguía limitar su funcionamiento. La primera versión contaba con 117 miles de millones de parámetros mientras que la última versión, publicada en 2020, posee 1,5 billones.

GPT-2 únicamente está disponible en inglés aunque puede hacer uso de GoogleTranslate API para generar textos en otros idiomas. Es importante resaltar que al depender del traductor se puede ver disminuida la calidad de

generación de lenguaje.

#### 2.4.3.2. BERT

BERT (Bidirectional Encoder Representations from Transformers) es un modelo NLP desarrollado por Google y publicado a finales de 2018. Está basado redes neuronales bidireccionales que tratan de predecir las palabras perdidas (enmascaradas) en una oración y determinar si dos oraciones consecutivas son continuación lógica entre sí para determinar si están conectadas por su significado. Aunque originalmente no estaba destinado a la generación de textos, Wang y Cho (2019) publicaron un método de utilización de este sistema para conseguir la generación de lenguaje que parece dar muy buenos resultados. De hecho, consiguió mejorar los resultados de la versión publicada en aquellos tiempos por GPT-2.

De este modelo han surgido numerosas variaciones que se han publicado a lo largo de estos años. Como se puede apreciar en la figura 2.9, encontramos distintas ramificaciones que podemos dividir en dos grupos: modelos preentrenados con un corpus específico perteneciente a un dominio y modelos Fine Tuned que se ajustan a una tarea específica utilizando un modelo previamente entrenado (Rajasekharan, 2019). Otras variaciones de BERT corresponden a los modelos contruidos a partir de él pero entrenados en otros lenguajes para generar textos en otra lengua distinta al inglés, que es la original. Beto es la versión en Español de BERT (Cañete et al., 2020) y ha sido entrenado con una gran corpus en dicho idioma.

#### 2.4.4. SimpleNLG

SimpleNLG es una API de Java que proporciona interfaces que ofrecen un control directo sobre la tarea de realización. Define un conjunto de tipos léxicos, correspondientes a las principales categorías gramaticales, así como formas de combinarlos y establecer valores de características.

Esta orientado a la generación de oraciones gramaticalmente correctas en sistemas data-to-text. Aunque originalmente solo estaba disponible para textos de lengua inglesa, actualmente se encuentra en muchos idiomas, entre ellos el español. La versión española de esta herramienta se llama SimpleNLG-ES y realmente se trata de una adaptación bilingüe de la versión original en inglés.

Esta herramienta se basa en la flexibilidad a la hora de generar textos mediante la utilización de manera combinada de sistemas basados en esquemas y otros sistemas más avanzados; robustez generando salidas (aunque en ocasiones incorrectas) cuando las entradas estén erróneas o incompletas; e independencia entre las operaciones de decisión de morfología y sintácticas.

