Augusto Villa Monte

Generación automática inteligente de resúmenes de textos con técnicas de soft computing

Tesis Doctoral en Ciencias Informáticas
PREMIO DR. RAÚL GALLARD | Año 2020

edulp
EDITORIAL DE LA UNLP

Generación automática inteligente de resúmenes de textos con técnicas de *soft computing*

Augusto Villa Monte

UNIVERSIDAD NACIONAL DE LA PLATA

UNIVERSIDAD DE CASTILLA-LA MANCHA

TESIS DOCTORAL EN COTUTELA

Generación automática inteligente de resúmenes de textos con técnicas de *soft computing*

Augusto Villa Monte

Directores

Dra. LAURA LANZARINI (UNLP, Argentina)

Dr. JOSÉ A. OLIVAS (UCLM, España)

La Plata, Argentina
Marzo de 2019

EDITORIAL DE LA UNLP

Villa Monte, Augusto
Generación automática inteligente de resúmenes de textos con técnicas de soft computing / Augusto Villa Monte. - 1a ed. - La Plata: EDULP, 2021.
Libro digital, PDF

Archivo Digital: descarga y online
ISBN 978-987-8348-96-4

1. Tecnología Informática. I. Título.
CDD 004.02

Generación automática inteligente de resúmenes de textoscon técnicas de soft computing

Augusto Villa Monte

EDITORIAL DE LA UNIVERSIDAD NACIONAL DE LA PLATA (EDULP)
48 Nº 551-599 4º Piso/ La Plata B1900AMX / Buenos Aires, Argentina
+54 221 44-7150
edulp.editorial@gmail.com
www.editorial.unlp.edu.ar

Edulp integra la Red de Editoriales de las Universidades Nacionales (REUN)

ISBN 978-987-8348-96-4

Queda hecho el depósito que marca la Ley 11.723

Esta tesis se ha desarrollado siguiendo las líneas de investigación queel Instituto de Investigacion en Informática LIDI (III-LIDI, Argentina) y el grupo de investigación Soft Managemement of Internet and Learning (SMILe, España) llevan a cabo de manera colaborativa.

Contó con el apoyo externo de los profesores doctores Cristina Puente (Universidad Pontificia Comillas), Aurelio F. Bariviera (Universidad Rovira i Virgili) y Alejandro Sobrino (Universidad de Santiago de Compostela).

Fue presentada por Augusto Villa Monte, en el marco de su doctorado en cotutela, como requisito para obtener el grado de doctor en Ciencias Informáticas por la Universidad Nacional de La Plata (UNLP, Argentina) y doctor en Tecnologías Informáticas Avanzadas por la Universidad de Castilla-La Mancha (UCLM, España).

A computer would deserve to be called intelligent if it could deceive a human into believing that it was human

— Alan M. Turing, 1950

Este trabajo es el resultado de mucho esfuerzo y dedicación.Dedico esta tesis a mi familia y amigos, a quienes quiero y agradezco profundamente.

Mi más sincero agradecimiento a mis directores de tesis, Laura Lanzarini y José Ángel Olivas, por brindarme su conocimiento, experiencia y confianza. En especial, a Cristina Puente, Alejandro Sobrino y Aurelio F. Bariviera, por colaborar en el desarrollo de esta tesis. Al instituto III-LIDI y al grupo SMILe, por la ayuda brindada. También a todos los alumnos, profesores y compañeros de trabajo con quienes compartí, tanto en Argentina como España, esta etapa de mi vida. Fundamentalmente, a mi familia y amigos, quienes esperaron estos añosverme finalizar la carrera de posgrado.

Por último, tanto a la Universidad Nacional de La Plata como a la Universidad de Castilla-La Mancha, por apoyar la realización de mi doctorado en cotutela.

AUGUSTO VILLA MONTE

Índice

CAPÍTULO 1

Introducción 17

1.1. Motivación 18

1.2. Objetivos 20

1.3. Contribuciones 21

1.4. Publicaciones 23

1.5. Organización 24

CAPÍTULO 2

Obtención de resúmenes automáticos 26

2.1. Introducción 27

2.2. Procesamiento de lenguaje natural 29

2.3. Proceso de extracción de conocimiento a partir de texto 31

2.4. Minería de datos, de texto y de la Web 35

2.5. Documentos y colecciones 37

2.6. Arquitectura general de sistemas de minería de textos 40

2.7. Recolección y selección 41

2.8. Preprocesamiento del corpus 44

2.9. Resúmenes de texto 56

2.10. Representación de documentos 60

2.11. Ponderación de sentencias 67

2.12. Conclusiones 74

CAPÍTULO 3

Resumen utilizando una técnica de optimización mediante cúmulo de partículas 75

3.1. Técnicas de optimización 76

3.2. Optimización mediante cúmulo de partículas 78

3.3. Método propuesto basado en PSO 82

3.4. Algoritmo de optimización utilizado 82

3.5. Representación de los individuos 87

3.6. Aspectos fundamentales del método 88

3.7. Diseño de la función de aptitud 90

3.8. Experimentación y resultados 95

3.9. Conclusiones 101

CAPÍTULO 4

Resumen mediante grafos causales y concomponentes temporales 103

4.1. Introducción a la causalidad 104

4.2. Fundamentos básicos de una relación causal 105

4.3. La causalidad y el tiempo en Medicina 108

4.4. Representación de información temporal 111

4.5. Relaciones de tiempo de Allen 113

4.6. Borrosificación de la ontología de Allen 114

4.7. Extracción y análisis de relaciones causales 117

4.8. Creación del grafo causal 120

4.9. Anotaciones de temporalidad en el grafo 142

4.10. Un enfoque práctico sobre el tema 147

4.11. Sistema de alerta para una adecuada administración demedicamentos 150

4.12. Conclusiones 159

CAPÍTULO 5

Conclusiones finales y futuras líneas de trabajo 161

5.1. Conclusiones generales 161

5.2. Representación de la información 163

5.3. Extracción de contenido relevante 163

5.4. Resumen utilizando grafos 165

5.5. Construcción del grafo usando métricas 167

5.6. Conclusiones finales 168

Bibliografía 169

ANEXOS 189

APÉNDICE A

Extracción de reglas de clasificación 191

A.1. Introducción 192

A.2. Método propuesto 194

A.3. Resultados obtenidos 199

A.4. Conclusiones 203

APÉNDICE B

Almacenamiento de documentos de maneraestructurada 207

B.1. Introducción 208

B.2. Descripción del diseño propuesto 210

B.3. Conclusiones 217

Anexos

A Extracción de reglas de clasificación **191**

A.1 Introducción 192
A.2 Método propuesto 194
A.2.1 Información a manejar en cada individuo 195
A.2.2 Inicialización de la población 196
A.2.3 Aptitud de una partícula 197
A.2.4 Desplazamiento de las partículas 198
A.2.5 Proceso de obtención de reglas 199
A.3 Resultados obtenidos 199
A.4 Conclusiones 204

