MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

ANÁLISIS MASIVO DE INFORMES MÉDICOS MEDIANTE TÉCNICAS DE NLP

Autor: Beltrán Rodríguez-Mon Barrera

Director: David Contreras Bárcena

Madrid

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

ANÁLISIS MASIVO DE INFORMES MÉDICOS MEDIANTE TÉCNICAS DE NLP

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2020/21 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido

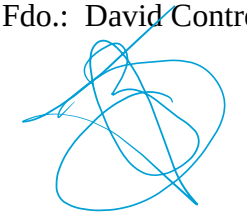
tomada de otros documentos está debidamente referenciada.

Fdo.: Beltrán Rodríguez-Mon Barrera Fecha: 09/07/2021

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Fdo.: David Contreras Bárcena Fecha: 09/07/202



AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESINAS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. BELTRÁN RODRÍGUEZ-MON BARRERA_____
DECLARA ser el titular de los derechos de propiedad intelectual de la obra:
ANÁLISIS MASIVO DE INFORMES MÉDICOS MEDIANTE TÉCNICAS DE NLP_____,
que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar "marcas de agua" o cualquier otro sistema de seguridad o de protección.
b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
e) Asignar por defecto a estos trabajos una licencia Creative Commons.
f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente)*.

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

a) Que la Universidad identifique claramente su nombre como autor de la misma
b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
c) Solicitar la retirada de la obra del repositorio por causa justificada.
d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

El autor se compromete a:

a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.

d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

6º. Fines y funcionamiento del Repositorio Institucional.

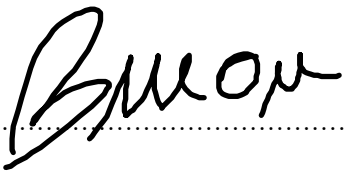
La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusive del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 10 de Julio de 2021

ACEPTA

Fdo……



Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

MÁSTER EN INGENIERÍA DE TELECOMUNICACIÓN

TRABAJO FIN DE MÁSTER

ANÁLISIS MASIVO DE INFORMES MÉDICOS MEDIANTE TÉCNICAS DE NLP

Autor: Beltrán Rodríguez-Mon Barrera

Director: David Contreras Bárcena

Madrid

Agradecimientos

Este trabajo supone el final de una etapa de mi vida que, no se si por suerte o por desgracia, se va a cerrar. Acabar mi etapa de estudios lo veo como un logro y, a la vez, es como irse de ese campamento de verano que uno tanto ha disfrutado sabiendo que nunca se va a repetir.

Este proyecto supone llegar a la meta de esta etapa de 6 años por la que he pasado, y tengo muy claro que, si no hubiera tenido a mi lado a las personas que me han acompañado, a toda la gente que me ha tenido que soportar durante este tiempo, me habría sido imposible llegar hasta aquí.

Gracias a mis padres y a mis hermanos, porque siempre han estado ahí para apoyarme en todas las decisiones (malas o peores) que he decidido tomar. Aun demostrando yo que soy la persona más difícil de aconsejar habida y por haber, ellos nunca se han cansado de guiarme y apoyarme para ser el mejor yo posible.

También quiero agradecer a Nacho y a Fran, que han sido casi parte de mi familia casi desde el primer día que pisé Madrid. Hemos pasado juntos estos 6 años desde el primer día, y sé que sin vuestro apoyo, sin todas esas tardes que hemos pasado juntos, esta etapa no habría sido lo mismo.

De la misma manera quiero dar las gracias a Calama, que desde ese día que nos conocimos en inglés, te he tenido a mi lado para todo lo que he necesitado.

Muchísimas gracias a Pablo y a Ana; desde los primeros días de cuarto he vivido junto a vosotros mil y una cosas. Pablo, desde que decidimos realizar el TFG juntos hasta hoy, puedo afirmar sin lugar a duda que no creo que pudiera hubiera podido pedir un mejor compañero de proyectos y de trabajo. Ana, muchas gracias por aguantar todas las tonterías, por estar siempre ahí cuando he necesitado alguien a quien contarle mis miserias e idas de olla. Gracias a ambos, millones de gracias :] .

Tampoco olvidarme de mis amigos con los que he compartido casi todo este último año. Cayetano, Jorge, Esther, Rodri, Juan, Marina… Habéis hecho de este máster una etapa para recordar (hasta me lo pasaba bien yendo a clase si estabais ahí, quien me lo iba a decir).

También quiero recordar a David. Eres el mejor director que he podido pedir para este proyecto. Siempre dispuesto a recibirme en el despacho, aunque no te haya ni mandado un correo, y escuchar mis problemas y mis conjeturas con este proyecto.

Por último, gracias a todos mis amigos que me estáis apoyando desde A Coruña: Santi, Miguel, Berto, Campos, Poti, Chris, Vila, Viti, Ian y Fran. Aunque solo pise Galicia 4 veces al año, siempre habéis estado ahí para todo.

En definitiva, gracias a todos, porque en mi vida hay una cosa que tengo clara: Sois vosotros quien habéis conseguido que sea la mejor versión posible de mí.

MUCHAS GRACIAS.

ANÁLISIS MASIVO DE INFORMES MÉDICOS MEDIANTE TÉCNICAS DE NLP

Autor: Rodríguez-Mon Barrera, Beltrán.
Director: Contreras Bárcena, David.
Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

RESUMEN DEL PROYECTO

Palabras clave: Big Data, NLP, UMLS, MetaMap

1. Introducción

La importancia del procesamiento de datos no estructurados dentro del mundo Big Data va en aumento, dado que existe un gran porcentaje de información recogida en formatos sin estructura, como textos, imágenes o audios.

Esto ha llevado al desarrollo de una clara nueva tendencia para el tratamiento de este tipo de información. Tanto en empresas como en otros organismos una buena implementación de un sistema para el tratamiento de este tipo de información puede suponer la adquisición de un valor no alcanzable de otras maneras.

2. Definición del proyecto

La propuesta de este proyecto es la elaboración de un sistema basado en técnicas de procesamiento de lenguaje natural (NLP) y la consulta de diccionarios y metatesauros biomédicos (UMLS) que se pueda utilizar de apoyo para los profesionales de la salud y del campo de la biomedicina en el tratamiento de la información masiva sobre recursos de carácter médico.

Los informes serán filtrados para extraer únicamente los datos más relevantes para su análisis posterior y buscando generar fenotipos de los pacientes, asociar las medicaciones con las dolencias observadas e identificar enfermedades a partir de los síntomas mostrados por un paciente.

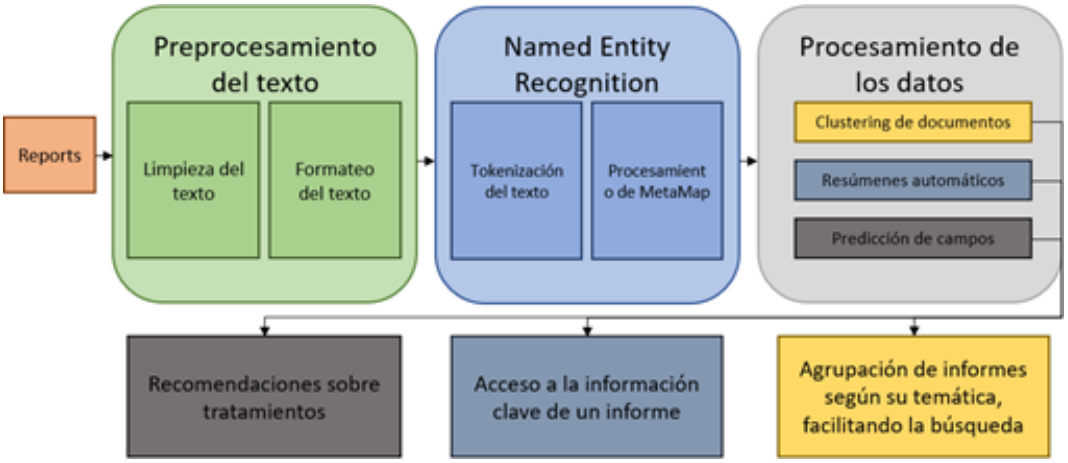
3. Descripción del modelo/sistema/herramienta

El desarrollo de este proyecto se ha dividido en tres bloques funcionales básicos:

Primero de todo, tomando como fuente de los datos los informes médicos, se realiza una limpieza de los datos mediante técnicas propias de Text Mining y NLP.

Tras esto, se lleva a cabo la integración de la aplicación MetaMap para integrar el conocimiento del dominio de la salud y la biomedicina dentro del proyecto.

Por último, se han programado tres módulos de apoyo a los profesionales con el objetivo de minimizar el tiempo dedicado a la búsqueda en el histórico de informes médicos y utilizando diversas técnicas del NLP, mediante la agrupación de informes por temáticas, la generación de resúmenes automáticos y las recomendaciones de acciones a tomar en un caso en concreto.



Pasos en el procesamiento de datos no estructurado

Además, el proyecto se ha desarrollado en *Python* para ser utilizado dentro de un servidor o de un cluster Big Data y cuenta con la integración de un acceso a la aplicación de MetaMap, desarrollada en *Java*.

4. Resultados

El resultado principal de este proyecto es una herramienta de apoyo a los profesionales que ofrece tres servicios complementarios para el tratamiento de la información: un sistema de clustering de documentos para agilizar la búsqueda de la información en históricos de datos, un sistema de creación de resúmenes capaz de mostrar un avance sobre lo que un usuario se podrá encontrar en un documento y un sistema de recomendación teniendo en cuenta de forma simultánea la información de diversos informes al mismo tiempo.

Estos resultados han demostrado servir de gran utilidad para poder disminuir el tiempo de búsqueda dentro del histórico de datos, pero la herramienta no debería de ser utilizada como tomadora de decisiones.

La influencia de la salida obtenida de MetaMap ha demostrado ser de gran utilidad para todos los módulos desarrollados, a su vez que la complementación de los términos obtenidos por la aplicación y la técnica de Embeddings desarrollada ha resultado determinante para que los resultados obtenidos fueran de la mayor calidad posible.

5. Conclusiones

Trabajar con fuente de datos no y en un dominio tan complejo como es la medicina provoca que las herramientas creadas no puedan ser utilizadas en ningún caso como tomadoras directas de las decisiones, y adquieran un enfoque de apoyo para los profesionales.

Además, durante todo el flujo de los datos por los diferentes módulos siempre se ha mantenido una traza sobre el origen de los datos para no omitir o escoinder en ningún caso información a los profesionales, permitiendo el análisis de los textos originales y de los diferentes resultados generados en caso de necesidad.

6. Referencias

[1] M. V. I. K. Milan Kubina, Use of Big Data for Competitive Advantage of Company, 2015. M. Gracia (Nov, 2020). IoT – Internet Of Things.

[2] E. M. Voorhees, «The TREC Medical Records Track,» de National Institute of Standards and Technology.

[3] «National Library of Medicine,» Available: https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html.

[4] National Institute of Health, «National Library of Medicine,» Available: https://www.nlm.nih.gov/research/umls/implementation_resources/metamap.html.

[5] I. A. Martínez, Análisis y optimización del recurso UMLS en la recuperación de la información biomédica mediante métricas de similitud semántica, Madrid, 2016.

ANALYSIS OF MASSIVE MEDICAL REPORTS USING NLP TECHNIQUES

Author: Rodríguez-Mon Barrera, Beltrán.
Supervisor: Contreras Bárcena, David.
Collaborating Entity: ICAI – Universidad Pontificia Comillas

ABSTRACT

Keywords: Big Data, NLP, UMLS, MetaMap

1. Introduction

The importance of unstructured data processing within the Big Data world is increasing, since there is a large percentage of information collected in unstructured formats, such as texts, images or audios.

This has led to the development of a clear new trend for the treatment of this type of information. Both in companies and in other organizations, a good implementation of a system to process this type of information can mean the acquisition of a value not achievable in other ways.

2. Proyect Definition

The proposal of this project is the development of a system based on natural language processing techniques (NLP) and the consult of biomedical dictionaries and metathesauri (UMLS) that can be used to support professionals in the fields of health and biomedicine in the treatment of massive information on medical resources.

The reports will be filtered to extract only the most relevant data for later analysis and seeking to generate phenotypes of the patients, associate the medications with the observed ailments and identify diseases from the symptoms shown by a patient.

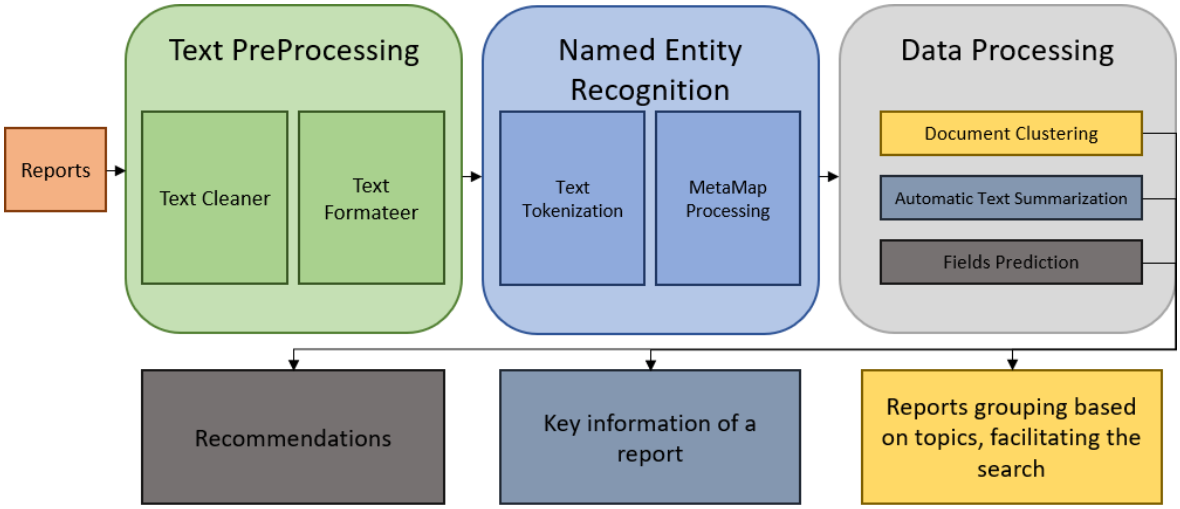
3. Model/System/Tools description

The development of this project has been divided into three functional blocks:

First, taking the medical reports as the source of the data, the data is cleaned using Text Mining and NLP techniques.

After this, the integration of the MetaMap application is carried out to complete the data extracted from the reports with the knowledge of the health and biomedical domains.

Finally, three support modules have been programmed with the aim of minimizing the time spent by professionals searching the medical records history and using various NLP techniques, by grouping reports by subject, generating automatic summaries and recommendations for actions to be taken in a specific case.



Steps in the unstructured data processing

In addition, the project has been developed in *Python* to be used within a server or a Big Data cluster and an integration to access the MetaMap application, developed in *Java*, has been implemented.

4. Results

The main result obtained from this project is a support tool for professionals that offers three complementary services for information processing: a document clustering system to speed up the search for information in data records, a summary creation system capable of showing a preview of what a user can find in a document and a recommendation system simultaneously considering the information from various reports at the same time.

These results have proven to be very useful to reduce the search time within the data history, but the tool should not be used as a decision maker.

The influence of the output obtained from MetaMap has proven to be very useful for all the developed modules, while the complementation of the terms obtained by the application and the Embeddings technique developed has been decisive to obtain high quality results.

5. Conclusions

Working with non-data sources and in a domain as complex as the medicine one means that the tools created cannot be used in any case as direct decision-makers, and therefore they acquire a supportive approach for professionals.

In addition, throughout the flow of data through the different modules a trace has always been kept of the origin of the data, so information is not omitted or hidden to professionals in any case, allowing the analysis of the original texts and the different results generated in case of need.

6. References

[1] M. V. I. K. Milan Kubina, Use of Big Data for Competitive Advantage of Company, 2015. M. Gracia (Nov, 2020). IoT – Internet Of Things.

[2] E. M. Voorhees, «The TREC Medical Records Track,» de National Institute of Standards and Technology.

[3] «National Library of Medicine,» Available: https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html.

[4] National Institute of Health, «National Library of Medicine,» Available: https://www.nlm.nih.gov/research/umls/implementation_resources/metamap.html.

[5] I. A. Martínez, Análisis y optimización del recurso UMLS en la recuperación de la información biomédica mediante métricas de similitud semántica, Madrid, 2016..

Índice de la memoria

Índice de figuras

Índice de tablas

Capítulo 1. INTRODUCCIÓN

En la actualidad existe una inmensa cantidad de información generada por una multitud de fuentes de datos, y esta diversidad de fuentes trae consigo una multitud de formatos que hacen que la extracción de información para adquirir conocimiento sea distinta dependiendo de los casos, yendo desde un proceso sencillo, como por ejemplo la lectura de una tabla en un archivo csv, hasta procesos muy costos como el de la interpretación de información compleja como puede ser un archivo de audio.

La cantidad de datos no estructurados generados en la actualidad es mayor a los que sí están estructurados, dado que éstos primeros tienen naturalezas muy diversas: desde textos como informes, correos electrónicos, hasta otros elementos como imágenes, sonidos y ondas. Esta inmensa cantidad de información, y la necesidad identificada por distintos sectores (ya sea empresas en busca de una ventaja competitiva, hasta organismos públicos que desean contar con las mejores herramientas posibles) ha encaminado el desarrollo de muchas y diferentes formas de abordar este nuevo tipo de datos.

De esta forma, se ha desarrollado una tendencia en cuanto a la captación y procesamiento de esta información. Tanto para las empresas, como se explica en [1], como en otros campos una buena implementación de un sistema de tratamiento de datos no estructurados puede generar un valor añadido muy difícilmente alcanzable.

Este proyecto tratará sobre la elaboración de un sistema de Procesamiento de Lenguaje Natural implementado sobre informes médicos de consultas a pacientes con distintas dolencias. Los textos de las consultas se filtrarán para eliminar las palabras carentes de información y se extraerán las palabras clave con el objetivo de poder generar fenotipos de los pacientes, de asociar las medicaciones con las dolencias de los pacientes y de ser capaces de poder identificar enfermedades a partir de los problemas expresados por el paciente.

La fuente principal de datos son los Reports médicos obtenidos del The Text Retrieval Conference (TREC [2]) en formato xml, los cuales poseen la información de la dolencia principal del paciente (en una frase corta), el identificador de la consulta, códigos de diagnósticos admitidos y descartados, y el propio texto del informe.

Además, estos datos se complementarán con información del metatesauro UMLS (Unified Medical Language System) [3] con el objetivo de identificar de forma inequívoca los términos médicos utilizados en los informes.

Durante todo este proceso se utilizarán diversas técnicas de procesamiento de textos, como librerías para el análisis sintáctico de oraciones o modelos de *Machine Learning* utilizados para generar vectores multidimensionales a partir de las palabras. Todo esto se combinará con técnicas genéricas para el procesado de datos con el objetivo de poder realizar clusters de los distintos informes para agruparlos según la enfermedad del paciente o sus molestias previas antes de acudir a la consulta.

1.1 MOTIVACIÓN DEL PROYECTO

En este proyecto se analizarán de forma masiva historia clínicas de pacientes anonimizados mediante técnicas de Procesamiento de Lenguaje Natural.

Al tratarse de una información no estructurada, habrá que tratar la información utilizando técnicas de *Word Embedding* y se usarán metatesauros con el objetivo de poder añadir contexto a los términos más importantes que aparezcan en los informes.