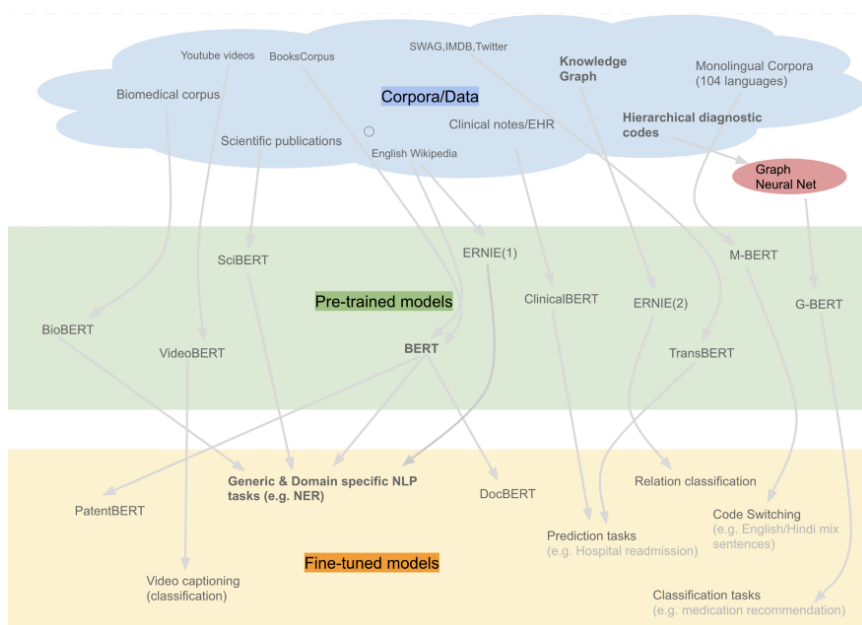


Figura 2.9: Modelos surgidos a partir de Bert

## 2.5. Proyectos relacionados

Muchos son los enfoques que se han estudiado para tratar de perfeccionar la generación de lenguaje natural. Los más tradicionales seguían la metodología presentada en la sección 2.3 de este documento. En ella dividían el problema principal en varios subproblemas o tareas, estos incluyen la selección de contenido, estructuración del texto, agregación, lexicalización, generación de expresiones de referencia y finalmente, la realización (Reiter y Dale, 1997). Sin embargo, en los últimos años ha crecido el interés por mirar más allá de aquella arquitectura. Los sistemas presentados en la sección 2.4 surgieron para romper con ella.

Todos estos sistemas presentados anteriormente, tienen muy poca capacidad o poca calidad a la hora de generar textos de manera controlada. Algunos de ellos como GPT-2, generan textos a partir de una oración inicial, resultando su salida un texto de tamaño variable que da continuidad a la oración de entrada. Otros como BERT, son capaces de generar palabras perdidas dentro de una oración. Sin embargo, pocos de ellos son capaces por sí mismos de generar contenido de manera controlada a partir de una información dada en todos los puntos de su generación.

Por todo esto, han surgido varios sistemas como DICE (?) que utilizarán estos modelos ajustándolos con el objetivo de controlar la generación del texto resultante. Como se muestra en la figura 2.10, este sistema está compuesto

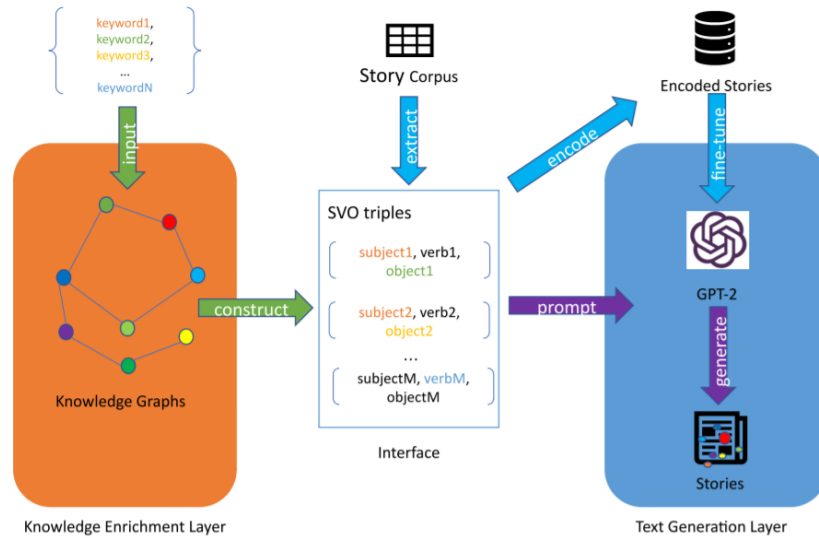


Figura 2.10: Arquitectura del sistema DICE

Sidhath   profession   Doctor && Sidhart  home_town   Bombay
Sisdath is a Doctor andis located in Bombay
Nie_Haisheng   birthDate   1994-10-13 && Nie_Haisheng   occupation   Fighter_pilot
Born on the 13th of October 1994, Nie Haisheng, was a fighter pilot

Figura 2.11: Entrada y salida del sistema T5nlg

por dos capas. La primera capa toma como entrada unas palabras clave introducidas por el usuario y forma un Grafo de Conocimiento. A continuación y para conectar ambas capas, utiliza tripletas como interfaz. Estas tripletas se pueden construir a partir del grafo o extraerse de un corpus de historias denominado ROCStory. Estas tripletas sirven como entrada a un sistema de generación de texto que utiliza GPT-2 para generar pequeñas historias.