B Almacenamiento de documentos de manera estructurada **207**

B.1 Introducción 208
B.2 Descripción del diseño propuesto 210
B.3 Conclusiones 217

Lista de figuras

1 Interacción de la *Lingüística* con distintos campos 30
2 Pasos que componen el proceso de *extracción de conocimiento en bases de datos* 32
3 Visión general del proceso de *extracción de conocimiento en textos* 34
4 Clasificación de la *minería web* según el aspecto de la web que se trate 35
5 Modelo simple de entrada/salida para sistema de *minería de* texto 40
6 Relación entre tareas de preprocesamiento de texto 44
7 Representación de bolsa de palabras utilizando pesos booleanos 62
8 Representación del *modelo de espacio vectorial* para tres documentos de tres términos. 63
9 Matriz "documento-término" correspondiente al VSM. Las filas representan cada uno de los documentos y las columnas los términos diferentes que ocurren en ellos w_{jk} es el peso del término k para el j-ésimo documento. 64
10 Ejemplos de inteligencia de cúmulo en la naturaleza 78
11 Movimiento de una partícula del instante t al instante $t + 1$ siendo $p_i(t)$ su posición actual, $p_i(t+1)$ su siguiente posición, $v(t)$ la velocidad actual (inercia), $pBest_i(t)$ la mejor solución hallada hasta el momento (memoria) y $theBest(t)$ la mejor solución del vecindario (cooperación). 81
12 Selección de métricas utilizando PSO 95
13 Metodología propuesta para el proceso de generación de resúmenes 96
14 Nivel de participación de las métricas ordenadas en orden descendente por valor de coeficiente. 100
15 Evolución de la precisión a medida que se agregan nuevas métricas para calcular el puntaje. 101
16 Representación (a) precisa y (b) difusa de un instante de tiempo. 115
17 Representación del intervalo difuso «between about 3 and about 6» 116
18 Estructuras condicionales y causales implementadas. 118
19 Proceso para obtener un grafo causal a partir de texto plano. 140
20 Grafo causal relacionado con «lung cancer» construido de forma automática 141
21 Representación gráfica de un nodo con sus modificadores. 141
22 Ejemplo de respuesta leyendo un grafo causal. 142
23 Causa según restricción temporal «before» representada en línea de tiempo. 144
24 Efecto según restricción temporal difusa «early» representado en línea de tiempo. 145

25 Grafo causal obtenido a partir de restricciones temporales. 146
26 Relaciones de similitud entre conceptos obtenidos por el proceso de filtrado y limpieza. 149
27 Relaciones causales limitadas por restricciones temporales. 150
28 Base de datos utilizada para la administración oportuna de medicamentos. 152
29 Proceso de control para administrar dosis de medicamentos críticos a lo largo del tiempo. 154
30 Pantalla inicial de la aplicación “My Medicine”. 156
31 Pantallas del panel de configuración. 156
32 Pantallas de restricciones temporal de la aplicación 158
33 Resultados obtenidos sobre 13 bases de datos de repositorio. 199
34 Simplicidad del modelo obtenido medida a partir de la cantidad de comparaciones totales realizadas en cada caso. 200
35 Resultados obtenidos sobre dos casos reales de empresas financieras que otorgan préstamos para consumo y dos bases de datos financieros de crédito al consumidor de repositorio. 202
36 Simplicidad del modelo. 202
37 Ejemplo de base de datos relacional compuesta por dos tablas. 208
38 Ejemplo de consulta SQL sobre los datos de la figura 37. 208
39 Estructura típica de un documento científico. 211
40 Modelo de base de datos propuesto para almacenar contenido textual de los artículos científicos. 214

Lista de tablas

1 Detalle de métricas que más se consideran en la literatura para representar documentos y obtener resúmenes extractivos..... 84
2 Coeficientes promedio obtenidos con cada métrica. Estos valores corresponden a la media y desviación correspondiente..... 116
3 Relaciones entre intervalos..... 130
4 Relaciones entre instantes e intervalos de tiempo..... 130
5 Relaciones entre instantes de tiempo. 130
6 Comparación de resultados obtenidos de la extracción y clasificación de sentencias causales por cada tipo de documento analizado. 136
7 Salida del POS Tagger de Stanford para la oración «If, forsome reason, surgery is not an option with early stage 1 and 2, treatment is usually radiation therapy»..... 142

Lista de acrónimos

A I Artificial Intelligence
A P I application Program
Interface AT S Automatic
Text SummarizationB I
Business Intelligence
B OW Bag of Words
C L Computational
LinguisticC S
Computer Science
C S S Cascading Style
SheetsD M Data
Mining
D U C Document Understanding
ConferencesG A Genetic Algorithm
H T M L HyperText Markup
LanguageH T T P HyperText
Transfer Protocol
I R Information Retrieval
I R L Iterative Rule Learning
I S F Inverse Sentence Frequency
I S M P Institute for Safe
Medication PracticesJ S O N
JavaScript Object Notation
K D D Knowledge Discovery
in DatabasesK D T Knowledge
Discovery in Texts
L SA Latent Semantic Analysis
LV Q Learning Vector Quantization
M C C Matthews Correlation
CoefficientM L Machine
Learning
N L P Natural Language
Processing O WA
Ordered Weighted
AveragingP L O S Public
Library of Science
P LY
Python
Lex-YaccP
O S Part-
Of-Speech

P S O Particle Swarm
OptimizationR E
Regular Expression
S I Swarm Intelligence
S O M Self-Organizing Map
S Q L Structured
Query LanguageT A
Text Analytics
T D M Text and Data
MiningT F Term
Frequency
T F - I D F Term Frequency – Inverse Document
FrequencyT M Text Mining
U M L S Unified Medical Language
SystemU R L Uniform Resource
Locator
V S M Vector Space
ModelW M Web
Mining
W W W World Wide Web
X M L eXtensible Markup Language

CAPÍTULO 1

Introducción

En la sociedad actual, la generación de datos acompaña al ser humano en la mayoría de sus actividades. Por eso el acceso y aprovechamiento de datos se han vuelto fundamentales en todos los ámbitos.