En este proyecto se utilizarán herramientas propias del tratamiento de datos no estructurados para poder adquirir conocimiento a partir de los informes de las consultas. A partir de los datos adquiridos, se aplicarán técnicas de *Text Mining* para analizar de forma automática todos los textos y extraer la información deseada, y se extraerá el conocimiento deseado utilizando modelos de *Machine Learning* propios del mundo NLP como otros genéricos.

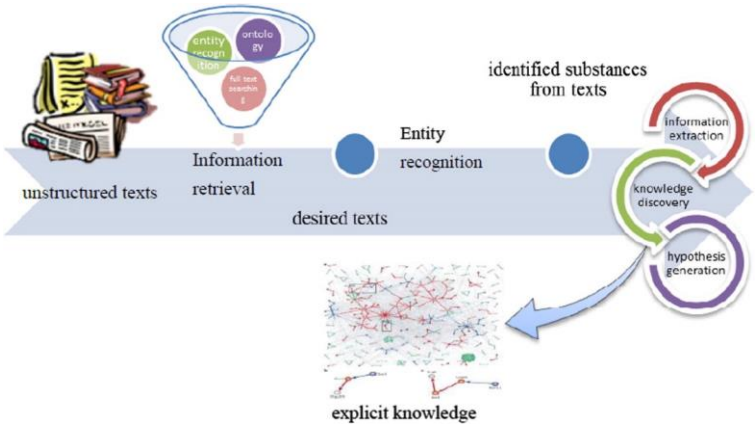


Figura 1. Pasos en el procesamiento de datos no estructurado [4]

Como se muestra en la Figura 1, el proceso del tratamiento de textos parte de la obtención de los datos sin tratar hasta la adquisición de conocimiento utilizando distintas técnicas. El objetivo de este proyecto será elaborar una herramienta que sea capaz de procesar la información de forma automática independientemente de la estructura de los informes, que pueda captar los elementos de valor dentro de los mismo y que, mediante la ejecución de distintos modelos y herramientas, pueda clasificar los documentos, sacar el fenotipo de un paciente o mostrar qué medicamento se podría prescribir según las dolencias del paciente.

De esta forma, se pretende proveer a los servicios médicos de nuevas herramientas de apoyo en su trabajo tanto a la hora de tomar decisiones como agilizando la agregación y búsqueda de información utilizando una base de datos de miles de documentos médicos con información no estructurada, de forma que se pueda proporcionar un sistema capaz de ofrecer al profesional los resultados más similares a un caso concreto con el que se tenga que enfrentar en un momento determinado.

Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS

Big Data:

Colección de datos grande y compleja, difícil de manejar con mecanismos tradicionales. Se puede resumir las características del Big Data mediante las "5 Vs" (Volumen, Velocidad, Variedad, Veracidad y Valor). La complejidad observada en los datos proviene principalmente de su naturaleza no estructurada debido a la fuente generadora de los datos.

NLP (*Natural Language Processing):*

Procesado del Lenguaje Natural – en inglés *Natural Language Processing* – se define como el uso de técnicas y herramientas software para el procesamiento de estructuras de datos no estructurados provenientes directamente del lenguaje humano, tal como los textos o grabaciones de habla. El objetivo principal de este campo es el estudio de las interactuaciones entre el lenguaje humano y las herramientas computacionales, buscando el diseño de técnicas y herramientas para establecer un entendimiento fiable y tecnológicamente eficiente entre estas áreas.

Clúster:

Conjunto de máquinas que están interconectadas entre ellas, utilizan un hardware compartido y se comportan como si fueran un único ordenador. Tienen servicios compartidos y están monitorizados entre sí.

Hadoop:

Framework de procesamiento distribuido y de código abierto que maneja el procesamiento de datos llevado a cabo en clusters Big Data. Este *framework* consta de una estructura de red maestro-esclavo pensado principalmente para usar un hardware de bajo coste y alta disponibilidad.

HDFS (*Hadoop Distributed File System*):

Sistema de almacenamiento de ficheros de manera distribuida. Este sistema cuenta con un almacenamiento redundante, dividiendo los archivos en bloques y almacenándolos en distintas localidades físicas, proporcionando así tolerancia a fallos.

Yarn (*Yet Another Resource Negotiator*):

Plataforma de negociación de los recursos de un clúster Hadoop hacia los nodos que contienen los datos necesarios, para después recoger el resultado obtenido y notificarlo a el programa que lo solicitó.

Jupyter Notebook:

Aplicación web para la creación y compartición de código de análisis, visualización, creación de modelos… pensado principalmente como entorno de desarrollo y test antes de llevar un programa a producción.

Metamap:

"MetaMap es una aplicación altamente configurable desarrollada por el doctor Dr. Alan (Lan) Aronson y el Centro Nacional de Comunicaciones Biomédicas de Lister Hill en la Biblioteca Nacional de Medicina (NLM) para mapear texto biomédico al metatesauro UMLS o, de manera equivalente, para identificar conceptos de metatesauro referidos en texto en inglés. MetaMap emplea un enfoque intensivo en conocimiento, procesamiento del lenguaje natural (NLP) y técnicas lingüísticas computacionales, y se utiliza en todo el mundo en la industria y el mundo académico. En NLM, MetaMap es una de las bases del *Medical Text Indexer* (MTI) de NLM, que se aplica a la indexación semiautomática y completamente automática de la literatura biomédica. Para obtener más información sobre MetaMap y la investigación relacionada, consulte *MetaMap Portal and Indexing Initiative* (II)." [5]

Regex (*Regular Expressions*):

Técnica de descripción de texto basado en patrones. El motor de regex procesa el *String* de entrada de este para, según las distintas funciones disponibles en las librerías utilizadas, procesar el texto. Esta tecnología es muy útil para el filtrado de cadenas de caracteres y la extracción de información de un formato concreto.

NLTK:

"NLTK es una plataforma líder para crear programas Python que funcionen con datos del lenguaje humano. Proporciona interfaces fáciles de usar para más de 50 corpus y recursos léxicos como *WordNet*, junto con un conjunto de bibliotecas de procesamiento de texto para clasificación, tokenización, derivación, etiquetado, análisis y razonamiento semántico, envoltorios para bibliotecas de NLP de nivel industrial, y un foro de discusión activo." [6]

Spacy:

Librería Python gratuita y de código abierto utilizada para el procesamiento avanzado en NLP. Esta librería proporciona herramientas de tokenización, dependencia sintáctica, lematización… que suponen de gran utilidad para realizar un procesamiento complejo de textos de diversas índoles. [7]

Gensim:

Librería de código abierto de entrenamiento de *vector embeddings* mediante algoritmos que utilizan rutinas paralelizables y altamente optimizadas en C y que proporciona múltiples modelos y corpus entrenados con fuentes de datos diversas.

Scikit-learn:

Librería de Python de aprendizaje automático que incluye diversos algoritmos de clasificación, regresión y análisis de grupos y diseñado para funcionar utilizando las librerías "Numpy" y "Scipy". [8]

Transformers:

Arquitectura *sequence-to-sequence* basado en algoritmos de *self-attention* (permitiendo la interactuación entre entradas del modelo) para manejar las dependencias entre las entradas del sistema. Este sistema permite el uso de modelos pre-entrenados y arquitecturas prediseñadas para casos de NLP focalizados en múltiples contextos.

Matplotlib:

Librería de visualización de Python que proporciona una API orientada a objetos para la representación de información y la incrustación de visualizaciones en aplicaciones. [9]

Capítulo 3. ESTADO DE LA CUESTIÓN

Este apartado del Estado del Arte se ha centrado principalmente en un análisis tanto de las áreas de trabajo del mundo de la medicina y del tratamiento de información no estructurada, principalmente mediante técnicas de Procesamiento de Lenguaje Natural, como de las tecnologías ya existentes y conocidas utilizadas para el procesamiento de textos, aproximando éstas a prácticas ya realizadas sobre información de índole médica.

Además, también se comentará dentro de este apartado la importancia de los metatesauros dentro del procesamiento de información no estructurada, y cómo el uso de estas herramientas puede nutrir de una forma muy potente todo el procesamiento realizado.

3.1 INTRODUCCIÓN

La gran cantidad de datos generados en la actualidad es inmensa, y cada día que pasa se crean una increíble cantidad de fuentes y formatos de las que podemos extraer una información que anteriormente parecía casi imposible. Estos datos llegan en numerosas cantidades y de diversas formas, desde una tabla Excel, con sus columnas bien estructuradas, hasta en simple texto plano del que un lector puede conocer cosas nuevas con una simple lectura, pero que aparentemente parece imposible que una máquina lo consiga.

A su vez, hay una gran cantidad de datos de acceso abierto ofrecidos tanto por organizaciones como por empresas, tanto de históricos recopilados para el análisis posterior como de datos en tiempo real sobre la situación en un momento exacto. Otro tipo de datos que acaban siendo disponibles de forma abierta son aquellos generados por propios usuarios de Internet (sin tener que ser participantes de una organización, pero sí de algún grupo o red social).

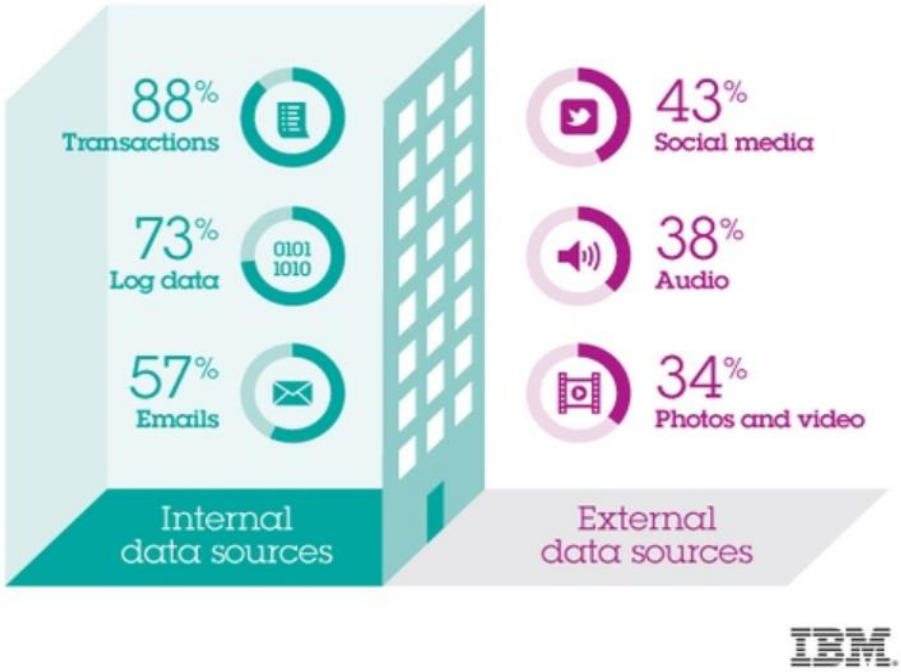


Figura 2. Distintos orígenes de los datos

Además, la cantidad de información almacenada en datos no estructurados crece a pasos agigantados, estimándose que ésta supondrá hasta un 90% de toda la información almacenada por las compañías [10].

La generación y el uso de estos datos está muy extendido en las organizaciones, que han entendido la importancia de almacenar esta información para su uso en diversas aplicaciones. Debido a este hecho, las compañías cada vez cuentan con diversas arquitecturas de almacenamiento de datos en Big Data en los que se guardan ficheros en formatos muy variados.

3.2 FUENTES DE DATOS DE ORIGEN MÉDICO

Profundizando sobre el mundo de la medicina, también existen multitudes de fuentes de datos abiertas accesible para todos los usuarios que lo deseen [11]. Estos sets de datos van desde informes médicos, como por ejemplo todos los ofrecidos por el gobierno de los Estados Unidos sobre consultas de los pacientes, información complementaria sobre los

servicios médicos existentes para las distintas patologías, hasta multitudes de investigaciones sobre diversos campos dentro de la medicina (estudios sobre enfermedades…).

Una de estas fuentes de datos son las reuniones y conferencias de expertos sobre el tema de la medicina, en las que buscan cómo enfrentarse a los sets de datos generados dentro del mundo médico desde una perspectiva "programática". Un caso en concreto es el de TREC (tal como se ha comentado anteriormente), conferencia en la que los expertos afrontan las dificultades de enfrentarse a un set de datos de texto libre y de un vocabulario muy especializado. En su línea de trabajo de la biomedicina los expertos se enfrentan a una terminología especializada y las distintas abreviaciones (no normalizadas dentro del lenguaje, pero altamente conocibles en el mundo de la medicina) hacen que un análisis trivial o genérico no sea suficiente para enfrentarse a estos casos.

Concretizando, las principales características que diferencian un texto médico como los provenientes de esta fuente de datos a un texto de carácter genérico (como podría ser, por ejemplo, un extracto de opinión o divulgación generalista dentro de un diario) podrían ser [12]:

- Construcciones gramaticales sencillas que buscan una comunicación clara y sencilla, sin construcciones complejas y sin darle gran importancia a la correctitud gramática.
- Uso de abreviaturas de forma frecuente y no estandarizada que representan conceptos clínicos.
- Faltas ortográficas debido a la omisión de una corrección del texto en el momento de su digitalización
- Dentro de un formato de datos no estructurada, se presenta información no textual y acompañada de palabras clave como son las mediciones de análisis en un laboratorio, las constantes vitales de un paciente o las cantidades diarias de una medicina.
- Delimitación del contexto de la información mediante estructuras de texto genéricas dentro de los documentos, como pueden ser los subapartados con las que los especialistas han dividido un informe médico a la hora de transcribirlo.

Estas características son las que provocan principalmente la necesidad de afrontar este problema con un enfoque especializado para esta área de trabajo, modificando las técnicas ya existentes para que se adapten a este caso de uso.

3.3 USO DE METATESAUROS

A las características desarrolladas sobre las fuentes de datos de origen médicos tan peculiares anteriormente también hay que sumarles tanto la complejidad como la ambigüedad de los términos utilizados.

Es por esto por lo que, dentro del análisis de documentos médicos se buscará implementar técnicas de optimización de la expansión de las consultas por parte de los usuarios, buscando ampliar el conocimiento sobre los contextos y el filtrado del ruido generado para optimizar las consultas realizadas.

Dentro de este contexto cobran una gran importancia los metatesauros utilizados dentro del ámbito médico. Estos amplios diccionarios están compuestos por multitud de términos y buscan agrupar en estructuras todo el vocabulario y los estándares provenientes, en el caso que nos aplica, del mundo de la salud y la biomedicina.

Dos claros ejemplos de casos de vocabularios controlados utilizados dentro del mundo de la salud y la biomedicina son los de UMLS y MeSH, utilizando el primero de forma muy amplia para la relación de conceptos con su significado y para la creación de estructuras jerárquicas, mientras que el segundo es utilizado para la indexación de documentos médicos y la búsqueda de estos a través de términos.

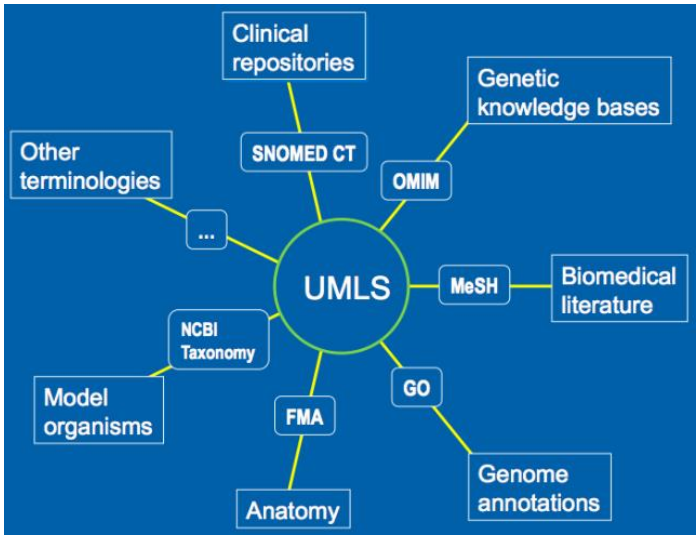


Figura 3. Composición del metatesauro UMLS [13]

Aparte de suponer una estandarización de los términos y la definición formal de los mismos, un metatesauro, mediante la clasificación de los términos en códigos y la indexación y catalogación de la literatura médica. busca establecer las relaciones entre estos términos almacenando toda la información en estructuras de tablas relacionales.

Las relaciones establecidas dentro de los metatesauros son de tipo Semántico, esto es, aproxima los distintos términos almacenados dentro del diccionario según las relaciones existentes entre los significados de cada uno de los términos, tal como se muestra en la Ilustración 4:

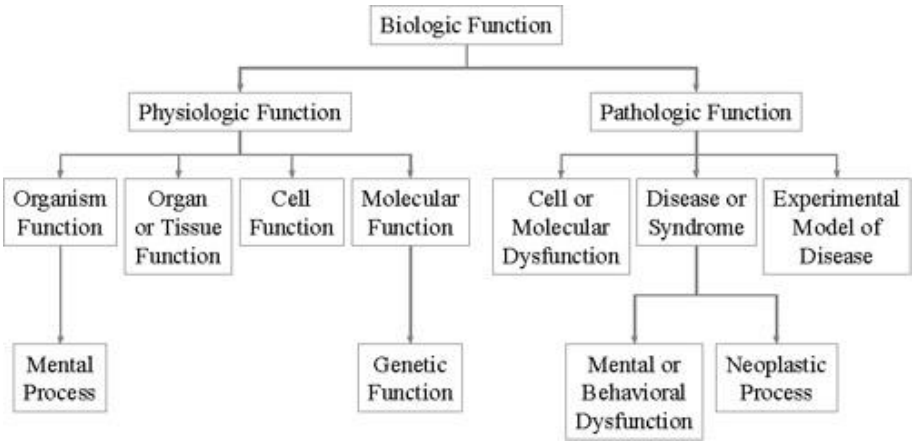


Figura 4. Red semántica del metatesauro UMLS [3]

Profundizando ligeramente sobre el vocabulario que se encuentra dentro de los metatesauros, y tal como se ha comentado anteriormente, se cuenta con una variedad bastante amplia dentro de la gran multitud de términos que nos encontramos dentro de estos diccionarios de más de millones de términos.