Otro enfoque parecido utiliza una conocida dataset llamada WebNLG para entrenar un Modelo de Lenguaje. Esta base de datos contiene correspondencias entre textos y tripletas y es utilizada para ajustar el Modelo de Lenguaje T5. De esta manera, es capaz de generar textos a partir de tripletas introducidas como entrada al sistema. En la figura 2.11, podemos ver el formato de entrada.

Con respecto a propuestas dentro del ámbito de generación de lenguaje en terapias de reminiscencia, no son muchos los sistemas que encontramos. Por una parte, de Jesús y García propone desarrollar e implementar un modelo

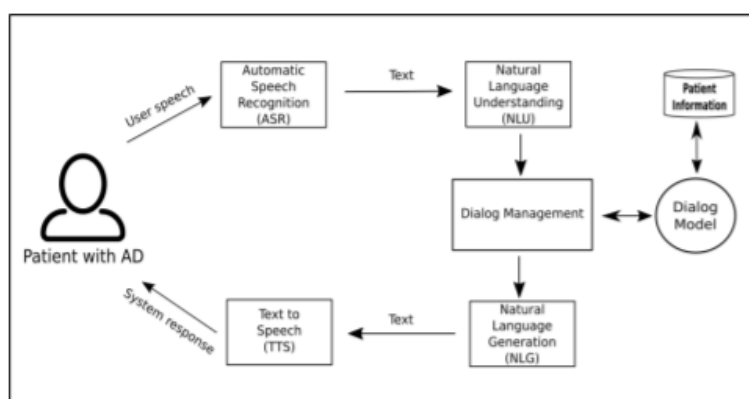


Figura 2.12: Arquitectura modelo conversacional (de Jesús y García)

conversacional que pueda ayudar a los cuidadores y a sus propios pacientes con Alzheimer a realizar un mayor número de terapias de reminiscencia periódicas para así, potenciar los beneficios de estas. Se centra en generar conversaciones personalizadas entre el prototipo del sistema conversacional y el paciente con el fin de recoger información relacionada con sus gustos, historial y estilo de vida. Su arquitectura, como se muestra en la figura 2.12, esta integrada por varios módulos: módulo de Reconocimiento Automático de Voz, Comprensión del Lenguaje Natural, Gestión de Diálogos, Modelo de Diálogos, Generación de Lenguaje Natural y Text-to-Speech.

En otra investigación (realizada por Shi y Setchi (2012)), se propone el desarrollo de un sistema computarizado llamado Life Story Book (LSB), que facilita el acceso y la recuperación de recuerdos almacenados que se utilizan como base para interacciones positivas entre ancianos y jóvenes, y especialmente entre personas con deterioro cognitivo y miembros de su familia o cuidadores. Para facilitar la gestión de la información y la generación dinámica de contenido, este artículo presenta un modelo semántico de LSB que se basa en el uso de ontologías y algoritmos avanzados para la selección de características y la reducción de dimensiones. Para terminar, propone un algoritmo llamado Onto-SVD que combina la selección de características semánticas y la ontología orientada al usuario con la utilización de SVD como método de reducción de dimensiones para lograr la identificación de temas basada en la similitud semántica.



# Capítulo 3

## Conclusiones y Trabajo Futuro

Conclusiones del trabajo y líneas de trabajo futuro.





# Chapter 3

## Conclusions and Future Work

Conclusions and future lines of work.



Apéndice **A**

Título

Contenido del apéndice



Apéndice	<b>B</b>
----------	----------

Título



# 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

ALZHEIMER'S ASSOCIATION INTERNATIONAL CONFERENCE. [www.alz.org](http://www.alz.org).  
27 de Julio de 2021.

CAÑETE, J., CHAPERON, G., FUENTES, R., HO, J.-H., KANG, H. y PÉREZ, J. Spanish pre-trained bert model and evaluation data. En *PML4DC at ICLR 2020*. 2020.

CHO, K., VAN MERRIENBOER, B., ÇAGLAR GÜLÇEHRE, BAHDANAU, D., BOUGARES, F., SCHWENK, H. y BENGIO, Y. Learning phrase representations using rnn encoder–decoder for statistical machine translation. En *EMNLP*. 2014.