Son numerosas las áreas interesadas en extraer conocimiento a partirde la información almacenada, y más aún tratándose de información no estructurada. Disponer de herramientas capaces de representar la información digital resulta sumamente útil para la toma de decisiones.

En la actualidad, la mayor parte de la información digital acumulada es información textual. Generalmente, el texto se almacena en forma de documentos digitales. Estos documentos no están estructurados ni mucho menos organizados en bases de datos tradicionales. El texto porsí mismo no tiene ningún tipo de estándar ni restricción alguna y, porlo tanto, procesarlo se ha vuelto una tarea extremadamente difícil.

El desarrollo de soluciones computacionales que permitan resumir texto constituye una de las líneas de investigación que pretende reducir los problemas generados por el crecimiento desmedido de informacióntextual.

Esta tesis tiene por objetivo desarrollar dos estrategias capaces de resumir documentos de texto en forma automática. Por un lado, identificando el criterio del usuario para seleccionar las partes principales de un documento

y, por otro, extrayendo a partir de documentos patrones textuales específicos sumamente útiles en la tomade decisiones.

A continuación, se desarrollan la motivación, objetivos y aportes dela tesis. Además, se detallan las publicaciones científicas derivadas de este trabajo y la distribución del documento en capítulos.

1.1. Motivación

Muchos años después de que el escritor y científico Toffler (1970) pronosticara que se produciría más información de la que sería posible procesar, la explotación de datos y el descubrimiento de conocimiento se volvieron fundamentalmente necesarios en todos los ámbitos. Los avances tecnológicos logrados en los últimos tiempos favorecieron la generación de grandes volúmenes de datos y, comoresultado, el desarrollo de métodos inteligentes capaces de representarla información disponible se ha vuelto esencial (Fayyad *et al.*, 1996).

Si bien los sistemas y aplicaciones actuales continuamente generan datos en diferentes formatos, la mayoría de ellos producen y almacenantexto. Este formato resulta algo menos atractivo que otros como el sonido, las imágenes y el video, pero es, sin lugar a duda, el principal medio de comunicación entre seres humanos en la actualidad (Schreibman *et al.*, 2016). Cada correo electrónico enviado, cada búsqueda realizada en Internet y cada publicación subida a la red implica, en mayor o menor medida, datos en formato texto. Incluso si tenemos en cuenta noticias, libros, blogs, artículos científicos e historiasclínicas, entre otros tipos de documentos digitales, todos ellos están formados por mucho texto.

Desde la invención de la escritura en la antigüedad, el ser humano ha almacenado el conocimiento en textos y lo continúa haciendo. Desde entonces, la cantidad de documentos disponibles ha aumentado exponencialmente mientras el costo de generar, almacenar, duplicar y compartir dicha información fue disminuyendo (Saracco, 2017). Al mismo tiempo, el consumo de información aumentó notablemente. Hoy, una persona normal consume información la mitad del día (Johnson, 2011). Dicho consumo se realiza con algún propósito, ya sea investigar sobre un tema de tesis, encontrar la solución a un error de programación o averiguar los síntomas de una enfermedad. Luego, el conocimiento obtenido se lo aplica a diario en la toma de decisiones y esto es lo que resulta realmente importante.

A partir de la evidente explosión de información, sería ideal que el ser humano pudiera recordar absolutamente todo, pero, en ese caso, el cerebro colapsaría. Aunque exista un desacuerdo sobre la capacidad de memoria del cerebro humano (Reber, 2010), su capacidad cognitiva se encuentra limitada. La cognición es la facultad de un ser vivo para procesar información a partir de la percepción, el conocimiento adquirido (experiencia) y características subjetivas que permiten valorar la información. El ser humano recuerda muchas cosas, pero al mismo tiempo que olvida muchas otras. Inconscientemente capta la información esencial para mantenerla en su memoria. Esta tarea, tratándose de texto, se conoce como "resumir". Esta acción es una característica cognitiva de la inteligencia humana para retener lo esencial.

Como ya se mencionó anteriormente, en la actual era de la información y revolución digital abunda la información, especialmente la textual. Durante los últimos 60 años se han logrado grandes avances en lo que refiere a

resúmenes automáticos. Desde entonces, se ha intentado reducir automáticamente el tamaño de los documentos a un formato legible por el propio ser humano, y por qué no también por lascomputadoras.

En los últimos años se comenzaron a desarrollar programas de computadora con suficiente "capacidad cognitiva" para responder preguntas tales como «¿Cuáles fueron los puntos importantes de este artículo?» (Torres Moreno, 2014). En este sentido, contar con estrategias capaces de identificar automáticamente lo principal de un documento facilita su procesamiento, no solo en lo que se refiere a lectores humanos, quienes son beneficiados agilizándoles la lectura,sino también a técnicas de aprendizaje automático y de recuperaciónde información concretamente.

Desarrollar un programa de computadora que resuma automáticamente un documento requiere instrucciones precisas (Borko and Berniers, 1975). Existen dos grandes enfoques para tratar un documento. Uno de ellos tiene que ver con la interpretación del significado del contenido del documento (enfoque abstractivo) y el otro con el análisis más bien de su estructura (enfoque extractivo) (Hahn and Mani, 2000). En esta tesis se tratan ambos enfoques a través de la propuesta de dos soluciones diferentes aplicadas a documentos médicos.

1.2. Objetivos

El objetivo principal de este trabajo consiste en contribuir al área conformada por el *procesamiento de lenguaje natural* y la *minería de texto* con dos soluciones diferentes capaces de construir un resumen automático a partir de un

conjunto de documentos. Dicho objetivo se llevará a cabo a través de los siguientes subojetivos:

- Explorar los aspectos claves de la obtención de resúmenes automáticos haciendo énfasis en los involucrados específicamente con el desarrollo de las soluciones propuestas.
- Analizar las tareas que intervienen en el preprocesamiento de texto e identificar cuáles permitirán representar adecuadamente los documentos en cada una de las soluciones.
- Obtener el corpus de documentos a utilizar en cada caso y representarlo.
- Diseñar y desarrollar las dos soluciones propuestas capaces deidentificar a partir de los documentos de texto lo considerado relevante y por ende digno de ser conservado en el resumen final aconstruir.
- Determinar la relevancia del contenido causal de un documento y evaluar si es lo suficientemente importante como para formar un resumen extractivo.
- Realizar experimentos y evaluar los resultados obtenidos de laaplicación de las soluciones desarrolladas.