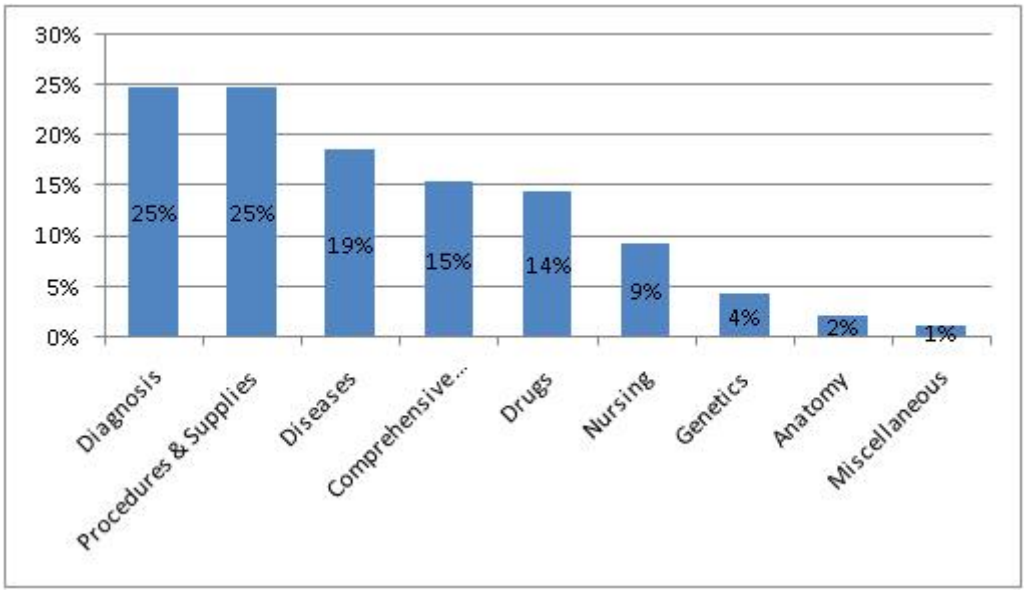


Figura 5. Fuentes del vocabulario contenido dentro del metatesauro UMLS [3]

Por último, ya centrándonos dentro del metatesauro UMLS, el *mapping* de la relación entre los conceptos se hace utilizando los identificadores únicos para cada término dentro del propio diccionario. La organización de los elementos dentro de este metatesauro es en cuatro niveles:

- **CUI** (*Concept Unique Identifier*): Agrupa de forma inequívoca cada concepto, utilizando un mismo código para todos los términos con el mismo significado.
- **LUI** (*Lexical Unique Identifier*): Cada una de las variantes léxicas de un mismo concepto.
- **SUI** (*String Unique Identifier*): Identifica cada una de las maneras de referirse dentro del diccionario a los conceptos recogidos, diferenciando entre minúsculas y mayúsculas, y entre singulares y plurales. Para cada término se escoge un "término preferido" como representación estándar del concepto.

- **AUI** (*Atom Unique Identifier*): Ocurrencia de cada cadena procedente de un recurso del diccionario.

Concept (CUI)	Terms (LUIs)	Strings (SUIs)	Atoms (AUIs) * RRF Only
C0004238 Atrial Fibrillation (preferred) Atrial Fibrillations Auricular Fibrillation Auricular Fibrillations	**L0004238** Atrial Fibrillation (preferred) Atrial Fibrillations	**S0016668** Atrial Fibrillation (preferred)	**A0027665** Atrial Fibrillation (from MSH) **A0027667** Atrial Fibrillation (from PSY)
		S0016669 Atrial Fibrillations	**A0027668** Atrial Fibrillations (from MSH)
	L0004327 (synonym) Auricular Fibrillation Auricular Fibrillations	**S0016899** Auricular Fibrillation (preferred)	**A0027930** Auricular Fibrillation (from PSY)
		S0016900 (plural variant) Auricular Fibrillations	**A0027932** Auricular Fibrillations (from MSH)

Figura 6. Ejemplo de la estructura de términos y conceptos [3]

Se debe de tener en cuenta que el metatesauro no es el vocabulario en sí mismo sino un compendio de los vocabularios utilizados dentro del área de la medicina, y lo que hace es recoger las relaciones jerárquicas entre conceptos de un mismo vocabulario y crea las relaciones entre términos de vocabularios distintos.

Tal como se describe en detalle en [14] y en [15], UMLS ha sido utilizado con éxito en diversas aplicaciones con el objetivo de extraer de los textos no estructurados la información médica relevante para poder ser procesada para el caso de uso correspondiente más adelante.

El origen de estos vocabularios controlados es mayoritariamente de la lengua inglesa (un 62%) pero también incluye terminología de vocabularios de otros idiomas.

3.4 APLICACIONES QUE INTEGREN EL USO DE METATESAUROS

Durante los años se han ido desarrollando multitud de programas que integran metatesauros como el de UMLS o MeSH con el objetivo de realizar de forma automática la relación entre textos médicos con el conocimiento almacenado dentro de los diccionarios. Herramientas como MicroMeSH [16], CHARTLINE [17] o SAPHIRE [18] son utilizadas para el análisis de textos aportando funciones como pueden ser un análisis léxico utilizando vocabulario especializado, análisis sintáctico aprovechando las identidades recogidas en los metatesuaros o búsquedas de relaciones parciales entre términos.

Otra de las aplicaciones desarrolladas aprovechándose de los metatesauros UMLS es la aplicación MetaMap [5], posterior a los mencionados anteriormente, que permite el acceso a los conceptos almacenados en el metatesauro de UMLS. Fundado por el Dr. Alan (Lan) Aronson en la Librería Nacional de Medicina de Estados Unidos -*National Library of Medicine* (NLM)- supone una potente herramienta de mapeo y búsqueda automática de términos médicos y proporciona herramientas de procesamiento automático de archivos de texto identificando los conceptos médicos dentro de un archivo. Este procesamiento lo realiza mediante una combinación de técnicas de Procesamiento de Lenguaje Natural y de técnicas de computación lingüística, demostrando la utilidad de integrar la consulta en metatesauros con las técnicas desarrolladas en el ámbito del NLP para crear herramientas de gran utilidad a la hora de tratar con documentos médicos.

3.5 TEXT MINING

El *Text Mining*, también denominado *Proceso de Extracción del Conocimiento – Knowledge Discovery*, en inglés – son las formas de extraer información desde fuentes de datos no estructurados en formato de texto, y lo que busca este proceso es ser una extensión o alternativa a los procesos de *Data Mining* utilizados comúnmente dentro de los procesos Big Data con formatos de datos estructurados.

Estos procesos de extracción de conocimiento tienen un gran valor debido a la gran cantidad de datos generados actualmente dentro de las compañías en formato de texto no estructurado (aproximadamente un 80%). Pero, a diferencia de los procesos de extracción tradicionales estos métodos son mucho más complejos, debido a la necesidad de tratar información no estructurada y difuminada en todo un documento, ya que no solamente hay que enfrentarse a la captura de palabras directamente como si un campo se tratara, sino que estos tipos de datos contienen información "almacenada" dentro de la semántica o la sintaxis de una frase. Debido a esto no toca enfrentarse únicamente al procesamiento de datos como el de una tabla; la información queda almacenada dentro del "sentido" o el "contexto" de todo el texto. Identificar las relaciones entre frases, recursos semánticos como la ironía o el sarcasmo o conocer el sentimiento inherente de un texto son algunas de las necesidades con las que los encargados de programar herramientas de este tipo podrán encontrarse a la hora de procesar esta información.

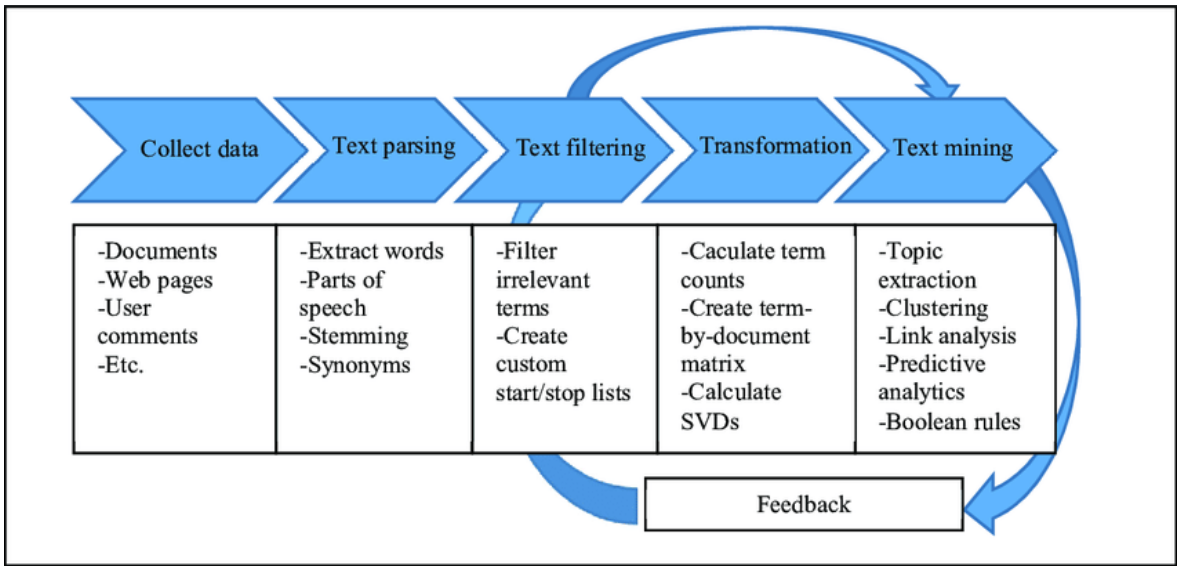


Figura 7. Procesos de Text Mining [19]

La minería de texto se puede definir como un procesamiento en dos fases:

El primero consiste en el refinamiento del texto, transforma los documentos de texto con una estructura libre en una forma intermedia, y la segunda parte es la extracción del conocimiento que se deduce tras la obtención de los patrones intermedios.

La forma intermedia puede ser semiestructurada, tal como una representación gráfica conceptual, o estructurada representando datos relacionales. Además, ésta puede basarse en documentos donde cada entidad representa un documento y el objetivo es obtener las relaciones entre documentos, o basada en conceptos donde cada entidad es un concepto de intereses en un dominio específico.

De forma resumida, se podría afirmar que una extracción basada en documentos es independiente del dominio, mientras que la basada en conceptos se ha de desarrollar pensando en un dominio en concreto.

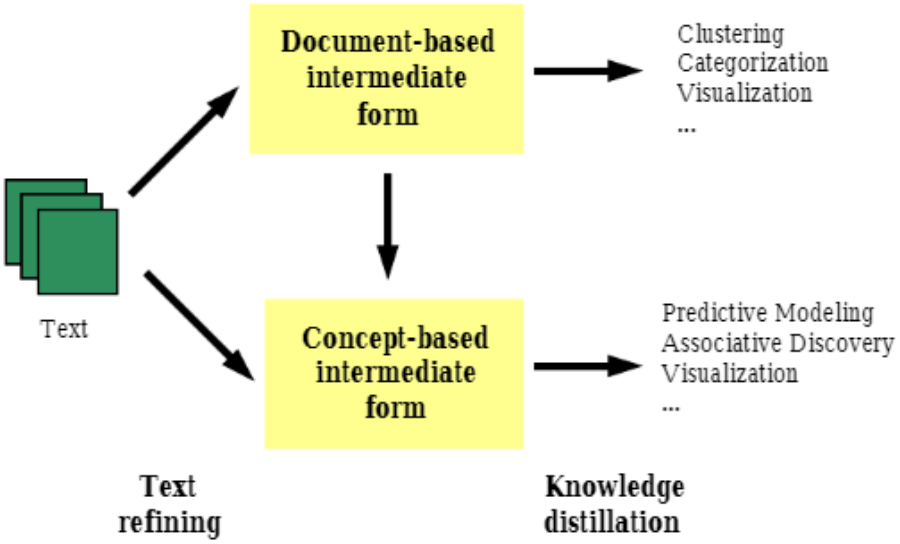


Figura 8. Text Mining basada en documentos y conceptos [20]

Además, existe la posibilidad de pasar de una forma intermedia basada en documentos a una basada en conceptos a través de la extracción de objetos relevantes de los documentos utilizados para un dominio en concreto.

Por poner un ejemplo, utilizando un set de datos sobre artículos de revistas, primero se podría realizar un refinamiento basado en documentos, para que a continuación se lleve a cabo una extracción de conocimientos basado en documentos con el fin de organizar los artículos según su contenido para visualizarlos. Por otro lado, para el descubrimiento de

conocimientos en un dominio específico, se puede trasladar la forma intermedia obtenida a otra basado en conceptos mediante la búsqueda de términos de un dominio concreto, según el requisito de la tarea (por ejemplo, en este caso se podría buscar elementos del concepto "empresa"), y con esta nueva forma también se podría elaborar un modelo predictivo sobre las características principales del área de búsqueda.

Concretizando sobre las herramientas ya existentes, multitud de empresas han desarrollado aplicaciones de *Text Mining* con multitud de posibilidades de uso que se categorizan utilizando conceptos explicados en este apartado, definidas en la Tabla 1:

Compañía	**Producto**	**Refinamiento del texto**	**Forma intermedia**	**Extracción de conocimiento**
Cartia	ThemeScape		Document-based	Clustering visualization
Canis	cMap		Doc-based Word histogram	Clustering visualization
IBM	iMiner	Info retrieval summarization	Document-based	Clustering categorization
Inxight	Linguist	Info retrieval text analysis summarization	Document-based	
Knowledge Discovery System	ConceptExplorer	Info retrieval	Concept-based	

TextWise	Dr_Link Cindor Chess	Info retrieval info extraction	Concept-based	
Cambio	Data Junction	Info extraction	Concept-based	
Megaputer	TextAnalyst	Info retrieval summarization	Document-based Semantic net	

Tabla 1. Aplicaciones de Text Mining

Con todas estas opciones disponibles, y más herramientas creadas específicamente para Text Mining, se abre un mundo de posibilidades para que desarrolladores elaboren distintas aplicaciones para el tratamiento de todos los datos disponibles (tanto en la compañía como de forma abierta para todo el mundo) y de este modo ayudar a los expertos a llevar a cabo una mejor toma de decisiones.

Aún, así, el proceso de la minería de texto trae consigo también diversos problemas con los que hay que enfrentarse a la hora de desarrollar una herramienta que necesite de estas técnicas. Algunos de estos desafíos son:

- **El formato intermedio de los datos**: La complejidad dentro del formato intermedio generado dependerá directamente del propósito del procesamiento del texto**.** En un caso de uso de nicho podría ser necesario llevar a cabo un enriquecimiento semántico del contenido, lo que puede llegar a ser muy costoso.
- **Refinamiento de texto multi-idioma**: Aunque en la teoría la minería de texto es independiente del lenguaje, existen multitud de herramientas que están más y mejor desarrolladas para el inglés, como es el caso de las herramientas de análisis sintácticos o los *tokenizers* de palabras.
- **Integración del conocimiento sobre un dominio concreto**: Conocer el área de trabajo supone una gran ventaja en todo este proceso (desde los primeros pasos de refinamiento) dado que te permite enfocar el uso de las herramientas de una mejor manera y optimizar todo el *pipeline* de procesamiento de los datos.

- **Creación de sistemas autónomos personalizados**: Actualmente, los procesos de minería de datos son realizados como la unión de técnicas y tecnologías en búsqueda de un caso de uso en concreto, lo que supone la necesidad de un experto que conozca tanto el dominio de trabajo como las diversas herramientas que se pueden utilizar para cada caso. Un objetivo en esta área es ser capaz de crear herramientas que puedan ser configurables no solo por los programadores, sino que también sean los profesionales del área de negocio capaces de diseñarlos aplicando su conocimiento dentro de un dominio de trabajo.

En [37] se habla en detalle sobre el potencial en la combinación de diversas técnicas de Text Mining y las limitaciones encontradas en este proceso a la hora de enfocar estas técnicas para el tratamiento de textos médicos.

3.6 MACHINE LEARNING EN NLP

Analizando los textos y procesándolos de distintas maneras como se ha ido explicando durante este estado del arte, se puede llevar a cabo la elaboración de distintos modelos de *Machine Learning* como propósitos muy diversos. Desde aplicaciones como el análisis del sentimiento hasta el desarrollo de *chatbots* que interactúan con personas de la misma forma que lo haría un humano, existen multitud de aplicaciones en las que integrar este tipo de tecnologías ha demostrado tener una gran utilidad.

Uno de los mayores desafíos dentro del mundo del Procesamiento del Lenguaje Natural es la capacidad de propagar el conocimiento inferido de una parte del texto al resto de este, o de relacionar conceptos y contextos entre distintas frases de un mismo texto.

Para solucionar estos problemas, las técnicas de NLP han evolucionado desde el uso de las *Recurrent Neural Networks (RNNs)* hasta la creación de las *Long Short-Term Memory Networks (LSTMs)*.

Las **RNNs** son variantes de las tradicionales redes neuronales típicas con conexión total entre capas añadiendo un componente de memoria dentro de las mismas. Se llaman

recurrentes dado que siempre realizan el mismo procesamiento para todas las entradas a las neuronas, siendo la salida dependiente del procesamiento de las anteriores neuronas. El componente añadido es la memoria que le permite almacenar conocimiento adquirido por el procesamiento de las entradas anteriores, modificando así su funcionamiento normal

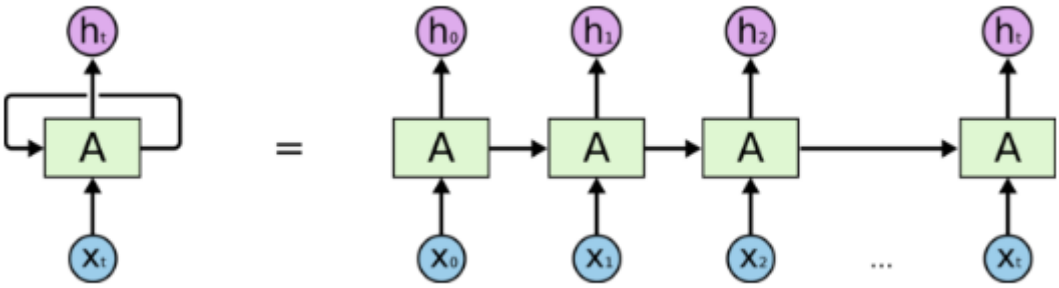


Figura 9. Esquema de RNN [21]

Por otro lado, existen las redes **LSTMs**, que suponen una modificación de las redes RNN dado que estas segundas tienen problemas para recordar características de datos lejanos en el tiempo (esto es, cuando han procesado muchos datos, el conocimiento recordado de los primeros se va "difuminando" en el tiempo hasta desaparecer). Es por esto por lo que el uso de estas redes está más extendido a la hora de trabajar con datos de mucha longitud o de tamaño indefinido.

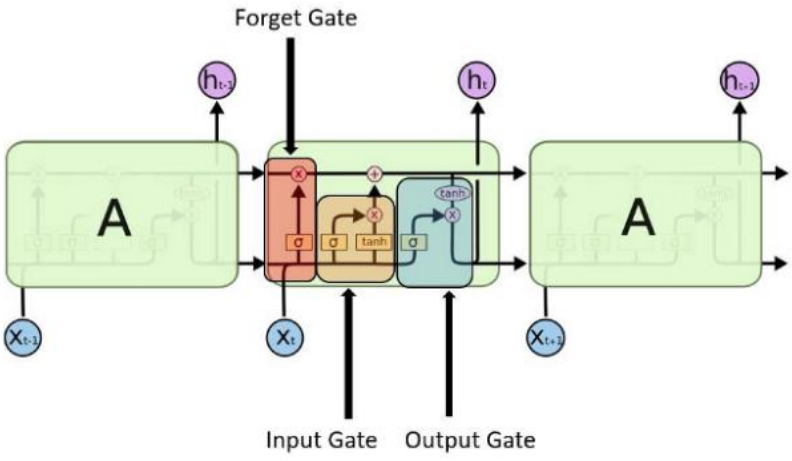


Figura 10. Esquema de LSTM [21]

Las redes LSTM y sus variaciones parecían ser la respuesta al problema del desvanecimiento de los gradientes. Sin embargo, existe una limitación en la cantidad de información que se puede guardar, debido a la compleja secuencia entre neuronas. Esto limita la longitud de las secuencias que una red LSTM puede recordar a unos pocos cientos de palabras.

Otro problema adicional son los altos requisitos computacionales de esta red. Debido a su naturaleza secuencial, son difíciles de paralelizar, lo que limita su capacidad para aprovechar los recursos informáticos modernos como GPU y TPU.