CLARKE, J. y LAPATA, M. Discourse constraints for document compression. *Computational Linguistics*, vol. 36(3), páginas 411–441, 2010.

GATT, A. y KRAHMER, E. Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, vol. 61, 2018.

GATT, A., PORTET, F., REITER, E., HUNTER, J., MAHAMOOD, S., WENDY, M. y SRIPADA, S. From data to text in the neonatal intensive care unit: Using nlg technology for decision support and information management. *AI Commun.*, vol. 22, páginas 153–186, 2009.

GOLDBERG, E., DRIEDGER, N. y KITTREDGE, R. I. Using natural-language processing to produce weather forecasts. *IEEE Expert*, vol. 9(2), páginas 45–53, 1994.

- DE JESÚS, V. M. M. y GARCÍA, M. J. S. A conversational model for the reminiscence therapy of patients with early stage of alzheimer. ????
- JOSHI, P. Natural language generation using pytorch: Model and generate text data. 2020.
- KARLSSON, E., SÄVENSTEDT, S., AXELSSON, K. y ZINGMARK, K. Stories about life narrated by people with alzheimer's disease. *Journal of Advanced Nursing*, vol. 70(12), 2014.
- KUKICH, K. Design of a knowledge-based report generator. En *ACL*. 1983.
- KUKICH, K. Techniques for automatically correcting words in text. *ACM Comput. Surv.*, vol. 24, páginas 377–439, 1992.
- LINDE, C. ET AL. *Life stories: The creation of coherence*. Oxford University Press on Demand, 1993.
- MATTSON, M. P. Pathways towards and away from alzheimer's disease. *Nature*, vol. 430(7000), 2004.
- MOLINA, M., STENT, A. y PARODI, E. Generating automated news to explain the meaning of sensor data. En *International Symposium on Intelligent Data Analysis*, páginas 282–293. Springer, 2011.
- O'ROURKE, N., CARMEL, S., CHAUDHURY, H., POLCHENKO, N. y BACHNER, Y. G. A cross-national comparison of reminiscence functions between canadian and israeli older adults. *Journals of Gerontology Series B: Psychological Sciences and Social Sciences*, vol. 68(2), 2013.
- RAJASEKHARAN, A. A review of bert based models. 2019.
- REITER, E. y DALE, R. Building applied natural language generation systems. *Natural Language Engineering*, vol. 3(1), 1997.
- ROMANO, M., NISSEN, M. D., DEL HUERTO, N. y PARQUET, C. Enfermedad de alzheimer. *Revista de posgrado de la vía cátedra de medicina*, vol. 75, 2007.
- SAI, A. B., MOHANKUMAR, A. K. y KHAPRA, M. M. A survey of evaluation metrics used for nlg systems. *arXiv preprint arXiv:2008.12009*, 2020.
- SHI, L. y SETCHI, R. User-oriented ontology-based clustering of stored memories. *Expert Systems with Applications*, vol. 39(10), páginas 9730–9742, 2012.
- VICENTE, M., BARROS, C., PEREGRINO, F. S., AGULLÓ, F. y LLORET, E. La generación de lenguaje natural: análisis del estado actual. *Computación y Sistemas*, vol. 19(4), páginas 721–756, 2015.



- WANG, A. y CHO, K. BERT has a mouth, and it must speak: BERT as a Markov random field language model. En *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, páginas 30–36. Association for Computational Linguistics, Minneapolis, Minnesota, 2019.

*–¿Qué te parece desto, Sancho? – Dijo Don Quijote –  
Bien podrán los encantadores quitarme la ventura,  
pero el esfuerzo y el ánimo, será imposible.*

*Segunda parte del Ingenioso Caballero  
Don Quijote de la Mancha  
Miguel de Cervantes*

*–Buena está – dijo Sancho –; fírmela vuestra merced.  
–No es menester firmarla – dijo Don Quijote–,  
sino solamente poner mi rúbrica.*

*Primera parte del Ingenioso Caballero  
Don Quijote de la Mancha  
Miguel de Cervantes*