1.3. Contribuciones

Esta tesis contribuye proporcionando una descripción general del proceso de *extracción de conocimiento en texto*. Se hará hincapié en la etapa de preprocesamiento mostrando casos reales que dan cuenta de la importancia de esta etapa.

El aporte central de esta investigación radica en la definición y presentación de dos soluciones diferentes para obtener resúmenes automáticos a partir de documentos.

La primera de ellas, para documentos que poseen cierta estructura, permite crear resúmenes extractivos utilizando una técnica de o*ptimización mediante cúmulo de partículas* propuesta a partir de una representación vectorial de los mismos, basada en un conjunto amplio de métricas de puntuación de sentencias. Dicha técnica identificaráel criterio del usuario para seleccionar las partes del documento que considera importante. En lugar de utilizar las distintas métricas en forma independiente para construir el resumen, la respuesta del método sugiere la combinación que mejor se ajuste a la valoración que el usuario realizó de cada parte de un documento en forma previa.El método propuesto, que combina una representación binaria y una continua utilizando una variante original de la técnica mencionada, no solo permite identificar los coeficientes asociados a cada una delas métricas sino también cuáles métricas permiten resumir lo másparecido al criterio del usuario.

La segunda solución utiliza varios programas para extraer las sentencias causales y con restricciones temporales existentes en un conjunto de documentos médicos y luego convertir dichas oracionesen un grafo causal equivalente. Para ello, se identificarán y extraerán patrones textuales específicos del texto. El modelo resultante estará formado por las relaciones que describen el contenido de los documentos originales, mostrando únicamente todos los vínculos "causa-efecto" encontrados junto con las restricciones temporales que afecten su interpretación. Identificar estos patrones específicos en documentos médicos resulta sumamente útil para la

toma de decisionesen el área de salud. El grafo resultante, además, tiene la información necesaria para generar nuevas frases a partir del recorrido entre sus nodos y arcos.

1.4. Publicaciones

Esta tesis doctoral está avalada por las siguientes publicaciones científicas en las cuales el tesista es autor:

Document summarization using a scoring-based representation. In 2016 XLII Latin American Computing Conference, pages 1–7, 2016
Evaluation of causal sentences in automated summaries. In 2017 IEEE International Conference on Fuzzy Systems, pages 1–6, 2017
Simplifying credit scoring rules using LVQ+PSO. Kybernetes, 46(1): 8–16, 2017
Obtaining and evaluation of extractive summaries from stored text documents. In Proceedings of the Third Conference on Business Analytics in Finance and Industry, pages 65–66, 2018b
Designing a system to extract and interpret timed causal sentences in medical reports. Journal of Experimental & Theoretical Artificial Intelligence, 31(1): 1–13, 2019
Alert system for timely medication administration. In Proceedings of the 2018 International Conference on Artificial Intelligence (ICAI'18), pages387–392, 2018
Text pre-processing tool to increase the exactness of experimental results in summarization solutions. In Proceedings of the XXIV Argentine Congress of Computer Science, 2018a

A continuación, se listan otras publicaciones del tesista relacionadascon uno de los temas desarrollados en esta tesis:

E-mail processing usingdata mining techniques. In Computer Science & Technology Series: XVI Argentine Congressof Computer Science - Selected papers, pages 109–120. Edulp, 2011
E-mail processing with fuzzy SOMs and association rules. Journal of Computer Science and Technology, 11(01): 41–46, 2011b
Obtención de reglas de clasificación usando SOM+PSO. In Proceedings of the XVIII Argentine Congress of Computer Science, pages 210–219, 2012
Obtaining classification rules using lvqPSO. In Advances in Swarm and Computational Intelligence, volume 9140 of Lecture Notes in Computer Science, pages 183–193. Springer International Publishing, 2015a
Obtaining classification rules using LVQ+PSO: An application to credit risk. In Scientific Methods for the Treatment of Uncertainty in Social Sciences, volume 377 of Advances

in Intelligent Systems and Computing, pages 383–391. Springer International Publishing, 2015b
Analysis of methods for generating classification rules applicable to credit risk. Journal of Computer Science & Technology, 17, 2017.

1.5. Organización

Este documento está dividido en cinco capítulos y dos anexos que se describen brevemente a continuación.

C A P Í T U L O 1. Brinda una introducción al tema de tesis justificandola importancia de contar con soluciones capaces de sintetizar automáticamente el contenido de documentos de texto. En este capítulo inicial también se definirán tanto los objetivos de la investigación como sus principales contribuciones. Además, serán detalladas las publicaciones científicas que respaldan este trabajo.

C A P Í T U L O 2. Comienza con una descripción general de la *inteligencia artificial*, la *lingüística computacional*, el *procesamiento del lenguajenatural* y la *minería de textos*. Luego, se presenta un estudio y análisis de las tareas de preprocesamiento de textos y alternativas de representación de documentos, ambas necesarias para el desarrollo de los siguientes capítulos. Se introducen los dos tipos de resúmenes más importantes remarcando las diferencias entre ambos. Hacia el final del capítulo, se detallan las métricas de puntuación más utilizadas para obtener resúmenes extractivosy la representación de los documentos utilizada en el siguientecapítulo.

C A P Í T U L O 3. Presenta un método alternativo que permite generar automáticamente resúmenes extractivos de documentos al ponderar

adecuadamente las características de puntuación de oraciones. El aspecto principal del método propuesto es la identificación de aquellas características que están más cerca del criterio utilizado por el individuo al resumir. El capítulo comienza introduciendo las técnicas de optimización haciendo énfasis en la *optimización mediante cúmulo de partículas* utilizada para el desarrollo del método propuesto.

C A P Í T U L O 4. Desarrolla los conceptos de causalidad y temporalidad describiendo las razones por las cuales resulta de interés estudiarlos. En este capítulo se extraerá contenido específico a partir de documentos médicos. Se proporcionará un mecanismo de extracción y representación de sentencias causales afectadas por restricciones temporales cuya principal área beneficiaria será la salud. Será descrita una aplicación que a través de la identificación de ciertos patrones textuales específicos será útil para la toma de decisiones.