3.7 BERT Y LOS GRANDES MODELOS YA ENTRENADOS

Por último, la apuesta de las grandes compañías tecnológicas por modelos que trabajen con textos como fuente principal de datos, es muy alta [22]: Facebook ha creado un *chatbot* de código abierto (*BlenderBot*) [23] que es capaz de mantener una conversación con un ser humano de forma fluida y tratando varios temas en las mismas oraciones, y que ha sido entrenado con multitud de textos (9.4 mil millones de parámetros).

También Google tiene una creación propia y de un alto valor dentro de este campo. Ingenieros dentro de Google AI han creado un modelo llamado BERT [24], actualmente también de código abierto, para el tratamiento de textos utilizando una arquitectura de redes neuronales con el objetivo de dar un modelo pre-entrenado que sea bidireccional (esto es, que pueda interpretar tanto el contexto que acompaña a una palabra anterior como posteriormente) y que puedan ser entrenados (*fine-tuning*) para multitud de aplicaciones de NLP.

El modelo BERT utiliza el concepto de *Mecanismos de Atención*. Un *Transformer* codifica la información en vectores de palabras sobre lo importante del contexto, y este mecanismo le permite al modelo focalizar en palabras concretas cuáles son las partes importantes tanto del dato de entrada como de la secuencia de salida generada.

Este modelo utiliza una arquitectura de multi-atención por cabeceras, esto es, produce múltiples sets de representaciones distintas para cada una de las entradas al modelo, codificando a través de éstas diferentes características de la entrada, tal como se explica en [21].

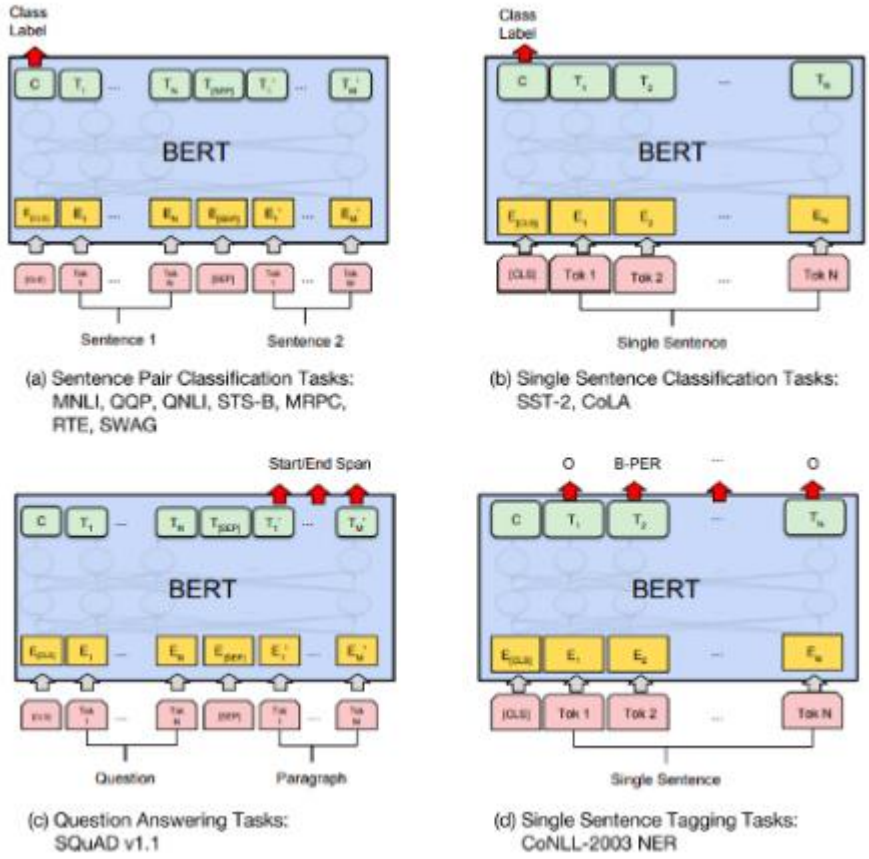


Figura 11. Esquema de BERT [21]

Existen dos modelos de BERT introducidos en el paper con el que fue presentado [24]: BERT Base, de 12 capas y que utiliza 12 cabeceras para el mecanismo de atención, contando con 110 millones de parámetros en total, y BERT Large, con 24 capas y 16 cabeceras, y llegando a los 340 millones de parámetros.

A partir de estos dos modelos "base", comenzaron a crearse multitud de variaciones de ambos para cumplir con las necesidades de diferentes casos de uso. Es así la forma en la

que modelos como ALBERT, RoBERTA, TinyBERT, DistilBERT, and SpanBERT vieron la luz.

Otro de los modelos creado a partir de BERT fue el de SciBERT, presentado en 2019 desde el instituto Allen de inteligencia artificial de Seattle, Estados Unidos. La razón de ser de este modelo es, según sus desarrolladores, debido a "la falta de un set de datos científicos etiquetados de gran escala y alta calidad", tal como se comenta en [25]. SciBERT está entrenado sobre un gran set de datos multi dominio dentro de las publicaciones científicas buscando mejorar el rendimiento de diversas tareas de Procesamiento de Lenguaje Natural a realizar:

- ***Named Entity Recognition*** (NER)
- ***PICO Extraction*** (etiquetado de secuencias) (PICO)
- **Clasificación de textos** (CLS)
- **Clasificación Relacional** (REL)
- ***Parsing* de Dependencias** (DEP)

La técnica del *fine-tuning* sobre estos complejos modelos ya creados por las grandes organizaciones es un clásico dentro del desarrollo en Machine Learning, y como se ve se aplica también en el mundo del Procesamiento de Lenguaje Natural. Pero la utilidad de esta técnica no es solamente útil al utilizar modelos complejos como es el BERT, sino que también puede utilizarse sobre elementos más sencillos.

Durante todo el estado del arte de este proyecto se han visto multitud de tecnologías y técnicas que se pueden utilizar en el procesamiento de textos, las aplicaciones de éstas dentro de las herramientas de algunas organizaciones y su modo de funcionamiento. Con todo esto, la idea más importante a recalcar a la hora de implementarlas dentro de un caso de uso de procesamiento de texto es la importancia del conocimiento del dominio de trabaja para saber qué utilizar y cómo adaptarlo para sacar el máximo potencial al proyecto desarrollado.

Capítulo 4. DEFINICIÓN DEL TRABAJO

4.1 JUSTIFICACIÓN

Tal como se ha ido mencionando anteriormente, el mayor atractivo de este proyecto es la integración de las nuevas tecnologías de Procesamiento del Lenguaje Natural para el tratamiento de informes médicos con el principal objetivo de crear una herramienta que suponga un apoyo al trabajo diario del médico.

El objetivo de este proyecto nunca será el de sustituir el factor humano del área de la medicina, dado que un programa como éste, por muy preciso que pueda llegar a ser, deberá siempre de tomar en cuenta los análisis realizados por una persona ya formada en el área. Es por esto por lo que principalmente se ha buscado elaborar herramientas que puedan dar soporte a través del tratamiento de datos masivos que puedan reducir el tiempo que el médico ha de dedicar a labores de búsqueda de información.

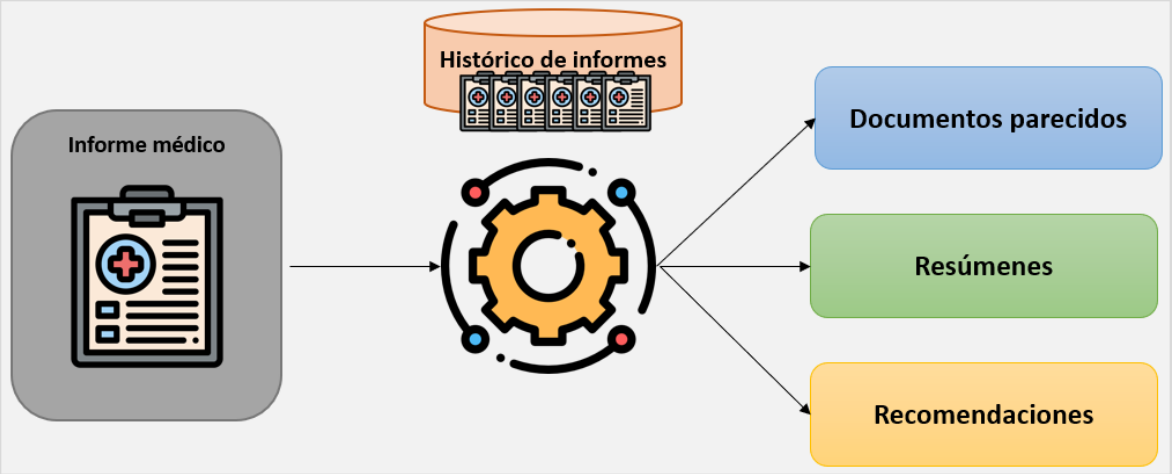


Figura 12. Esquema del proyecto

Además, en este proyecto se ha buscado integrar diferentes tecnologías y técnicas, unas propias del mundo de la medicina y otras más generalistas, para poder ofrecer al sector de la medicina una herramienta de utilidad que se haya podido nutrir de diferentes

experiencias provenientes tanto de médicos como de otros sectores que utilicen el tratamiento de textos.

Es por esto por lo que en el proyecto cobra gran importancia el uso de la aplicación de MetaMap (explicada y referenciada en el apartado de *Tecnologías*) para aportar todo el valioso conocimiento proveniente del metatesauros UMLS al análisis de los documentos y, de esta forma, poder establecer relaciones mucho más certeras.

4.1.1 DATOS UTILIZADOS

La principal fuente de datos utilizados en este proyecto son los más de 100.000 extractos de informes médicos obtenidos de *The Text Retrieval Conference*, en formato .xml y que se han generado, como se indica en la figura contigua, como un compendio de las distintas visitas generadas:

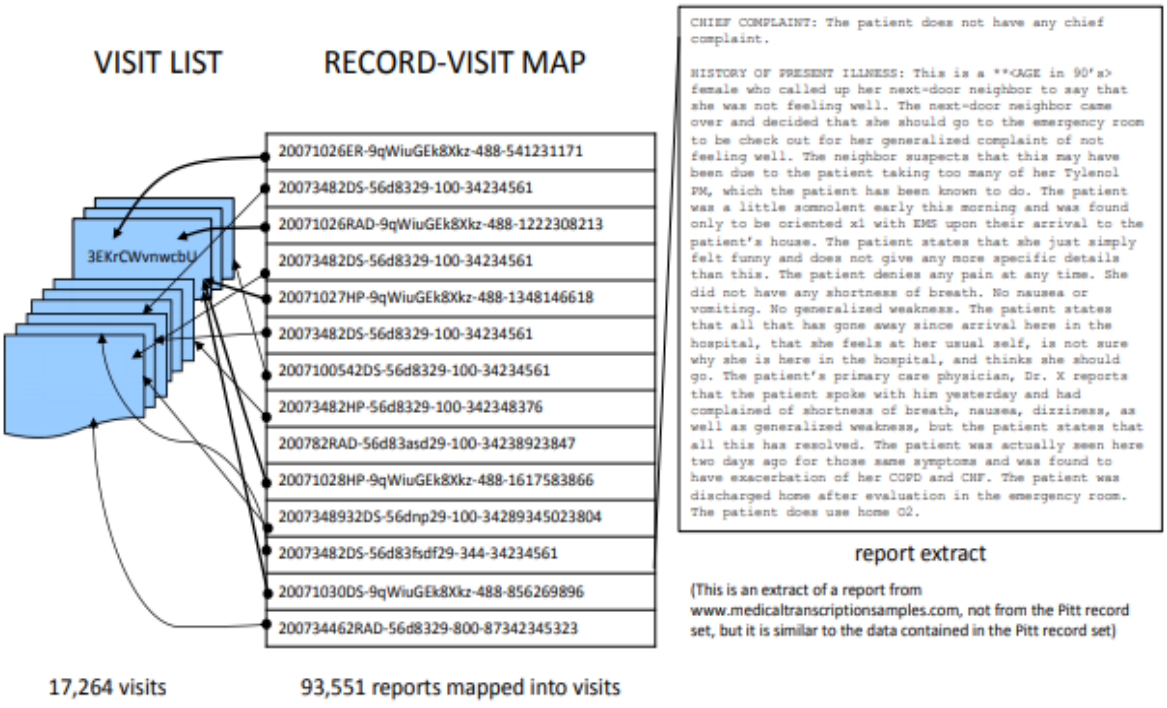


Figura 13. Recopilación de las visitas en extractos médicos [2]

Estos archivos contienen información sobre la dolencia principal del paciente en una frase corta, el identificador de la consulta, códigos de diagnósticos admitidos y descartados, y el propio texto del informe.