C A P Í T U L O 5. Resume las conclusiones de los capítulos anteriores y presenta algunas líneas de investigación futuras que darán continuidad a este trabajo.

A N E XO S A Y B. El primero describe cómo la técnica desarrollada en elcapítulo 3 pudo aplicarse en la obtención de reglas de clasificación y el segundo detalla el proceso de obtención de los documentos que conformaron el corpus utilizado en aquel capítulo.

CAPÍTULO 2

Obtención de resúmenes automáticos

La explosión de la información fue y continúa siendo un problema real presente en muchas áreas. A lo largo de la historia, la enorme cantidad de información disponible y la dificultad para procesar dicha información de forma manual han sido dos de los mayores obstáculos de la extracción de conocimiento.

En los últimos tiempos, la información textual en forma de documento digital se ha acumulado en enormes repositorios por sobre otros tipos de datos. La mayor parte de estos documentos no se encuentran estructurados y procesarlos se ha vuelto extremadamente difícil.

El resumen automático de texto ayuda a procesar de manera eficiente el volumen cada vez mayor de información textual que el ser humano simplemente no puede manejar.

Este capítulo presenta una mirada general de lo que involucra cada una de las tareas que deben llevarse a cabo necesariamente antes de obtener conocimiento a partir de texto. Se explica por qué se considera la obtención automática de resúmenes de texto un tema digno de ser investigado. Además, se describe el marco completo en el cual se desarrollan los dos capítulos centrales de esta tesis.

2.1. Introducción

En la sociedad actual, los datos acompañan al ser humano en todas sus actividades y se han vuelto cada vez más importantes. Actualmente, se convirtieron en el recurso más preciado de las empresas, organizaciones e instituciones. Si bien no existen reglas claras para calcular su valor,se los considera tan valiosos como el oro. Es que a través de ellos, por ejemplo, se busca generar mayores ingresos.

Los datos no valen nada si no se usan para tomar decisiones que permitan obtener beneficios concretos. Sin embargo, el crecimiento exponencial de información digital de los últimos años ha obstaculizadosu uso. Los datos se encuentran disponibles pero, al disponerse detantísimos datos, el tiempo necesario para analizarlos y procesarloses muy elevado. La obtención de conocimiento a partir de los datos,sin importar su tamaño e inaccesibilidad, resulta fundamental en estos casos.

Actualmente, Internet es el medio más popular para difundir información. Esto hace que los usuarios tengan una sobrecarga de información altísima. Hoy en día siendo que más del 80 % de los datos disponibles en el mundo se almacena en formato texto, su procesamiento automático termina siendo una tarea crucial (Miner et al., 2012).

Al tratarse de texto, el conocimiento se encuentra codificado en formato no estructurado y esto hace su extracción más difícil. Estáclaro que no todo lo que nos rodea es texto, pero se encuentra en todas partes (Schreibman *et al*., 2016). Desde correos electrónicos, reclamos de clientes, noticias periodísticas, páginas web hasta libros completos, artículos científicos,

reportes técnicos, publicaciones en redes sociales, entre otros, están repletos de texto.

En 2010, el CEO de Google, Eric Schmidt, declaró que cada dos días se producía el mismo volumen de información que la generada hastael año 2003. Esta es la situación actual, en la cual el crecimiento de la información no se detiene, el consumo diario de información se vuelve excesivo y la identificación automática de lo que es relevante sigue siendo un desafío.

Desafortunadamente, la mayor parte de la información que recuperamos hoy en día es irrelevante. Desde entonces, separar lo que es esencial de lo que no lo es ha demostrado ser una tarea difícil de realizar manualmente cuando el volumen de información es inmenso.

Según Kosala and Blockeel (2000), algunos de los problemas queencuentran los usuarios de Internet al interactuar con la información disponible son: (1) recuperación de información específica y relevante al utilizar los servicios de búsqueda, (2) extracción de conocimiento nuevo y útil a partir de los datos obtenidos tras el proceso de búsqueda, (3) visualización del contenido y tipo de información de manera personalizada según los gustos y preferencias de cada usuario.

El desarrollo de soluciones computacionales basadas en técnicas de *procesamiento de lenguaje natural* (NLP por sus siglas en inglés)y *minería de texto* (TM por sus siglas en inglés) constituye una de las líneas de trabajo que buscan actualmente reducir los problemas
generados por la sobrecarga de información textual. Dentro de estetipo de soluciones, la obtención de *resúmenes automáticos de texto* (ATS, por sus siglas en inglés) es una de ellas ya que reduce el enorme volumen de información no estructurada a su contenido más importante para facilitar su

manipulación. Esto permite obtener los contenidos principales de un documento en menos tiempo que si se los obtuviera en forma manual.

2.2. Procesamiento de lenguaje natural

Durante años se ha tratado de entender cómo pensamos, entendemosy actuamos. Desde sus comienzos, la *inteligencia artificial* (AI, por sus siglas en inglés) se ha ocupado de imitar patrones de razonamientoque permitan construir máquinas capaces de realizar tareas como silas hiciera el propio ser humano (Russell and Norvig, 2009). En la actualidad, la artificial intelligence (AI) abarca una gran variedad de campos, entre los cuales se encuentra el natural language processing (NLP). Este subcampo de la *Ciencia de la Computación* (CS, por sus siglas en inglés) se encarga de formular e investigar métodos y técnicas informáticas capaces de procesar y analizar grandes cantidades de datos en lenguaje natural.

La información existente en Internet y el conocimiento en general se describen en lenguaje natural. El lenguaje natural es una forma de comunicación imprecisa y ambigua. Su comprensión depende de cada situación, edad, cultura o incluso lugar. Dicho lenguaje no es más que el medio utilizado por los seres humanos para comunicarse en forma oral,gestual o escrita. El lenguaje revela muchos aspectos: pensamientos, creencias, sentimientos, etc. A diferencia de los lenguajes formales basados en la unión de símbolos previamente especificados, el lenguaje natural es complejo y espontáneo. Los datos textuales contienen mucho“ruido” por la naturaleza del propio lenguaje. Desde un punto de vista computacional, el lenguaje natural

es el punto de partida hacia lenguajes estructurados interpretables por una máquina.