A la hora de empezar a trabajar con estos archivos, hay que enfrentarse a una estructura como la siguiente:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<report>
        <checksum>HASH</checksum>
        <subtype>STRING</subtype>
        <type>STRING</type>
        <chief_complaint>STRING</chief_complaint>
        <admit_diagnosis>INT</admit_diagnosis>
        <discharge_diagnosis>INT</discharge_diagnosis>
        <year>YYYY</year>
        <downlaod_time>YYYY-MM-DD</downlaod_time>
        <update_time/>
        <deid>v.__</deid>
        <report_text>REPORT_TEXT</report_text>
</report>
```

En estos archivos encontramos metadatos sobre la generación del propio fichero (*checksum, year, download_time, update_time, deid)* y otros que buscan definir la clase de fichero según su contenido, como son:

- *Admit_diagnosis* & *Discharge_diagnosis*: Códigos ICD [26] del report, esto es, resumen en forma de códigos de las principales dolencias y problemas identificados a un paciente en forma de códigos.
- *Type* & *Subtype*: Indica el tipo de report mediante siglas, en concreto:
    - Radiology Reports

- History and Physicals
- Consultation Reports
- Emergency Department Reports
- Progress Notes
- Discharge Summaries
- Operative Reports
- Surgical Pathology Reports
- Cardiology Reports

- *Chief_Complaint*: Texto libre para resumir la dolencia principal por la cual el paciente ha decidido asistir a una consulta.
- *Report_text*: Recopilación de los informes de las visitas de un paciente acumuladas en el extracto generado, y explicado más profundamente en el apartado 4.1.1.1.

Por último, tal como se ha comentado, cada documento es la acumulación de reports generados por cada una de las visitas de un paciente, quedando el set de datos distribuidos tal como se muestra en la figura siguiente:

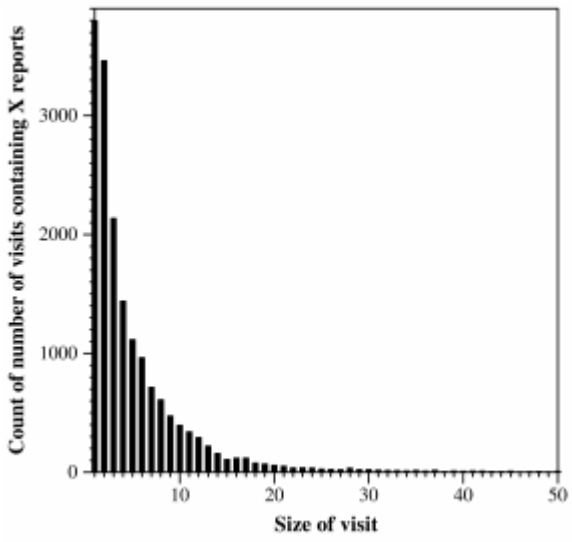


Figura 14. Acumulación de visitas por extracto [2]

4.1.1.1 Campo Report_text

Este campo, como se ha ido explicando a lo largo de esta memoria, supone los datos principales con los que se han trabajado durante este proyecto.

Tanto el tamaño como la estructura de este campo del archivo es variable y está generado como se explicó anteriormente, suponiendo la recopilación de los informes generados por médicos sobre cada uno de los pacientes.

Esto supone que, para la elaboración de los modelos de Machine Learning y el procesamiento mediante técnicas NLP este campo suponga de gran importancia, dado que contiene todos los análisis y recomendaciones realizadas por los expertos para cada uno de los pacientes, y un correcto tratamiento de esta información posibilitará la extracción de conocimiento para poder llevar a cabo todas las recomendaciones deseadas.

Para poder trabajar con estos datos, aunque ya se desarrollará con detalle más adelante en el capítulo sobre el *Sistema Desarrollado*, se buscará estructurar lo máximo posible el texto de los informes siendo capaz de dividirlo entre el análisis previo de un paciente, su curso durante un ingreso o el diagnóstico realizado y su explicación.

Un correcto procesamiento previo de estos datos permitirá dividir el texto en la estructura deseada, la captura de los elementos del texto que poseen la información principal del mismo y la elaboración de componentes matemáticas de representación de los documentos lo más precisa posible.

4.1.2 METAMAP

Esta herramienta ya explicada anteriormente se ha implementado dentro del *pipeline* del proyecto encapsulada dentro de una API a la que, a la hora de procesar los documentos, se le llamará para que lleve a cabo un procesamiento del texto y pueda *taggear* el documento.

Aunque ya se desarrollará en detalle esta herramienta posteriormente, la finalidad por la cual se utiliza es la de integrar el metatesauros UMLS de una forma sencilla y eficaz dentro de la herramienta elaborada. Este programa encapsula dentro de sí todo el diccionario de

términos con sus correspondientes códigos, es capaz de analizar automáticamente un texto (en formato String) y dividirlo en palabras o grupos asignándoles un CUI (identificador) y explicando la entidad a la que hace referencia cada término (por ejemplo, identificar el término "brazo" como "extremidad").

4.2 OBJETIVOS

El objetivo de este proyecto es la utilización de técnicas de Procesamiento de Lenguaje Natural para extraer información de forma automática sobre informes médicos con la intención de poder utilizar distintos modelos para adquirir conocimiento sobre los informes. Utilizando herramientas propias del NLP, las proporcionadas por la aplicación de MetaMap y modelos de Machine Learning se busca ser capaz de clasificar los documentos según las relaciones identificadas, poder realizar predicciones sobre los mismos también ser capaz de la elaboración de resúmenes automáticos de informes, identificando las partes más importantes de los mismos.

De esta forma, los objetivos de este proyecto quedan marcados de la siguiente forma:

- **T1. Procesado y Limpieza**: Formateo de los textos y filtrado de las palabras para eliminar aquellas que no posean información relevante y para llevar a la raíz todos los términos, apoyándonos en análisis sintácticos de las oraciones.
- **T2. Named entity recognition**: Extracción de términos médicos de los textos mediante el uso de metatesauros UMLS utilizando la herramienta MetaMap, para clasificarlos y obtener relaciones entre los mismos.
- **T3. Desarrollo de los modelos**: Desarrollo de modelos de Clustering y búsquedas de términos en documentos mediante modelos de Machine Learning no-supervisados.
- **T4. Summarization**: Obtención de resúmenes de documentos de forma automática utilizando técnicas de identificación de la relevancia de cada parte del documento y también mediante la elaboración de modelos (pre-entrenados o desde cero) para llevar a cabo la creación de resúmenes.

- **T5. Predicciones de medicamentos**: Utilización de modelos no supervisados de Machine Learning con el objetivo de predecir los posibles medicamentos a recomendar al médico para que sean recetados a partir de los datos extraídos de los textos.

4.3 METODOLOGÍA

El primer mes del proyecto se utilizarán para familiarizarse los datos que se van a utilizar y la identificación de las mejores librerías y modelos para preparar un sistema automático para el tratamiento de los informes.

Tras esto la metodología a utilizar será un plan cíclico en el que se ejecutarán distintas labores del procesamiento de texto y la elaboración de modelos.

La primera fase del proyecto consistirá en la automatización del procesamiento del texto, realizando un filtrado de palabras e intercambiando los términos restantes por su raíz.

Tras este primer procesado, podremos extraer los términos médicos más relevantes utilizando el metatesauros UMLS.

Estas primeras fases nos permitirán crear el primer entregable del proyecto, que serán los resúmenes de textos mediante el uso de los términos extraídos.

Utilizando técnicas de *Word-Embedding*, podremos proceder a los procesos de clasificación de textos utilizando modelos no supervisados propios del NLP.

La última fase por desarrollar será la de realizar modelos que nos permitan predecir, mediante el análisis de los informes, ciertas causas de las dolencias o posibles medicamentos para tratar a un paciente en concreto.

Cada iteración tendrá una duración estimada de dos meses. Mientras que en la primera iteración se seguirá el orden definido, a partir del segundo ciclo se elegirá qué pasos

seguir según las necesidades del proyecto, buscando complementar de la mejor forma posible el sistema ya elaborado.

4.4 PLANIFICACIÓN Y ESTIMACIÓN ECONÓMICA

4.4.1 PLANIFICACIÓN

Como ya se ha comentado en el apartado de Metodología, este proyecto se ha realizado siguiendo una planificación cíclica, similar a un *Sprint* en una metodología *Agile* (aunque de pocos *Sprints*). El motivo principal de esta decisión fue que, tras un análisis inicial previo al comienzo del proyecto, se identificó la necesidad de volver a los puntos de desarrollo tras una primera iteración para poder adaptarlos a las nuevas necesidades de desarrollo identificadas a lo largo de la realización del proyecto.

De esta forma, la planificación del proyecto queda definida de la siguiente manera:

FASES	ENERO				FEBRERO			
	S1	S2	S3	S4	S1	S2	S3	S4
Toma de contacto con los datos								
Pre-procesado del texto								
Reconocimiento de términos								
Creación de resúmenes								
Clustering de Documentos								
Modelos de Predicción								
Documentación								

Tabla 2. Planificación Enero - Febrero

FASES	MARZO				ABRIL			
	S1	S2	S3	S4	S1	S2	S3	S4
Toma de contacto con los datos								
Pre-procesado del texto								
Reconocimiento de términos								
Creación de resúmenes								
Clustering de Documentos								
Modelos de Predicción								
Documentación								

Tabla 3. Planificación Marzo - Abril

FASES	MAYO				JUNIO			
	S1	S2	S3	S4	S1	S2	S3	S4
Toma de contacto con los datos								
Pre-procesado del texto								
Reconocimiento de términos								
Creación de resúmenes								
Clustering de Documentos								
Modelos de Predicción								
Documentación								

Tabla 4. Planificación Mayo - Junio

4.4.2 ESTIMACIÓN ECONÓMICA

El presupuesto necesario para la ejecución del proyecto Big Data de análisis de textos no estructurados se puede dividir en dos partes claramente diferenciadas: Los equipos utilizados para el procesamiento informático y la mano de obra

Elementos utilizados	***€/ud***	***uds***	***total***
Clúster ICAI	80.000	1	80.000
Dell XPS 13''	1.400	1	1.400
PC sobremesa	1.250	1	1.250
		TOTAL	82.650

Tabla 5. Estimación económica de los recursos materiales utilizados

Partes del proyecto	***horas***	***€/hora***	***total***
Estudio previo	40	7	280
Pre-Procesamiento	80	8,25	660
MetaMap	30	5	150
Clustering	100	10	1.000
Resúmenes	40	9	360
Predicciones	40	10	400
		TOTAL	2.850

Tabla 6. Estimación económica de los recursos humanos utilizados

Capítulo 5. SISTEMA/MODELO DESARROLLADO

Tal como se ha ido explicado anteriormente, este proyecto se basa en el tratamiento de archivos de informes médicos sobre pacientes escritos en inglés y almacenados en archivos .xml.

Mediante la aplicación de técnicas y herramientas de Procesamiento de Lenguaje Natural, tratamiento de variables en formato *String* y modelos de Machine Learning se tratarán los archivos de entrada y, tras cruzar la información obtenida con el resultado obtenido de la respuesta de la aplicación MetaMap, se utilizarán los modelos expuestos para generar de esta forma distintas herramientas que sirvan de apoyo a los profesionales del sector médico. Este proceso queda explicado en la figura siguiente.

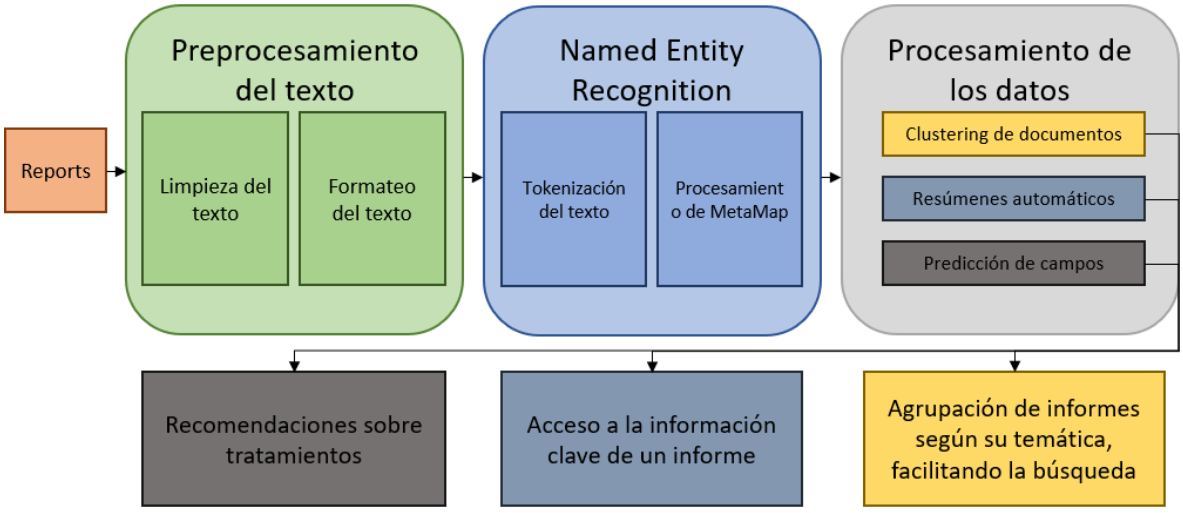


Figura 15. Esquema del sistema desarrollado

Además, todas estas herramientas programadas, así como la aplicación MetaMap, serán ejecutadas en el Cluster de la universidad, cuyas características quedan definidas a continuación:

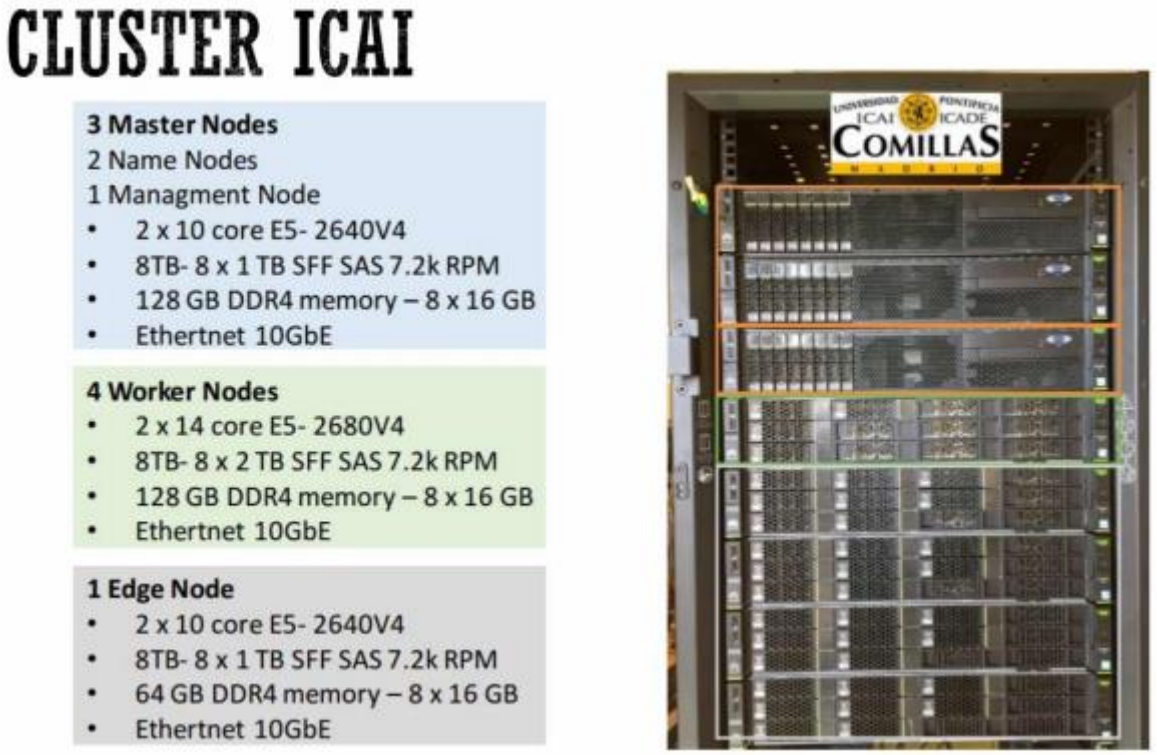


Figura 16. Características del Cluster ICAI

5.1 PROCESAMIENTO DEL TEXTO

El primer paso desarrollado dentro de este proyecto es el del procesamiento del texto. Para la programación de esta parte se ha de tener en cuenta que el formato del texto recibido es libre. El objetivo de este apartado será conseguir obtener una estructuración del texto lo más alta posible, para poder tratar este mismo en los apartados siguientes con las mayores facilidades que se hayan conseguido obtener.

Lo primero de todo, se ha de leer cada uno de los archivos de los informes, almacenados como *'reportXXXXX.xml'* dentro del directorio principal (para este proyecto se contará con más de 100.000 informes).

Aunque ya se comentó dentro del apartado *4.1.1. Datos Utilizados*, en el siguiente bloque de código se puede observar cómo es uno de estos archivos que componen nuestro set de datos:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<report>
<checksum>20070618RAD-2RXdmXr6vXMo-848-1459314475</checksum>
<subtype>NM</subtype>
<type>RAD</type>
<chief_complaint>MOUTH PAIN</chief_complaint>
<admit_diagnosis>873.62</admit_diagnosis>
<discharge_diagnosis>873.60,E920.8,493.90,</discharge_diagnosis>
<year>2007</year>
<downlaod_time>2009-08-18</downlaod_time>
<update_time/>
<deid>v.6.22.07.0</deid>
<report_text>[Report de-identified (Safe-harbor compliant) by De-ID
v.6.22.07.0]

EXAMINATION PERFORMED:
CT HEAD W CONTRAST CTA   **DATE[Jun 18 07]     0513 HOURS

CLINICAL HISTORY:
Subacute stroke.

COMPARISON:
None.

FINDINGS:
A CT angiogram of the neck is unremarkable. There are no focal
stenoses of the cervical portions of the common carotid arteries,
carotid bifurcations and cervical portions of the bilateral
internal carotid and the bilateral vertebral arteries. The origins
of the bilateral vertebral and common carotid arteries are
unremarkable without significant stenoses.

CT angiogram of the head is unremarkable. There are no areas of
stenosis or occlusion. There are no gross changes of vasculitis.
No gross aneurysms are noted in the region of the circle of
Willis.

IMPRESSION:
1. UNREMARKABLE CT ANGIOGRAM OF THE HEAD.
2. UNREMARKABLE CT ANGIOGRAM OF THE NECK.

My signature below is attestation that I have interpreted
this/these examination(s) and agree with the findings as noted
above.

END OF IMPRESSION:
</report_text>
</report>
```

Centrándose en el campo *'report_text'*, que es el que se utilizará durante este proyecto, se puede observar que, aunque no se percibe un formato claro, sí que existen ciertas características explotables a la hora de realizar el procesamiento (filtrado y formateo) del texto.

Primero de todo, todos los archivos de texto van encabezados por un identificador del informe, entre corchetes, que no posee ningún tipo de información valiosa para realizar un Procesamiento de Lenguaje Natural posterior.

Además, se aprecia que en el texto se va dividiendo el informe en bloques de información utilizando palabras clave en mayúsculas, lo cual se aprovechará para estructurar cada uno de los informes en una estructura de apartados para el análisis posterior.

También se puede observar que estos informes están anonimizados, representando los campos ofuscados mediante " **TIPO_CAMPO[Valor_Aproximado]". Dado que estos campos, aunque no tengan el valor exacto, sí que pueden tener información de valor (por ejemplo, en el caso de un campo de edad, no se indica la edad exacta, pero sí la década de la misma).

Por otro lado, aunque en este ejemplo no se puede apreciar en el primer ejemplo expuesto, existen veces en las que el archivo comienza mostrando parámetros del informe pero que no tienen valor para el procesamiento, como se muestra a continuación

```
**INSTITUTION
EMERGENCY DEPARTMENT
PATIENT NAME:  **NAME[AAA, BBB M]
ACCOUNT #:  **ID-NUM
DATE OF SERVICE:  **DATE[Jun 18 07]
PRIMARY CARE PHYSICIAN: K  U
```

En estas líneas de texto no existe tampoco valores que se desean trasladar al resto del procesamiento, por lo que también las descartaremos.

Otros campos de este tipo también se encuentran al final de estos archivos:

```
_____________________________________
**NAME[CCC ZZZ], M.D.

Dictator:  **NAME[WWW M. XXX], M.D.
**INITIALS
D:  **DATE[Jun 18 2007] 07:02:01
T:  **DATE[Jun 18 2007] 07:55:15
R:  **DATE[Jun 18 2007] 07:55:15/cme
Job ID:  350875/**ID-NUM
Cc:

*** Dictated By: -**NAME[XXX, VVV M]. ***
Electronically Signed by **NAME[YYY ZZZ]  **DATE[Jun 20 2007] 04:16:05 PM
```

Pero en este caso las líneas de texto que se desean descartar se encuentran bajo la secuencia de caracteres

```
_____________________________________
```

Por lo que se explotará esta característica del texto para dividir los datos.

5.1.1 LECTURA DE LOS .XML

El primer paso será la lectura de los archivos .xml que se encuentran en el directorio. Utilizando la librería '*DOM'* se lee cada uno de los archivos capturando el campo del texto que será procesarlo a continuación.

A partir de aquí se trabaja con una variable para cada uno de los informes, que se procesará más adelante para acabar con una estructura de generada a partir de ésta.

5.1.2 FILTRADO DEL TEXTO

Ahora lo que se debe de realizar es, utilizando todo el conocimiento expuesto en la introducción de este apartado, eliminar las estructuras de texto no deseadas para que el

formateo posterior del texto sea lo más sencillo posible, y se realice utilizando texto útil y no caracteres indeseados.

Primero de todo, buscando regularizar el texto lo máximo posible, se utilizará un diccionario de contracciones del inglés para deshacerlas, de forma que los analizadores sintácticos y léxicos posteriores tengan menos problemas a la hora de identificar estos términos.

```
contraction_mapping = {"ain't": "is not", "aren't": "are not","can't":
"cannot", "'cause": "because", "could've": "could have", "couldn't":
"could not", ...}
```

El segundo paso es el de descartar aquellas líneas que contienen metadatos de la generación de los archivos (como se ha mostrado anteriormente), para lo cual se capturan aquellas cadenas de caracteres que nos delimiten que un apartado del texto no es necesario para el análisis posterior.

Aparte de utilizar el divisor por '_' visto antes, también se utilizará un archivo de *keywords* para que, si una línea comienza por estas palabras, sea descartada del texto.

Tras esto, para este filtrado se utiliza principalmente Expresiones Regulares, también conocidas como *RegEx* – del inglés *Regular Expressions* -, apoyando también este procesamiento con el uso de sustituciones de literales (*.replace()* en Python).

Los casos concretos para los que se ha utilizado RegEx en este proyecto son los siguientes:

El primer paso es la desencapsulación de la información personal ofuscada para quedarse únicamente con valores numéricos útiles.

RegEx: `\*\*[A-Z]+\[.*?\]`

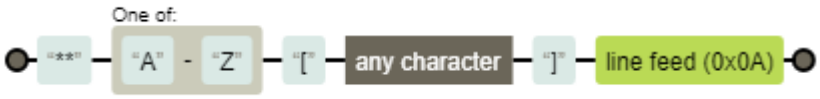


Figura 17. RegEx para capturar datos personales ofuscados

Tras eso, se eliminan aquellos rastros de datos personales que han sido omitidos al generar el informe (pero no sustituidos).

RegEx: `\*\*[A-Z\-]+`

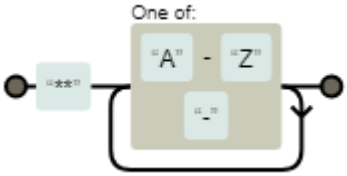


Figura 18. RegEx para capturar datos personales omitidos

El siguiente paso será capturar los datos entre paréntesis para extraerlos de ahí, dado que el tratamiento de Lenguaje Natural posterior no considera los paréntesis, pero puede llegar a provocar fallos en éste.

RegEx: `\([a-zA-Z]+\)`

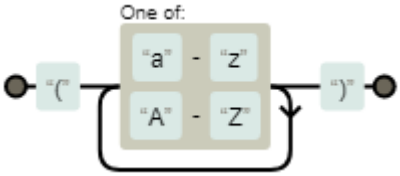


Figura 19. RegEx para extraer los caracteres entre paréntesis

En el caso de la información entre los delimitadores "[ ]" y "< >" se puede descartar totalmente, dado que en ninguna situación tienen información de utilidad (son metadatos incrustados dentro del campo de texto).

```
RegEx: \[.*?\]
       \<.*?\>
```

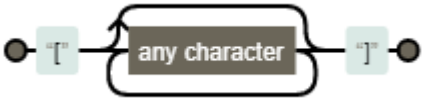


Figura 20. Regex para eliminar los caracteres entre "[ ]"

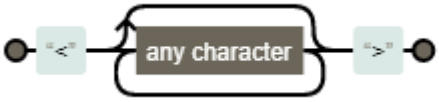


Figura 21. Regex para eliminar los caracteres entre "< >"

Por otro lado, tras haber desencapsulado la información especial y haber identificado estructuras que no son de interés, se pueden eliminar totalmente todos aquellos caracteres sin utilidad dentro de un texto que represente el Lenguaje Natural.

```
RegEx: [^a-zA-Z0-9\.\,\:\-\/\n ]
```

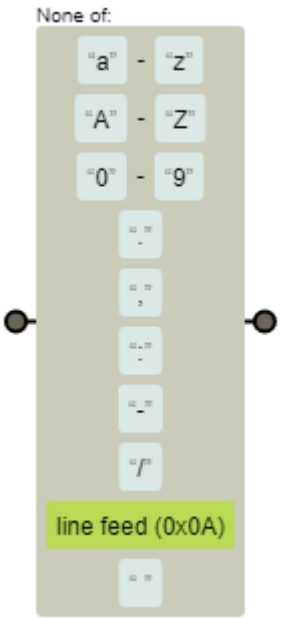


Figura 22. Regex para eliminar caracteres no deseados

El siguiente paso es eliminar las estructuras propias que identifican listas (pero no los elementos listados en éstas). Estas estructuras son las siguientes:

1: a: A: I:

1. a. A. I.