La Lingüística es la ciencia que estudia el lenguaje humano y, por ende, se relaciona con muchos otros campos. La intersección de la *Lingüística* y la *Computación*, da origen a distintas ramas que se denominan de diversas maneras. La *lingüística computacional* (CL por sus siglas en inglés) es una de ellas y se orienta a la construcciónde modelos de lenguajes "entendibles" por las computadoras. En cambio, el NLP se ocupa más de los aspectos técnicos, algorítmicos y matemáticos de la aplicación de dichos modelos a grandes volúmenes de texto para estructurar, extraer y transformar la información contenida en ellos. Ambas disciplinas tienen al lenguaje natural como objeto de estudio pero considerándolo desde diferentes enfoques.

Figura 1: Interacción de la Lingüística con distintos campos.

Si bien inicialmente el NLP estaba relacionado con las gramáticas formales de Chomsky y la búsqueda de analizadores sintácticos,en la actualidad incluye

otras tantas áreas, como la extracción de conocimiento o la recuperación de información.

Esta tesis se encuentra relacionada tanto con los enfoques antesmencionados como con el denominado comúnmente con el nombre de *minería de texto*. Si bien pueden encontrarse diferentes definiciones, la mayoría de los autores la definen como el proceso de analizar texto no estructurado mediante técnicas de aprendizaje automático para extraer conocimiento útil para propósitos particulares.

La *minería de texto* es un área que surge de la combinación de técnicas, herramientas y algoritmos de *minería de datos* (DM, por sus siglas en inglés), *aprendizaje automático* (ML, por sus siglas en inglés) y *procesamiento de lenguaje natural* (NLP, por sus siglas en inglés). Todas ellas se relacionan entre sí e intervienen en un proceso más general y complejo llamado *extracción de conocimiento en bases de datos* (KDD, por sus siglas en inglés) que les proporciona un marco de trabajo.

2.3. Proceso de extracción de conocimiento a partir de texto

En las últimas décadas, el avance de la tecnología ha permitido la generación de enormes repositorios de información digital. Desde entonces, ha crecido considerablemente el interés por extraer de formaautomática conocimiento a partir de los datos que se encuentran disponibles. Esto ha dado lugar a lo que se conoce como proceso de Knowledge Discovery in Databases (KDD).

Según Fayyad *et al*. (1996), "KDD es el proceso no trivial de identificar patrones válidos, novedosos, potencialmente útiles y en última instancia comprensibles a partir de los datos". El conocimiento obtenido como

resultado final de este proceso es lo que da valor agregado al almacenamiento de la información y resulta de gran ayuda al momento de tomar decisiones.

Este proceso ha sido descrito con distinto nivel de detalle por muchosautores. Sin embargo, pareciera haber consenso en reconocer al menostres grandes etapas que pueden verse claramente en la figura 2: la primera, relacionada con la recolección y preparación de la informacióncon la cual se va a trabajar, la segunda, referida a la construcción del modelo aplicando técnicas de *minería de datos* y la última, que consiste del análisis e interpretación del modelo obtenido y eventualmente su comunicación a quienes deben tomar decisiones.

La *minería de datos* es la parte del proceso de KDD que reúne el conjunto de técnicas que permiten obtener los patrones mencionados previamente. Su objetivo es generar una representación alternativa de

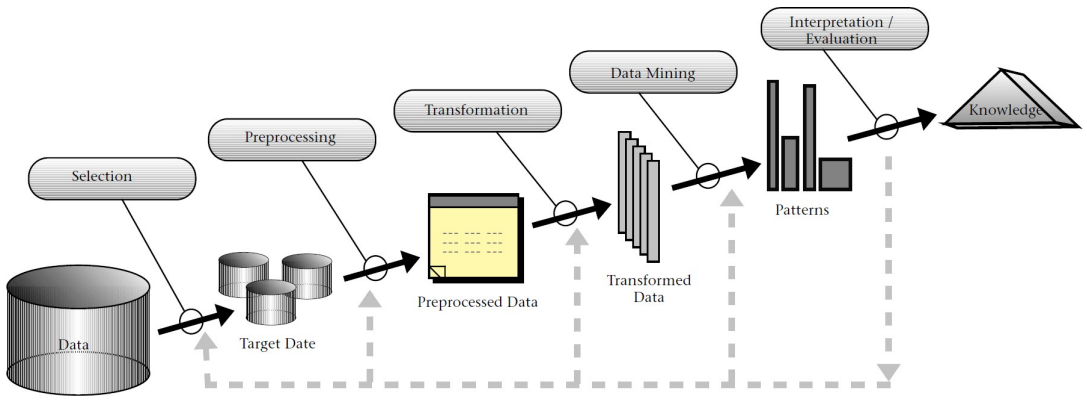


Figura 2: Pasos que componen el proceso de extracción de conocimiento en bases de datos.

Fuente: (Fayyad et al., 1996)

la información que deje de manifiesto las relaciones existentes en ellos. Luego, a partir de su análisis, se podrán comprender dichas relaciones y así obtener

conocimiento que permita que, por ejemplo, las empresas tomen medidas correctivas y/o preventivas.

No obstante, para extraer el conocimiento deseado es preciso analizar en profundidad los datos antes de aplicar cualquier técnica. Luego, una vez obtenidos los patrones, es necesario interpretarlos para determinar su utilidad y en caso de ser necesario, revisar uno, varios o todos los pasos previos del proceso. Gran parte del éxito de un proceso de KDD reside en el procesamiento previo que debe realizarse sobre los datos.

De todos los modelos de DM que pueden trabajarse, pareciera serque las técnicas de extracción de reglas son las preferidas de quienes deben tomar decisiones. Las reglas tienen capacidad de explicarse por sí mismas y de su lectura se obtiene una justificación inmediata de las decisiones sugeridas. En el anexo A, se describirá la manera de obtener reglas de clasificación aplicando la técnica utilizada en el capítulo 3 pararesumir.

Tanto las reglas de clasificación como las otras técnicas de DM, de la misma manera que el resto de etapas del KDD, asumen datos estructurados provenientes, generalmente, de bases de datos relacionales o planillas de cálculo. En los últimos años, la predominancia de información textual proveniente de redes sociales, periódicos en línea, correos electrónicos o chats (entre tantas fuentes) ocasionó que el proceso de KDD sirva de marco para organizar el conjunto de tareasque involucran el desarrollo de sistemas de minería de textos.

La *minería de texto* es un aspecto importante de *inteligencia de negocios* (BI, por sus siglas en inglés) que ayuda a los usuarios y empresasa analizar mejor el texto almacenado para tomar mejores decisiones, mejorar la satisfacción del cliente y obtener una ventaja competitiva (Gupta and Narang, 2012).