1) a) A) I)

Para este filtrado utilizaremos la siguiente expresión:

RegEx: `?\n[a-zA-Z0-9]{1,2}[\.\:\)]`

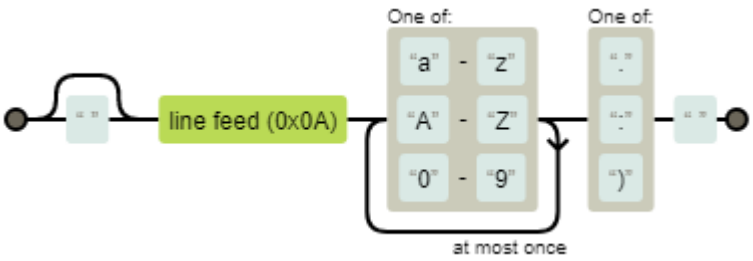


Figura 23. RegEx para eliminar los identificadores de listas

El último paso es el de eliminar espacios y los saltos de línea múltiples generados tanto a la hora de unir las visitas de los pacientes en el informe como los creados artificialmente, manteniendo únicamente una repetición de cada una de estas cadenas consecutivas.

RegEx: `' +'`
`'\n+`

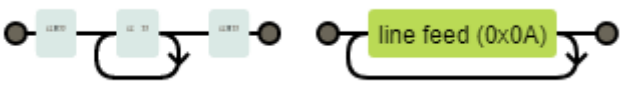


Figura 24. Regex para eliminar espacios y saltos de línea consecutivos

Este proceso queda ejemplificado en los siguientes bloques, definidos por el patrón de colores mostrado.

Versión antes del filtrado:

```
DATE OF SERVICE:  **DATE[Feb 02 06]
HISTORY OF PRESENT ILLNESS:
The patient is a **AGE[in 20s]-year-old gentleman who was  sticking a pin
in his mouth when his roommate accidently hit it...
IMPRESSION:
1. UNREMARKABLE CT ANGIOGRAM OF THE HEAD.
2. UNREMARKABLE CT ANGIOGRAM OF THE NECK.
EMERGENCY DEPARTMENT COURSE:  ... He is not having any pain at this time
and is resting comfortably.  I spoke with Dr. [_____]*** and
discussed these findings with him including the ...
(\n)
My signature below is attestation that I have interpreted this/these
examination(s) and agree with the findings as noted above.
```

Versión después del filtrado:

```
HISTORY OF PRESENT ILLNESS:
The patient is a 20-year-old gentleman who was sticking a pin in his
mouth when his roommate accidently hit it...
IMPRESSION:
UNREMARKABLE CT ANGIOGRAM OF THE HEAD.
UNREMARKABLE CT ANGIOGRAM OF THE NECK.
EMERGENCY DEPARTMENT COURSE:
... He is  not having any pain at this time and is resting comfortably.
I spoke with  Dr. and discussed these findings with him including the ...
My signature below is attestation that I have interpreted this/these
examination and agree with the findings as noted above.
```

Habiendo procesado ya el texto ahora mismo se sigue teniendo una única cadena de caracteres para cada uno de los informes, pero de tamaño resumido y con una estructura más clara, por lo que ya se puede comenzar la parte del formateo del texto.

5.1.3 FORMATEO DEL TEXTO

En este paso del procesamiento del texto se cuenta con una variable que contiene toda la información filtrada de un informe. El siguiente paso a realizar será el de, mediante la identificación de cadenas de caracteres concretas, dividir el texto en sus distintas partes

para poder realizar un análisis posterior más preciso y dirigido únicamente a partes concretas de un informe.

Para llevar a cabo esta parte lo que se ha realizado es, primero de todo, crear unos "archivos de configuración" (en formato .txt) en los que se almacenarán todas las cadenas de caracteres que representen títulos de un apartado en concreto de cada uno de los informes. La ventaja de utilizar estos archivos es que, si en la ejecución de la herramienta creada un documento falla, se pueda capturar fácilmente qué frase o frases representan un apartado, y añadirlos al final del fichero para ajustar su forma de funcionamiento.

La siguiente representación constituye el archivo de configuración para el apartado de laboratorio:

```
lab.txt

LABORATORY DATA
LABORATORIES/DIAGNOSTICS
LABORATORIESDIAGNOSTICS
LAB WORK
```

En la parte de procesamiento, para cada uno de estos archivos, se sustituirá cada una de las líneas contenidas dentro del archivo por unas etiquetas encabezadas por el nombre del archivo en el que se encontró la cadena (sin la extensión .txt) que delimitarán cada uno de los apartados del texto, para que posteriormente sea más sencillo dividirlos y conocer a qué apartado corresponde cada uno de los bloques.

Siguiendo con el ejemplo anterior, las etiquetas generadas e intercaladas en formato texto serían:

```
[LAB]
Text...
...
[END]
```

Teniendo ya el texto dividido en bloques, pero aun dentro de una única variable, se procede a dividir el texto según las etiquetas para tener distintos bloques. Aquellos que provengan del mismo archivo se unirán dentro de una misma variable, y con toda la información recogida y separada se crea una instancia del objeto *Report*, que almacenará toda la información generada y formateada procedente del archivo original. La estructura de este archivo es la mostrada en la Figura 25.

REPORT
ID del report
Alergias
Síntomas de entrada
Diagnóstico
Evolución en el hospital
Historial clínico
Análisis de laboratorio
Medicamentos
Operaciones realizadas
Signos del paciente
Recomendaciones de los médicos

Figura 25. Clase Report

Además, la información almacenada no se va a guardar en formato *String*, sino que se utilizará otra clase (*Block*) que almacenará por un lado el texto original y por otro una clase *Words* con información adicional (explicada con profundidad en el apartado *5.2 MetaMap*).

```
class Block:
    def __init__(self, text, words = None):
        self.text = text
        self.words = words
```

Figura 26. Clase Block

```
class Words:
    def __init__(self, raw, cui, explanation, tag):
        self.raw = raw
        self.cui = cui
        self.explanation = explanation
        self.tag = tag
```

Figura 27. Clase Words

Ya con la información filtrada y formateada según las necesidades de los bloques posteriores, se procede a crear la integración con la aplicación de MetaMap, para llevar a cabo de esta forma el proceso de *Named-Entity Recognition*.

5.2 METAMAP

La segunda parte de este proyecto queda constituida por la integración de la aplicación MetaMap dentro del procesamiento del texto con el objetivo principal de integrar todo el conocimiento experto aportado por el metatesauro UMLS.

Los resultados devueltos por MetaMap se utilizará para completar la información de la clase Word, definida en la Figura 27, de forma que para cada palabra o grupos de palabras se posea información sobre el identificador de la misma dentro del metatesauro, una explicación generada por este, y las etiquetas según los grupos semánticos definidos dentro de la aplicación.

El objetivo principal de MetaMap es el mapeo automático de los conceptos del Metatesauro UMLS referenciados en el texto, prerrequisito primordial para ciertas aplicaciones relacionadas con la recuperación de información, *Text Mining*, categorización, resúmenes y otras tareas del procesamiento del lenguaje natural. Para llevar esto a cabo, MetaMap utiliza aproximaciones basadas en NLP y técnicas lingüísticas computacionales.

El Reconocimiento de Entidades Nombradas – NER, des sus iniciales en inglés-, es un trabajo de extracción de información del texto localizándolas y clasificándolas según unas categorías previamente definidas.

El problema principal al que se enfrentan las tecnologías de NER es la aparición de palabras con varias etiquetas posibles. Para solucionar este problema se llevan a cabo dos técnicas principales:

Primero de todo, se han de identificar las características sintácticas de una frase, para conocer cuál es el significado de una palabra en una oración, y así identificarla correctamente (o disminuir el número de etiquetas posibles).

Por otro lado, se ha de considerar que hay veces en las que una palabra no tiene significado por sí mismo, dado que se puede estar tratando con un nombre compuesto. De esta forma, sería considerable que el etiquetador haga un procesamiento también de *n-grams* a la hora de llevar a cabo el reconocimiento de entidades.

Aun considerando estas técnicas, siguen existiendo casos en los que, para una frase, más de una posible combinación de NERs es factible, por lo que ya diferirá el funcionamiento de cada una de las herramientas existentes el cómo tratarán esta información.

5.2.1 ESTRATEGIAS DE MAPEO EN METAMAP

Para cada expresión de entrada al sistema, la aplicación sigue los siguientes pasos:

1. Analizar el texto en sintagmas o frases nominales, llevando a cabo los siguientes pasos para cada una de las divisiones generadas.

2. Generar variaciones para cada frase nominal (consistentes en una o más frases con variaciones, abreviaciones, acrónimos, sinónimos …).

Para generar estas variaciones se utiliza un algoritmo basado en el conocimiento, extraído de:

– El lexicón SPECIALIST y la tabla de formas canonícas derivadas.

– La base de conocimiento SPECIALIST de acrónimos y abreviaturas.

– La base de conocimiento SPECIALIST de reglas de morfología derivacional y de sinónimos.

3. Formar el conjunto de candidatos, conteniendo cada una de las variaciones.

4. Para cada candidato calcular su fuerza mediante funciones.

5. Seleccionar y combinar aquellos candidatos que mejor se ajusten al mapeo de la frase original, generando de esta forma los llamados **Meta Mappings**

Los **Meta Mappings** serán los elementos finales que definen las diferentes variaciones de conceptos asociadas a la cadena de entrada [12].

5.2.2 ARQUITECTURA DE METAMAP

La arquitectura de la aplicación MetaMap consta de diversos bloques. Primero se ha de llevar a cabo un análisis léxico y sintáctico de cada una de las entradas a la misma, para que tras esto la aplicación pueda generar una lista de candidatos y que, sobre los mismos, cruce la información con el metatesauro UMLS y que de esta forma añada todo el conocimiento específico de los campos de la salud y la biotecnología al procesamiento de los informes médicos que se lleva a cabo.

La arquitectura de MetaMap queda explicada en la siguiente Figura:

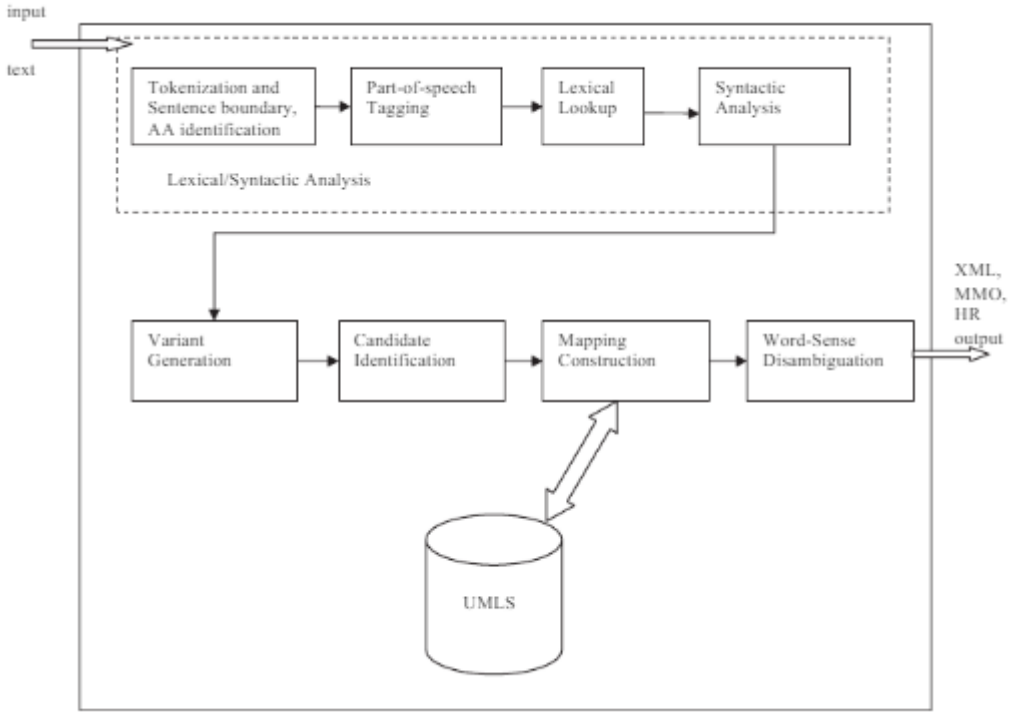


Figura 28. Arquitectura de MetaMap

5.2.3 TOKENIZACIÓN EN EL PROCESAMIENTO DEL TEXTO

Un proceso de gran importancia dentro del Procesamiento de Lenguaje Natural es la parte de la *tokenización* del texto.

El proceso de *tokenización* del texto es la forma de separar cada pieza de texto en unidades de menor tamaño llamadas *tokens*, que pueden ser palabras, caracteres o *substrings*. El objetivo de este proceso es el de llevar acabo un modelado de los datos de texto, de forma que representen de la forma más fidedigna posible el texto del que provienen para el análisis posterior, adaptándolo al caso de uso concreto.

Es importante destacar que en este proyecto se hicieron dos pruebas de *tokenización* distintas. Primero de todo se optó por llevar a cabo un proceso de *lemmatization* (esto es,

llevar cada una de las palabras a su raíz mediante un análisis sintáctico de las oraciones) independiente a la aplicación de MetaMap.

Este proceso, tras la limpieza de los datos, extrae cada frase del documento y la analiza sintácticamente para conocer cuál es la familia léxica de cada uno de los términos, y de esta forma saber la raíz de la que proceden.

Por otro lado, también existe el proceso de *stemming*, el cual en vez de llevar la palabra a la raíz la cambia por su forma base.

La ventaja del primer método sobre el segundo es que, a costa de una mayor complejidad computacional, el proceso se asegura de no confundirse entre palabras homónimas (que, aunque se escriben igual, tienen una distinta etimología), y además la palabra base por la que se sustituye siempre está recogida dentro de los diccionarios.

Por ejemplo:

Studies --- Stemming ---> studi

Studies --- Lemmatization ---> study

Además, dentro del proceso de *lemmatization*, el proceso de análisis sintáctico llevado a cabo puede utilizarse para filtrar ciertas palabras y descartar aquellas que no proporcionan significado a la frase (como, por ejemplo, los determinantes o las preposiciones, llamadas comúnmente *stopwords*).

Por otro lado, también se llevó a cabo un procesamiento del texto omitiendo la parte de *tokenización* del texto, para simplemente integrar la salida del filtrado y formateo del texto hacia la aplicación de MetaMap.

Este segundo método generó mejores resultados que el primero dado que su eficiente integración dentro de la aplicación beneficia a la división del texto de entrada en los sub-grupos definidos en el apartado 5.2.1 *Estrategias de mapeo en MetaMap*,

independientemente de que a la aplicación también entraran palabras sin valor ni significado dentro del dominio de trabajo (como las *stopwords* comentadas anteriormente).

5.2.4 SALIDA DE LA APLICACIÓN DE METAMAP

Tras la limpieza del texto por parte de la aplicación, se obtiene una salida en formato de líneas de texto sin terminación, esto es, un fichero en el que cada *Meta Mapping* de los distintos sintagmas queda definido en una línea del fichero, tal como se muestra al final de este apartado.

Por otro lado, todas las posibles salidas del programa son:

- **ID** – Identificador único para identificar el texto procesado
- **MMI**
- **Score** – *MetaMap Indexing* (MMI), valoración con un máximo sobre 1000,00. Cuanto mayor sea, significa que el concepto UMLS es de mayor importancia según los algoritmos MMI. El resultado MMI se presentan de mayor a menor relevancia.
- **UMLS Concept Preferred Name** – *UMLS preferred name* para cada uno de los conceptos UMLS identificados en el texto.
- **UMLS Concept Unique Identifier (CUI)** – CUI para los conceptos UMLS identificados.
- **Semantic Type List** – Lista de abreviaciones semánticas (separadas por comas) de los conceptos UMLS identificados.
- **Trigger Information** – Tupla de 6 elementos separados por comas que muestran los disparadores MMI para identificar el concepto UMLS.
    - UMLS Concept (Preferred or Synonym Text)
    - loc – Posición en el texto (si es identificable).
        - ti – Título
        - ab – Abstract
        - tx – Texto Libre
    - locPos – Número del enunciado dentro de la ubicación
    - text – Texto mapeado con el concepto UMLS identificado.

  - Part of Speech – Determinado por el *MedPost Tagger* o el *Lexicon Lookup*
  - Negation Flag – 1 si el texto se considera una negación para MetaMap (y 0 en el caso contrario).
- **Positional Information** – Información posicional separada por ';'. Se define el punto de comienzo del texto y la longitud dentro de la cadena de entrada.
- **Treecode(s)** – Lista separada por punto y coma de cualquier código de árbol MeSH asociado con el concepto UMLS. El campo puede ser nulo si no se encontraron códigos de árbol para el concepto.

En este proyecto, dado la limitación del conocimiento experto y los requerimientos necesarios para la parte de análisis posterior, solamente se han utilizado los campos proporcionados por la salida anteriormente mencionada, que son:

- ***Concept Score***
- ***CUI***
- ***UMLS string matched***
- ***Concept's Preferred Name***
- ***Concept's Semantic Type(s)***

Siendo un ejemplo de la salida del programa con esta configuración el mostrado a continuación:

```
Phrase: right sided subclavian line in place again.
Meta Mapping (719):
612   C0205090:right sided (Right) [Spatial Concept]
581   C0589488:Subclavian (Subclavicular approach) [Spatial Concept]
748   C0205132:Line (Linear) [Spatial Concept]
581   C0442504:Place [Spatial Concept]
Meta Mapping (719):
612   C0205090:right sided (Right) [Spatial Concept]
581   C0589488:Subclavian (Subclavicular approach) [Spatial Concept]
748   C0205132:Line (Linear) [Spatial Concept]
581   C1533810:Place (Placement action) [Health Care Activity]
...
```

De esta salida cabe destacar la utilidad que resulta el campo del *Semantic Type*, utilizado para realizar el *Named Entity Recognition (NER)*, utilizando así el conocimiento adquirido del dominio del caso.

Tras todo este procesamiento, aplicando el conocimiento adquirido gracias a MetaMap y el metatesauro UMLS, se puede proceder a las fases de análisis de la información recabada para generar modelos de recomendación para los expertos dentro de este dominio de trabajo.

En vez de utilizar las técnicas más conocidas de *tokenización* dentro del NLP (realizando *POS-Tagging* para realizar un análisis sintáctico y tras esto *lemmatization*, o por otro lado realizar el proceso de *stemming* con cada una de las palabras del documento) en las siguientes partes de este proyecto se ha procedido a utilizar las estructuras resultantes de *MetaMap* de división del texto, en forma de una palabra o un grupo de palabras, además del conocimiento extra obtenido gracias al *metatesauro* UMLS para elegir las palabras más representativas de un documento a la hora de realizar un procesamiento en concreto, dado que son estos términos los que mejor representan los conceptos y el conocimiento recogido en cada uno de los textos.

5.3 CLUSTERING

El primero de los módulos de apoyo creados para este proyecto es la elaboración de un sistema de clasificación de documentos según el contenido almacenado en cada uno de los informes, con el objetivo de agrupar los documentos según los contenidos de cada uno de los apartados mencionados anteriormente y definidos en la Figura 25.

Esta herramienta servirá de apoyo a los profesionales agilizando la búsqueda en el histórico de informes, descartando automáticamente aquellos sin importancia para un caso de uso determinado, y posibilitando el trabajo posterior con grupo reducido de los datos.

El primero de los procesamientos llevado a cabo dentro de este módulo es el de la representación matemática de las estructuras de texto que componen los informes médicos; este concepto se denomina *Word Embeddings*.

Los *Word Embeddings* son la forma de representar de palabras por una máquina buscando captar la comprensión humana del lenguaje. Estas representaciones de texto se proyectan sobre un espacio de n dimensiones de forma que las palabras con un significado similar se encuentran cercanos entre ellos dentro de este espacio. Esta característica es esencial para resolver la mayoría de los problemas de procesamiento del lenguaje natural relativos a este concepto.

Como el objetivo en este proyecto es buscar la similitud entre documentos, se utiliza una extensión de este concepto denominada *Doc Embeddings* los que, aprovechando técnicas de *Word Embeddings*, buscan definir todo un documento dentro del espacio vectorial.

Existen diversas formas de representar vectorialmente los textos: Cálculos estadísticos, utilización de modelos predictivos o basados en casos de uso concretos (ya sean redes neuronales, grafos…) …

Además, en la configuración de hiperparámetros de los métodos de *Word/Doc Embeddings* a utilizar, un parámetro clave es el de la **dimensión de los vectores**. En este proyecto se ha decidido utilizar una **dimensión de vectores de 200** en todos los métodos desarrollados dado que los tamaños de los documentos a tratar son mayores al tamaño típico de documentos sobre los que se realizan *Embeddings*.

En este proyecto se han utilizado dos métodos de creación de *Doc Embeddings* distintos:

5.3.1 TF-IDF

El primero de los métodos de *Embeddings* desarrollado en este proyecto es TF-IDF, el cual se basa en computar la relevancia de una palabra según una medida estadística de

frecuencia de esta en un documento, comparándola con su aparición en toda una colección de documentos. La fórmula utilizada es la multiplicación de dos métricas: cuántas veces aparece una palabra en un documento y la frecuencia inversa del documento de la palabra en un conjunto de documentos. El método TF-IDF queda definido en la siguiente ecuación.

$$w_{i,j} = tf_{i,j} \times \log\left({}^{N}/_{df_i}\right)$$

W = Peso de *i* en *j*

tf = Número de ocurrencias de *i* en *j*

df = Número de documentos que contienen *i*

N = Número total de documentos

Su forma de funcionamiento es sencilla, la aparición de una palabra dentro de un único documento hace que el peso suba, pero si esta misma palabra aparece en el resto de los documentos, baja. De esta forma, podemos computar qué elementos son característicos y diferenciadores para un documento en concreto, representando así un vector de características que recoja la información más importante de cada texto.

La generación del *Doc Embeddings* utilizando lo explicado anteriormente se realiza mediante la ponderación de los pesos de cada uno de los términos contenidos en el documento, generando de esta forma diferentes vectores según la combinación de términos del informe.

5.3.2 DOC2VEC

Por otro lado, Doc2Vec es un modelo de predicción que se basa en aprender cómo proyectar un documento (matemáticamente) dentro de un espacio vectorial latente de n dimensiones. Existen dos métodos fundamentales de funcionamiento: el basado en *skip-grams* y en *Bag of Words*, utilizando en este proyecto el segundo método.

Doc2Vec está basado en el modelo *Word2Vec* [27], método de *Word Embeddings* que se utiliza para la representación vectorial de términos de documentos. La forma de

funcionamiento de este modelo se basa en aprender a predecir el valor de una palabra basándose en su contexto.

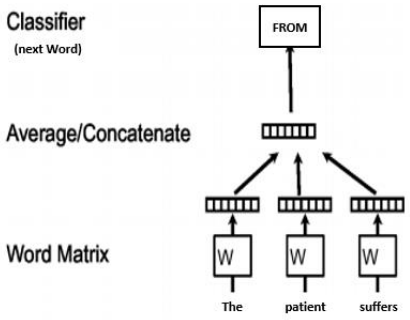


Figura 29. Ejemplo de procesamiento de Word2Vec

Tanto *Word2Vec* como *Doc2Vec* son modelos que se benefician cuando hay que tratar con una gran cantidad de datos sin etiquetar, característica que en este proyecto es muy apreciable dado que en todos los sistemas desarrollados se utilizan técnicas no-supervisadas, debido a la naturaleza de los datos.

Para la generación de los *Doc Embeddings*, en *Doc2Vec* se añade una entrada más al modelo, el *Document ID*, y el objetivo del modelo será el de, además de predecir las representaciones de cada palabra, también predecir el valor de este identificador, que corresponderá a la representación utilizada para un documento.

5.3.3 CLUSTERING CON K-MEANS

Tras obtener las representaciones vectoriales de los documentos, el siguiente paso realizado es la agrupación de los mismos utilizando K-Means.

De forma resumida, K-Means asegura una agrupación en la que la distancia entre los elementos de un mismo cluster es mínima maximizando a su vez la distancia entre documentos de distintos clusters.

Para determinar el número de clusters para la agrupación, se ha integrado un sistema de cálculo automático de éste utilizando el *Silhouette Coefficient* de los grupos, esto es, una

medida de cuán similar es un objeto a su propio grupo (cohesión) en comparación con otros grupos (separación).

En la presentación de resultados mostrada a continuación se han utilizado imágenes correspondientes a una representación de los vectores en dos dimensiones sobre las dos componentes principales. Debido a esto se ha de considerar que la representación es orientativa de la distribución de los informes, pero a su vez no es totalmente exacta (y se aprecian solapamiento en los grupos) debido a la falta de representación del resto de dimensiones.

A continuación, se muestran las representaciones utilizando los dos métodos definidos anteriormente para clasificar los informes según los Síntomas de entrada de los pacientes.

En el caso de TF-IDF, obtenemos la representación mostrada en la Figura 30,

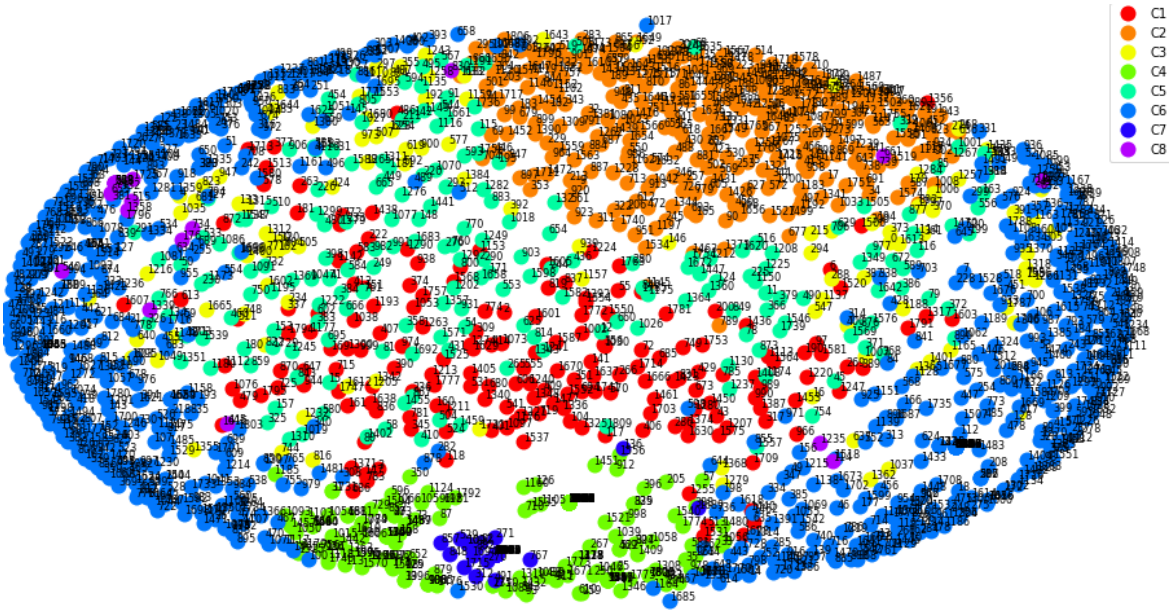


Figura 30. Representación del Clustering utilizando TF-IDF

Por otro lado, los resultados obtenidos utilizando el método de *Doc2Vec*, para la misma configuración de K-Means, son:

Figura 31. Representación del Clustering utilizando Doc2Vec

En este caso se aprecia que el codo se encuentra para únicamente 2 clusters.

5.3.4 TOPICS DE CADA CLUSTER

Un añadido en este módulo es la integración de un sistema de obtención de los topics de cada cluster utilizando técnicas de LDA (*Latent Dirichlet Allocation*), y definiendo así de forma automática un tema orientativo para cada grupo.

Aun así se reitera que la funcionalidad principal de esta parte del proyecto es la de búsqueda dentro de un cluster según un informe de interés y el análisis posterior del resto de los informes.

5.3.5 COMPARATIVA DE LOS RESULTADOS

Por un lado, para la salida obtenida utilizando TF-IDF se obtiene, para el cluster #6, el siguiente resultado:

```
Topic:["pain" "Abdominal" "nausea" "vomiting" "back")]
```

Siendo el contenido dentro de varios de los informes:

```
Report4.xml:
Abdominal pain. Abdominal pain and change in mental status.
Report1111.xml:
Abdominal pain.
Report15340.xml:
Abdominal pain and vomiting.
```

Mientras que en el caso de Doc2Vec, observando el primero de los clusters:

```
Topic: "pain" "normal" "creatinine" "Chest" "count"
```

Se ve que los conceptos resumidos en el topic son menos clarificadores. Por otro lado, analizando los documentos de este cluster.

```
Report1.xml:
Tooth pain.
Report108.xml:
I am coping with depression. BUN 25, creatinine 1. 5. Chest x-ray from
the 29th of showed an enlarged heart. No pulmonaryedema. I personally
reviewed the examinations and agree with thatinterpretation.
Report10282.xml:
Right hearing loss times three days.
```

Se aprecia que en este caso no existe ninguna relación entre los Síntomas de entrada de los pacientes, lo que invita a descartar el método de Doc2Vec para un uso como el de este proyecto.

5.4 GENERACIÓN DE RESÚMENES AUTOMÁTICOS

El concepto de Resúmenes Automáticos de Texto dentro del *Machine Learning* recoge el conjunto de técnicas empleadas por una máquina para reducir un texto o un conjunto de documentos en breves párrafos o líneas que condensen la idea principal expresada en éste utilizando métodos matemáticos.

Los modelos creados para este trabajo están programados con el objetivo de primero entender el contenido de los documentos, para crear más adelante la salida requerida.

El objetivo buscado en este módulo del proyecto es la creación de una herramienta de apoyo capaz de condensar la información más relevante de los apartados de cada uno de los informes, ahorrando así tiempo a los profesionales a la hora de tener que buscar información dentro del histórico de los datos.

Para llevar a cabo este proceso existen dos aproximaciones principales llevadas a cabo en el área del NLP, ambas probadas durante el desarrollo de este proyecto, y que quedan definidas a continuación:

5.4.1 EXTRACTIVE TEXT SUMMARIZATION

La primera de las aproximaciones utilizadas en este proyecto se basa en identificar los segmentos más importantes de un informe utilizando un sistema de rankings de cada una de las oraciones en las que se ha dividido los textos (esta división es configurable para utilizar una segmentación óptima para cada caso de uso). El resultado obtenido utilizando esta técnica es un resumen generado utilizando las frases más importantes identificadas dentro del texto, por lo que el modelo no ha de ser capaz de crear de cero una salida en lenguaje natural.

Los distintos modelos probados han sido:

- **TextRank** [28]. Este algoritmo se basa en 2 pasos, que son:
  - Identificar oraciones relevantes: Construcción de un grafo donde los vértices representan cada oración en un documento y los cruces entre las oraciones se basan en la superposición de contenido.
  - Identificar palabras clave relevantes: Construcción de una red de palabras según términos consecutivos. Se establece un enlace entre dos palabras si se suceden, el enlace adquiere un peso mayor si estas 2 palabras aparecen con mayor frecuencia una al lado de la otra en el texto.

- **LexRank** [29]. Una oración que es similar a muchas otras oraciones del texto tiene una alta probabilidad de ser importante.
- **LSA** (Latent semantic analysis). Análisis de relaciones entre un conjunto de documentos y los términos que contienen mediante la producción de un conjunto de conceptos relacionados con los documentos y términos, asumiendo que las palabras con un significado cercano aparecerán en fragmentos de texto similares.
- **Luhn** [30]. Enfoque basado en TF-IDF. Es útil cuando las palabras muy poco frecuentes y las palabras muy frecuentes (stopwords…) no son significativas.
- **KL.Sum** [31]. Selecciona oraciones basándose en la similitud de distribución de palabras con el texto original. Su objetivo es reducir los criterios de divergencia KL. Utiliza un enfoque de optimización greedy y sigue agregando oraciones hasta que disminuye la divergencia KL.
- **BERTSUM** [32]. Adaptación del modelo BERT para utilizarse en procesos de generación de resúmenes automáticos. El modelo BERT se utiliza dentro de este contexto como forma de identificar dentro de un documento el peso que tiene cada una de las frases. Este peso se computa en base al contexto del texto (tanto anterior como posterior, característica fundamental en BERT). La salida de este modelo son valoraciones para cada una de las frases del documento según el valor que aporten dentro del mismo, escogiendo las n frases con mayor valoración.

5.4.2 ABSTRACTIVE TEXT SUMMARIZATION

La otra de las aproximaciones utilizadas en esta parte del proyecto se basa en la interpretación del contenido del texto de entrada por un modelo *seq-to-seq* y la creación de un nuevo texto en lenguaje natural más corto pero que resuma el contenido principal del mismo.

Este enfoque implica la creación de un resumen basado en *Deep Learning*, utilizando técnicas de NLP y *Machine Learning* más complejas que la aproximación de *Extractive Text*

Summarization, generando nuevas frases y términos diferentes a las del documento real, suponiendo una técnica mucho más compleja que el otro enfoque.

En esta técnica se utilizan modelos seq-to-seq de cadenas de texto (tanto la entrada como la salida del modelo es texto) que se entrenan para el caso de uso específico de la elaboración de resúmenes.

Los modelos probados correspondientes a esta técnica han sido:

- **T5** [33]. Transformer elaborado por Google basado en un modelo codificador-decodificador que enfoca todos los casos de uso a un caso text-to-text y entrenado utilizando un gran corpus de textos.
- **BART** [34]. Modelo seq-2-seq basado en un codificador bidireccional (como BERT) y un decodificador left-to-right (como GPT). Su entrenamiento se lleva a cabo ordenando de forma aleatoria todas las frases donde partes del texto se encapsulan tras tokens de enmascaramiento. Modelo eficiente en la generación de texto.
- **GPT-2** (Generative Pre-trained Transformer 2) [35]. Modelo entrenado con el objetivo de predecir la siguiente palabra de una oración, dadas todas las palabras anteriores dentro de un texto. La diversidad del conjunto de datos hace que este simple objetivo contenga demostraciones naturales de muchas tareas en diversos dominios.
- **XLM** [36]. Modelo especializado en el aprendizaje de relaciones entre términos de distintos idiomas y en procesos de clasificación de términos multi-idioma.

5.4.3 MODELO DEFINITIVO

Tras todas las pruebas realizadas se ha concluido que la mejor de las técnicas según los resultados obtenidos es la de *Extractive Text Summarization*, debido principalmente a la no dependencia de un modelo entrenado como fuente principal de generación de los resúmenes. Además, los modelos utilizados ofrecen unos resultados más óptimos cuando

están entrenados con textos del mismo dominio de trabajo, además de utilizar etiquetas para poder validar el buen entrenamiento del mismo.

El área de trabajo tan concreta y especializada en la que se elabora este proyecto hace que, sin un entrenamiento específico de modelos de gran complejidad con una fuente de datos propia de este ámbito, la aproximación de *Abstractive Text Summarization* no aporte valor al proyecto. Es más, las oraciones obtenidas como resultado de estas técnicas apenas diferían de las frases del texto original, debido a que los modelos no podían generar frases originales de lenguaje natural.

Concretizando más, dentro de las distintas técnicas de *Extractive Text Summarization* se ha decidido escoger el método de *BERTSUM*, en el que se integra modelos de *Machine Learning* (*BERT*) dentro de esta técnica debido a que el funcionamiento del modelo dentro de la estructura de esta técnica de generación de resúmenes no tiene las limitaciones por las cuales se descartan las anteriores, dado que el modelo se utiliza para ponderar las frases y no para crearlas desde cero.

Un ejemplo de resultado es el siguiente:

Para el caso de tener un texto de entrada como el siguiente

```
Additionally, the patient denies having any fevers, chills, headache,
neck pain, abdominal pain, diarrhea, or dysuria. Remaining review of
systems is otherwise negative

 This is a well-appearing male sitting in bed in no acute distress

 Pupils are equal and round bilaterally. Extraocular movements are
intact. Oropharynx is without erythema or exudate. No lymphadenopathy or
JVD. Regular rate and rhythm. Clear to auscultation bilaterally without
wheezing, rales, or rhonchi. Soft, nontender, and nondistended without
guarding or rebound. Without clubbing, cyanosis, or edema. Alert and
oriented x3. No focal deficits are noted
 The patient remained stable in the emergency department. He was given an
inch of nitro paste and 4 baby aspirin. He is not having any pain at this
time and is resting comfortably

 I spoke withand discussed these findings with him including the CT scan
of the chest. I also spoke with Dr the attending radiologist who read the
```

```
CT scan. He states that this type of volvulus does not require urgent
surgical consultation especially if the patient has no findings of
abdominal pain or vomiting. However, he noted that the volvulus may need
to be corrected in the future if any problems arise

 Dr asked that we consult Dr and order a stress test for the morning
```

Se aprecian 3 temas principales dentro del texto:

- Una descripción de las condiciones del paciente
- Los estudios realizados por el personal médico
- Comentarios realizados a los distintos doctores (cuyos nombres están anonimizados y por ende no aparecen)

El resultado generado por el modelo SUMBERT es:

```
Additionally, the patient denies having any fevers, chills, headache,
neck pain, abdominal pain, diarrhea, or dysuria.

This is a well-appearing male sitting in bed in no acute distress.

The patient remained stable in the emergency department.

I spoke withand discussed these findings with him including the CT scan
of the chest.
```

En la salida anterior resultante de la ejecución del modelo se aprecia claramente que las oraciones que se han identificado como más representativas de todo el texto por esta técnica recogen todos los temas mencionados anteriormente.

Además, cabe destacar que el número de frases utilizadas se basa en un método de clustering utilizado por el modelo, donde hay tantas frases como clusters, y que además este número de grupos se calcula de forma automática calculando el codo obtenido mediante la aplicación del *Silhouette Coefficient*.

5.5 PREDICCIÓN DE CAMPOS

El último de los componentes desarrollados para este proyecto ha sido el módulo utilizado para la recomendación de campos dentro de los informes médicos a partir de todo el procesado previo realizado, influenciado por el conocimiento adquirido sobre el sector de la salud y la biomedicina en el apartado 5.*2 - MetaMap*.

Principalmente, este módulo del proyecto se basa en la búsqueda de similitudes entre apartados de los documentos con el objetivo de deducir contenidos de procesos todavía desconocidos dentro del ciclo de vida de un informe médico.

Por ejemplo, un caso de uso de esta herramienta sería la de conocer los posibles **Análisis de laboratorio** que recomendar a un médico realizar a partir del **Historial clínico** de un paciente.

Los resultados presentados más adelante en esta memoria se han obtenido tras un análisis de los informes buscando predecir los **Medicamentos** que se le podrían recomendar a un médico como aptas para un paciente a partir de los **Síntomas de entrada** presentados por el mismo.