A diferencia del proceso de KDD tradicional, el proceso de *descubrimiento de conocimiento en textos* (KDT, por sus siglas del inglés)parte únicamente de texto en forma de documento y se lo preprocesa utilizando un conjunto de tareas que se aplican exclusivamente a dicho tipo de dato (desarrolladas en la sección 2.8). A partir del texto preprocesado, se lo transforma y representa adecuadamente (generando características) para poder continuar con el procesoaplicando las técnicas específicas de Text Mining (TM) y/o de Data Mining (DM) (adaptadas de ser necesario) que permiten descubrir patrones (Liang and Tan, 2007). Al poderse aplicar tanto técnicas de TMo DM, dependiendo de la representación utilizada, hay quienes llaman esta etapa directamente *minería de datos y de texto* (TDM). Como puede verse en la figura 3, el knowledge discovery in texts (KDT), al igual queel KDD, no es un proceso lineal.

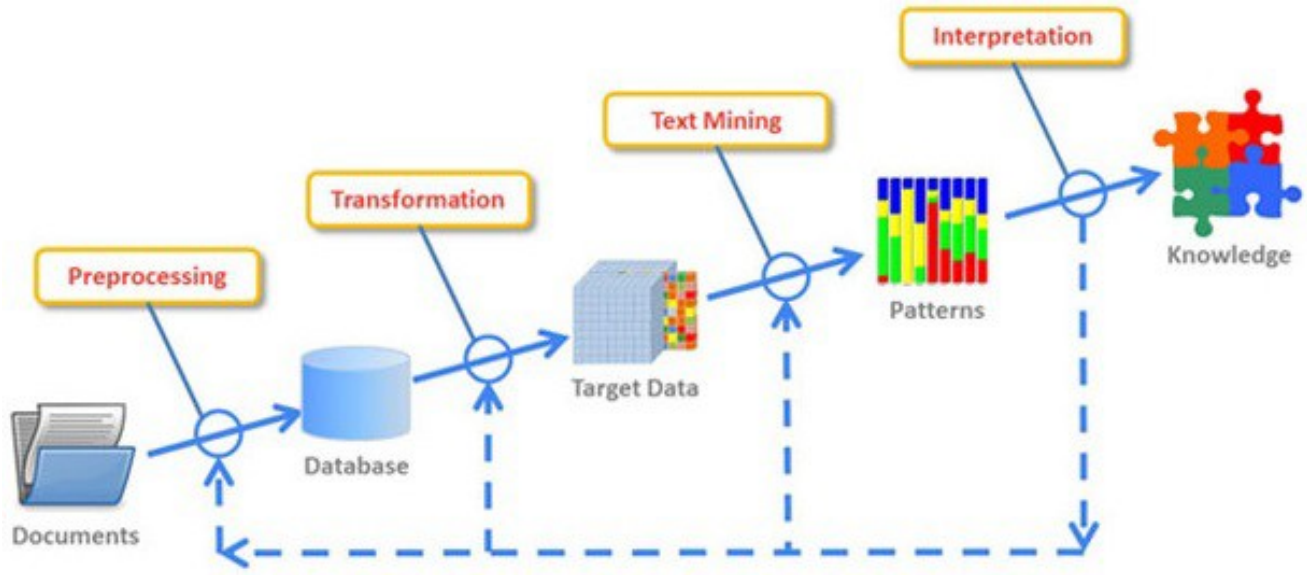


Figura 3: Visión general del proceso de Extracción de Conocimiento en Textos.

Fuente: `tlab.it/img/text_mining03.jpg`

2.4. Minería de datos, de texto y de la Web

Data mining (DM) es la etapa del KDD a través de la cual se descubre de manera automática, a partir de grandes repositorios de datos, información útil en forma de patrones. Incluye técnicas estadísticas y de aprendizaje automático para extraer el conocimiento. *Text mining* (TM) es un caso particular de DM a través del cual sebusca extraer información, pero a partir de texto únicamente. Tanto las tareas de preparación de datos como las técnicas que luego se aplican difieren significativamente de las utilizadas en DM. Durante el preprocesamiento, DM asume datos estructurados para trabajar mientras que, en TM, la identificación y extracción de características representativas de los documentos utilizando técnicas de NLP es fundamental. Generalmente, los documentos se transforman en datos estructurados para poder ser analizados y utilizados (véase el anexo B).

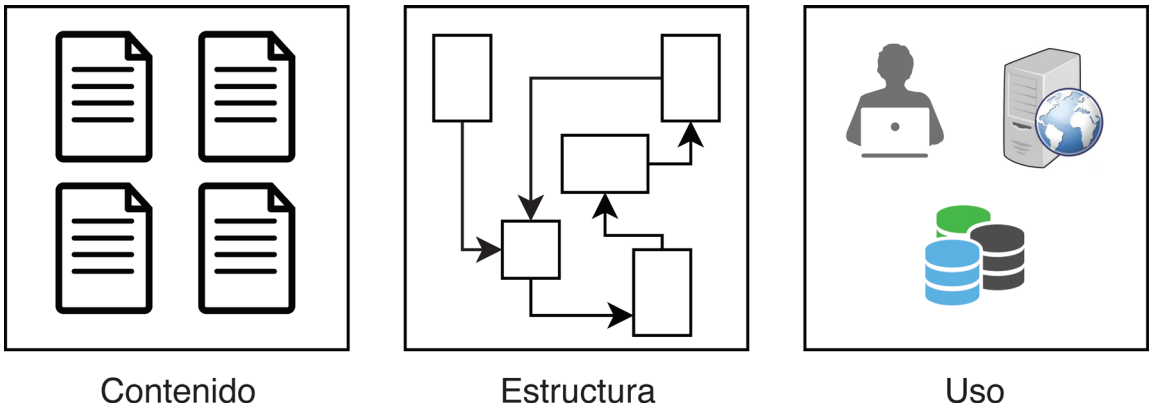


Figura 4: Clasificación de la minería web según el aspecto de la web que se trate.

Web mining (WM) tiene muchas similitudes con el TM, pero sus técnicas difieren significativamente al incorporar tareas propias de la Web. La World Wide Web (WWW) es el repositorio más grande existente donde los documentos contienen datos de muy diverso tipo: texto, imágenes, audio,

hipervínculos, etc. Esta diversidad, dada por datos no estructurados o semiestructurados, permite minar la web basándose en tres conceptos, tal como puede verse en la figura 4: el contenido, la estructura y el uso (Hernández Orallo *et al*., 2004).