Cabe destacar que este caso de uso es extrapolable a la predicción de otros campos dentro de todos los definidos en el apartado *5.1 – Procesamiento del Texto*.

Para este parte de predicción se ha decidido realizar un modelo basado en **filtros colaborativos**, concepto muy utilizado en la creación de sistemas de recomendación, y típico en sets de datos como por ejemplo los de valoraciones de usuarios a contenidos.

Los filtros colaborativos trabajan con una estructura de matriz (definida para este caso de uso más adelante en la *Tabla 7*) y se basan la predicción del valor de un campo a partir del resto de los valores almacenados, buscando similitudes con otros *users* (informes, en nuestro caso) de la matriz. Cuando existen dos casos que contengan valores muy similares en varios campos de la matriz, será más probable que el contenido a predecir para un informe adquiera al valor de aquellos que sean en el resto de los casos similares.

En el caso de este proyecto, los datos que compondrán la matriz serán un *one-hot encoding* en el que las medicinas recetadas por un médico se formateen con un valor de '1', mientras que el resto sean un '-1'. Para la entrada de un informe nuevo habrá que enfrentarse al problema de *Cold Start* definido a continuación.

5.5.1 COLD START

A la hora de programar un sistema como el explicado para este módulo, existe la problemática de tener que lidiar con aquellos informes que no tenga ningún valor definido dentro del atributo del que se desea una recomendación, lo que implica que la matriz del filtro colaborativo estará configurada totalmente con valores void (todos los valores a '0'), por lo que le será imposible al recomendador predecir alguna medicina.

Para resolver este problema se ha implementado un sistema para inicializar los campos según la búsqueda de similitudes entre informes médicos mediante la utilización de la **Distancia del Coseno** consiguiendo así utilizar algunos de los valores contenidos dentro del documento más parecido al de caso de uso, y permitiendo así la inicialización del recomendador.

Los *Doc Embeddings* utilizados en este módulo son los establecidos mediante el sistema TF-IDF utilizado anteriormente, dado que para esta necesidad ha demostrado un mejor funcionamiento que un sistema de generación de vectores más complejo.

Además, la técnica desarrollada en esta parte del proyecto sirve como herramienta única de recomendación mediante el cruce de varios campos, debido a que permite al usuario conocer automáticamente qué informes poseen un contenido lo más similar posible al informe que se analiza por el profesional. De esta forma exponiendo un caso en concreto, el sistema podría recomendar unos **Análisis de Laboratorio** según los **Síntomas de entrada** de un paciente.

5.5.2 FILTROS COLABORATIVOS CON KNN

La configuración de la matriz mediante el *one-hot encoding* mencionado anteriormente consiste en valores '1' para las medicinas que aparecen en un informe, '-1' las medicinas que se sabe que no son válidas, y un valor de '0' aquellos valores desconocidos que deseamos predecir.

Además, para configurar el eje vertical de la matriz (las medicinas) se utilizan los tags resultantes de MetaMap para conocer qué términos dentro del apartado **Medicamentos** son distintos nombres de éstos y no otro tipo de palabras (como podrían ser, por ejemplo, cantidades) y a partir de ahí se forma el índice vertical con todos los valores únicos obtenidos.

	Medicina1	**Medicina2**	…	**MedicinaN**
Report1	1	-1	…	-1
…	…	…	…	…
ReportM	0	1	…	0

Tabla 7. Estructura de la matriz del filtro colaborativo

De este modo, el sistema buscará predecir qué valores con '0' en la matriz son las más 'cercanas' a tener un valor de '1'.

Además, en el sistema de recomendación se utiliza el método de clasificación supervisada de K-NN (*k-nearest neighbors*) para conocer los informes más próximos del que se desee una recomendación. Esto se realiza debido a que el sistema no ha de identificar únicamente el informe más cercano al caso en cuestión, sino que debe de computar los mejores valores a partir de muchos documentos.

En este caso mostrado a continuación se buscará recomendar sobre qué medicinas podrían ser útiles para un paciente según sus molestias. El informe primero ha sido inicializado mediante el método de *Cold Start* para adquirir los primeros valores de las medicinas, y tras esto se ha realizado la clasificación mediante K-NN configurando en este

método el **número de vecinos a 10**, mientras que se ha configurado el **número de recomendaciones a 4**.

De esta forma obtenemos la siguiente salida en nuestro programa

```
The list of the Medicines listed in report1:
Fentanyl, Valium, Neurontin, Dilaudid

The list of the Recommended Medicines:
1: Ranitidine
2: Steroids
3: Atorvastatin
4: Sulbactam
```

Capítulo 6. ANÁLISIS DE RESULTADOS

Tras la definición de todo el trabajo desarrollado durante este proyecto de creación de un sistema de apoyo a los trabajadores dentro de los dominios de la salud y la biomedicina, procederemos al análisis de los resultados obtenidos.

Primero de todo, la alta dificultad encontrada durante el tratamiento de la fuente de datos utilizada hizo fundamental la planificación del trabajo utilizando una distribución del tiempo circular o agile. La necesidad de solventar problemas de limpieza y estructuración de la información durante la creación de los tres servicios creados (clustering, creador de resúmenes automáticos y recomendador) hubiera supuesto un gran problema si se hubiera utilizado una planificación en cascada.

En cuanto a la parte de pre-procesamiento de la información, aunque su finalidad es la de utilizarse para preparar la información para los tres módulos mencionados anteriormente, también sirve como una herramienta que se puede encapsular para trabajar con esta información en otros programas Python. La capacidad de acceder a la aplicación y extraer información importante del metatesauro UMLS supone una gran ventaja dentro del proyecto.

Debido al tiempo de procesamiento requerido para ejecutar la aplicación de MetaMap, todos los resultados de éste se han almacenado dentro de variables utilizando la librería pickle y que son fácilmente implementables en cualquier otro desarrollo Python, gracias a la estructura en clases desarrollada. De esta forma, permite tanto a las herramientas propias como a otros desarrollos ahorrarse un gran tiempo de computación.

Es debido a esto por lo que se considera que el parte de limpieza del texto y conexión con MetaMap se considera de gran utilidad tanto dentro del proyecto como para otros proyectos.

Entrando en los servicios desarrollados, primero cabe destacar que a la hora de crear los Doc Embeddings para el proyecto, el uso de los términos obtenidos de MetaMap (agrupados en una o varias palabras según el caso) han supuesto una mejora respecto a las pruebas realizadas generando los tokens mediante herramientas de lemmatization externas.

En cuanto al módulo de Clustering desarrollado es fácilmente apreciable la funcionalidad que aporta como ayuda para reducir drásticamente la búsqueda en los informes, permitiendo a los profesionales centrarse en un número mucho menor de informes. También queda patente la importancia en la elección de la técnica de Word/Doc Embeddings utilizada, dado que los resultados obtenidos varían claramente entre una herramienta que ofrece unos resultados buenos (Apartado 5.3.1 - TF-IDF) y una cuyos resultados no han sido como se esperaban (Apartado 5.3.2 – Doc2Vec).

El servicio desarrollado de Creación de Resúmenes Automáticos ha demostrado tener utilidad a la hora de presentar un avance sobre el contenido que el usuario se podrá encontrar dentro de un informe, pero en ningún caso los resultados obtenidos invitan a que estos resúmenes sirvan como sustitución total de los informes, dado que en estos documentos la información recogida no tiende a tener elementos que se puedan omitir sin perder información de valor.

Por último, el Sistema de Recomendación elaborado muestra un gran potencial debido a la integración del tanto de la herramienta de búsqueda del informe más cercano como el filtro colaborativo desarrollada, dado que considera más de un informe para realizar las predicciones. Además, en este módulo también se aprecia claramente la influencia de un creador de vectores correctamente elegido, dado que en el sistema elaborado para el *Cold Start* los informes más cercanos son claramente similares al que está siendo tratado en ese momento.

Aun así, en la elaboración de este proyecto se aprecia que el enfoque principal debería de haber sido la comparación de informes para poder recomendar campos según otra información, mientras que en el proyecto se ha enfocado mayormente en la elaboración del

filtro colaborativo en vez de haber puesto como foco central las técnicas utilizadas solamente para resolver el problema de *Cold Start*.

Como conclusión a este apartado, los resultados obtenidos muestran una gran utilidad como herramientas de ayuda a los profesionales y que son capaces de suponer una reducción en el tiempo necesario para el análisis de casos de pacientes, sobre todo para en el estudio del historial de datos para saber en qué información apoyarse.

En contraparte a lo anterior, la mayor limitación de esta herramienta es que en ningún caso debería ser utilizada como tomadora de decisiones, sino que necesitará de la interpretación de los expertos para ofrecer un mejor servicio.

Capítulo 7. CONCLUSIONES Y TRABAJOS FUTUROS

7.1.1 CONCLUSIONES

Trabajar utilizando una fuente de datos sin etiquetar y perteneciente a unas áreas de estudio tan complejas como es la salud y la biomedicina supone la obligación de crear herramientas que no sean utilizadas tomar decisiones directamente. Es debido a esto por lo que el proyecto ha adquirido un enfoque orientado a crear sistemas de apoyo para los profesionales.

Además, durante todo el procesamiento realizado sobre los datos se ha mantenido siempre, mediante el uso de las clases, la procedencia de cada una de las variables que representan a los informes, de forma que los profesionales puedan comprobar el texto original sobre el que se basa cada uno de los servicios realizados y que en ningún caso se omita información directamente al responsable de tomar las decisiones.

Las herramientas de clustering de informes suponen una gran utilidad dado que limitan la cantidad de documentos que han de ser analizados por un profesional gracias a descartar todos aquellos que no son relevantes para un caso en concreto, y claramente se aprecia que la mayor importancia recae sobre la técnica de *Word Embedding* elegida, dado que es la parte encargada de representar el texto de la forma más fidedigna posible.

Por otro lado, el generador de resúmenes automáticos supone la omisión de información dentro de un informe y, aunque el modelo pondere las frases más importantes como relevantes, el contenido de los documentos suele estar limitado a temas relevantes.

Por último, aunque todo sea funcional y ofrezca los resultados esperados, se ha apreciado claramente que el mayor potencial del sistema de recomendación desarrollado reside en la capacidad de predecir nuevos campos de los que no se disponga información basándose en el contenido del resto de apartados, más que el uso de la matriz colaborativa tal como se ha explicado en el apartado correspondiente.

7.1.2 TRABAJO FUTURO

Como trabajo futuro se podrían definir dos visiones complementarias.

A corto plazo se podrían implementar más sistemas de *Word Embeddings* y comprobar su funcionamiento en este proyecto, complementando así el sistema basado en *TF-IDF*, mediante el uso de modelos de *Machine Learning* de mayor complejidad que el desarrollado con *Doc2Vec*.

Además, en el sistema de recomendación sería aconsejable integrar más sistemas de comparativa de la similitud de los informes además del ya hecho mediante la Distancia del Coseno, potenciando de esta forma la recomendación sobre decisiones futuras a partir del histórico de datos disponible, tal como se ha explicado en las conclusiones. Una recomendación como trabajo futuro es el de desacoplar este sistema de comparativa del de recomendación mediante filtros colaborativos de forma que supongan dos servicios complementarios, pero a su vez diferenciados.

Por otro lado, a largo podría considerarse también una modificación del proceso de integración con MetaMap para considerar un mayor número de salidas de la aplicación, analizando si el resto de los campos no utilizados en este proyecto supondrían de utilidad en la complementación del conocimiento del metatesauro UMLS con el área del NLP. Como el foco principal de este proyecto es la integración del conocimiento del metatesauro UMLS y el procesamiento de lenguaje natural, avanzar en esta área podría suponer explotar todo el potencial de esta herramienta.

Además, conseguir datos etiquetados para poder entrenar de una forma más eficiente los modelos utilizados (*BERTSUM*, los utilizados en *Word Embedding*…) supondría una mejor implementación de éstos dentro de la aplicación y posibilitaría una mejora sustancial en todos los módulos elaborados.

En este proyecto se ha visto el potencial de integrar herramientas tan modernas como las de NLP en un ámbito como el de la medicina, pero también ha quedado claro la necesaria intervención de profesionales con conocimiento de ambas áreas (programáticas y de la salud) para que el tratamiento de los datos sea lo más cuidadoso posible.

Capítulo 8. BIBLIOGRAFÍA

[1] M. V. I. K. Milan Kubina, *Use of Big Data for Competitive Advantage of Company*, 2015.

[2] E. M. Voorhees, «The TREC Medical Records Track,» de *National Institute of Standards and Technology*.

[3] «National Library of Medicine,» [En línea]. Available: https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html.

[4] P. P. C. Z. Y. Y. J. C. A. M. W. V. B. S. Fei Zhu, *Biomedical text mining and its applications in cancer research*, 2 ed., vol. 46, 2013, pp. 200-211.

[5] National Institute of Health, «National Library of Medicine,» [En línea]. Available: https://www.nlm.nih.gov/research/umls/implementation_resources/metamap.html.

[6] NLTK, «NLTK,» [En línea]. Available: https://www.nltk.org/.

[7] Spacy, «Spacy,» [En línea]. Available: https://spacy.io/.

[8] «Scikit-Learn,» [En línea]. Available: https://scikit-learn.org/stable/.

[9] «Matplotlib,» [En línea]. Available: https://matplotlib.org/.

[10] X. Pornain, «wired,» [En línea]. Available: https://www.wired.com/insights/2014/07/rewiring-tackle-unstructured-data/.

[11] ODSC, «Medium,» Open Data Science, 26 July 2019. [En línea]. Available: https://medium.com/@ODSC/15-open-datasets-for-healthcare-830b19980d9.

[12] I. A. Martínez, *Análisis y optimización del recurso UMLS en la recuperación de la información biomédica mediante métricas de similitud semántica*, Madrid, 2016.

[13] The Bioinformationist, [En línea]. Available: https://bioinformationist.wordpress.com/2012/12/21/semantic-medline-the-use-of-semantic-web-in-pubmed-searches/.

[14] M. C. T. M. P. L. O.-M. Son Doan, *NATURAL LANGUAGE PROCESSING IN BIOMEDICINE: A UNIFIED SYSTEM ARCHITECTURE OVERVIEW*, San Diego.

[15] C. M. Y. Z. H. V. L. Y. Y. Z. L. R. G. J. J. P. Y. L. Himanshu Sharma, «Developing a portable natural language processing based phenotyping system,» de *The Sixth IEEE International Conference on Healthcare Informatics* , New York, 2018.

[16] G. O. B. Henry J. Lowe, «MicroMeSH: A Microcomputer System for Searching and Exploring the National Library of Medicine's Medical Subject Headings (MeSH) Vocabulary,» 4 November 1987 .

[17] F. M. G. J. K. V. G. F. C. R A Miller, «CHARTLINE: providing bibliographic references relevant to patient charts using the UMLS Metathesaurus Knowledge Sources,» 1992.

[18] R. A. G. W R Hersh, «SAPHIRE--an information retrieval system featuring concept matching, automatic indexing, probabilistic retrieval, and hierarchical relationships,» 23 October 1990.

[19] P. G. Chakraborty, 2013. [En línea]. Available: https://www.researchgate.net/figure/Text-mining-process-Source-Chakraborty-Pagolu-and-Garla-2013_fig1_262413948.

[20] A.-H. Tan, «Text Mining:The state of the art and the challenges,» Kent Ridge Digital Labs, Singapore, 2000.

[21] A. Agrawal, «Medium,» 6 June 2020. [En línea]. Available: https://medium.com/dataseries/the-current-state-of-the-art-in-natural-language-processing-nlp-5c440f889e15.

[22] C. Santana, *La Siguiente Gran Revolución: NLP (Procesamiento del Lenguaje Natural)*, 2020.

[23] Facebook, «Facebook Ai,» 29 April 2020. [En línea]. Available: https://ai.facebook.com/blog/state-of-the-art-open-source-chatbot/.

[24] M.-W. C. Jacob Devlin, «Google AI Blog,» 1 November 2018. [En línea]. Available: https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html.

[25] K. L. A. C. Iz Beltagy, *SciBert: A Pretrained Language Model for Scientific Text*, Seattle, WA: Allen Institute for Artificial Intelligence, 2019.

[26] Centers for Medicare & Medical Services, «cms,» 03 04 2021. [En línea]. Available: https://www.cms.gov/medicare/coding/icd10.

[27] K. C. G. C. D. Tomas Mikolov, *Efficient Estimation of Word Representations in Vector Space*, 2013.

[28] P. T. Rada Mihalcea, *TextRank: Bringing Order into Texts*, Texas.

[29] D. R. R. Güneş Erkan, *LexRank: Graph-based Lexical Centrality as Salience in Text Summarization*, Michigan.

[30] H. P. Luhn, «The Automatic Creation of Literature Abstracts,» *IBM Journal of Research and Development*, vol. 2, nº 2, pp. 159-165, 1958.

[31] A. H. L. Vanderwende, «Proceedings of Human Language Technologies,» de *Association for Computational Linguistics*, Boulder, Colorado, 2009.

[32] Y. Liu, *Fine-tune BERT for Extractive Summarization*, Edinburgh, 2019.

[33] Google AI, «Google Blog,» 24 February 2020. [En línea]. Available: https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html.

[34] Y. L. N. G. M. G. A. M. O. L. V. S. L. Z. Mike Lewis, *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension*, Facebook AI.

[35] J. W. R. C. D. L. D. A. Alec Radford, *Language Models are Unsupervised Multitask Learners.*

[36] A. C. Guillaume Lample, *Cross-lingual Language Model Pretraining*, Facebook AI.

[37] E. H. H. L. J.-J. L. A. Pernille Warrer, «Using text-mining techniques in electronic patient records to identify ADRs from medicine use,» de *British Journal of Clinical Pharmacology*, The British Pharmacological Society, 2011, pp. 674-684.

ANEXO I - ODS

Los Objetivos de Desarrollo Sostenible son una serie de principios definidos por las Naciones Unidas con el objetivo de asegurar para el futuro el fin de la pobreza, la sostenibilidad del planeta y la prosperidad de todos aquellos que viven en éste.

Los objetivos desarrollados en este proyecto se han alineado con varias de las pautas definidas por la ONU buscando no solamente crear un servicio, sino que además esta herramienta se alinee con los valores recogidos en las ODSs.

En concreto, el proyecto se alinea con 2 objetivos en concreto:

3 – Salud y Bienestar. El dominio de la salud sobre el que se realiza este proyecto supone la necesidad de crear un producto que busque mejorar la calidad de los servicios de salud ofrecidos por los profesionales de este sector, y la focalización del proyecto en facilitar el trabajo de los médicos y especialistas supone un claro apoyo a favor del desarrollo de este sector en concreto.

9 – Industria, innovación e infraestructura. Por otro lado, este proyecto supone la integración de las nuevas tecnologías de análisis de datos no estructurados con el sector de la salud y la biomedicina, sobre el que existe una gran cantidad de profesionales con una gran experiencia en el mismo. Este proyecto supone un claro caso de innovación en el sector con el objetivo de acercar estas nuevas técnicas de análisis de datos para facilitar el trabajo de los profesionales.

La importancia de trabajar a favor de un desarrollo sostenible en nuestro planeta es vital, tal como se ha querido enseñar a la sociedad con la definición de los 17 Objetivos de Desarrollo Sostenible, y durante el desarrollo de este proyecto también se ha querido estar alineado con esta visión propuesta, orientando los objetivos, metodología y herramientas desarrolladas a favorecer un desarrollo más sostenible para el mundo.