Un documento en este caso es una colección de caracteres (texto) que puede contener, a través de los hipervínculos, referencias a otros documentos distribuidos en la web. Estos documentos o páginas web están escritos en una gran diversidad de idiomas y abarcan todos los tópicos del conocimiento humano.

Desde su aparición en 1990, la web ha experimentado un crecimiento exponencial. Desde entonces, su objetivo fue compartir informacióna través de computadoras, permitiendo la comunicación entre las personas. Para ello, se desarrollaron: (I) un identificador de recursos universal (URL por sus siglas en inglés) para referirse a cualquier documento (u otro tipo de recurso), (II) un protocolo de transferencia de hipertexto (HTTP por sus siglas en inglés) utilizado para el intercambio de información, y (III) un lenguaje de marcas para hipertexto (HTML por sus siglas en inglés).

Esta tesis tiene una mayor relación con la minería del contenidode la web ya que será analizado el contenido de varios documentos provenientes de Internet más que la presencia de enlaces o la información de accesos a dichos documentos.

Los documentos HyperText Markup Language (HTML) tienen marcas fundamentalmente de formato que suelen eliminarse para tratardirectamente con el texto. En los últimos años se incrementó el usode documentos semiestructurados a través de eXtensible markup language (XML) para representar datos con cierta estructura. Este formato, a diferencia del HTML,

permite marcar un documento por su contenido y no por cómo debe presentarse dicho contenido en un navegador web. Con XML el contenido queda desacoplado de su visualización facilitando su procesamiento.

2.5. Documentos y colecciones

Las técnicas de TM tienen el foco en colecciones de documentos para analizar. En términos prácticos, una colección de documentos, también llamada corpus, puede ser cualquier agrupación de documentosbasados en texto a partir de la cual se descubren patrones aplicando soluciones de minería de texto. Por ejemplo, si se analizara una encuesta con preguntas abiertas, el corpus estaría conformado por el conjunto de respuestas realizadas por los encuestados (Bécue-Bertaut, 2010).

En cuanto al tamaño de tales colecciones, este puede variar. Pueden ser estáticas o dinámicas según permitan incluir documentos nuevos o actualizados a lo largo del tiempo. Un ejemplo de colección típica de documentos del mundo real es *Medline*, una base de datos bibliográfica producida por la *Biblioteca Nacional de Medicina de los Estados Unidos*.[1]

Otro ejemplo es PLOS Medicine[2], una revista de acceso abierto en la cual se publican artículos de interés general sobre los principales desafíos de la salud humana en todo el mundo. Este tipo de repositorios reciben gran atención por parte de investigadores interesados en emplear técnicas de minería de textos.

[1] https://medlineplus.gov/
[2] https://journals.plos.org/plosmedicine/

Generalmente, los repositorios ofrecen algún servicio en línea a través del cual se accede a los resúmenes de los documentos o a los documentos de forma completa. El tamaño de estos repositorios hace que los intentos manuales por relacionar los documentos o identificar tendencias, por ejemplo, sean tareas extremadamente costosas y en el peor de los casos imposibles de realizar. Las técnicas de exploración automática de documentos provistas por la *minería de texto* no son solo un complemento útil para los lectores de dichos documentos, sino también un requisito básico para reconocer patrones en una gran cantidad de documentos que permitan automatizar otro tipo de tareas.

Un documento es el elemento básico de la minería de textos. A efectos prácticos, un documento puede definirse como una unidad de datos textuales dentro de una colección. Sin embargo, un documento no necesariamente existe dentro del contexto de una colección particular.

Los informes comerciales, los correos electrónicos, los boletines oficiales de estado, los artículos de investigación, los comunicados de prensa,las tesis o las noticias periodísticas son solo algunos ejemplos de documentos.

Los documentos pueden pertenecer a cualquier tipo de colecciones, desde las organizadas formalmente hasta las construidas en forma *ad hoc.* Incluso, un documento puede integrar diferentes colecciones al mismo tiempo o diferentes subconjuntos de la misma colección de documentos. Por ejemplo, cualquiera de los artículos científicos publicados en PLOS Medicine también se encuentran en *PubMed*[3].

[3] https://www.ncbi.nlm.nih.gov/pubmed/

A pesar de considerarse el contenido de un documento información no estructurada, puede verse al documento como un objeto con cierta estructura. En este contexto, los documentos digitales se pueden definir como un soporte de información que constituye una unidad estructurada. Desde una perspectiva lingüística, un documento tiene cierta estructura semántica y sintáctica, aunque dicha estructura esté implícita y, hasta cierto punto, oculta en su contenido textual. Además, el uso de signos de puntuación, mayúsculas, caracteres numéricos y especiales, combinados con espacios en blanco, saltos de línea, asteriscos, entre otros, sirve como un primer marcado leve del contenido. Esto, sumado a la incorporación de otro tipo de marcas para ayudar a identificar los subcomponentes importantes de los documentos como títulos, autores, párrafos, encabezados y notas al pie, permite estructurar aún más el contenido del documento. Sin ir más lejos, la secuencia de palabras también se considera estructurada.

En general, los documentos que no utilizan etiquetas de marcado para denotar su estructura tienen un formato libre y se denominan documentos con una estructura débil o directamente sin estructura. El resto de los documentos, como la mayoría de los trabajos de investigación científica, informes comerciales, expedientes judiciales y libros completos, son algo más extensos y consistentes, a partir de los cuales puede inferirse su estructura. Su contenido se estructura de manera tal que con algo de esfuerzo puede recuperarse exactamente lo que se quiere del texto. A pesar de eso, se consideran documentos semiestructurados ya que, en muchos casos, no poseen una estructura clara y no resulta fácil identificarla.

2.6. Arquitectura general de sistemas de minería de textos

En general, los sistemas de TM toman como entrada un conjunto de documentos sin procesar y generan varios tipos de salida, tal como puede verse en la figura 5 (Feldman and Sanger, 2006). Aunque, desde un punto de vista centrado en el ser humano, dicho sistema de entrada-salida resulta algo más complejo. En este caso, el usuario forma parte del descubrimiento de conocimiento a partir del texto. El sistema podría verse como un bucle interactivo y prolongado de consultas seguidas de acciones de análisis y refinamiento que permiten obtener una serie de resultados que, a su vez, guían al usuario hacia una nueva serie de consultas, navegación y refinamiento.

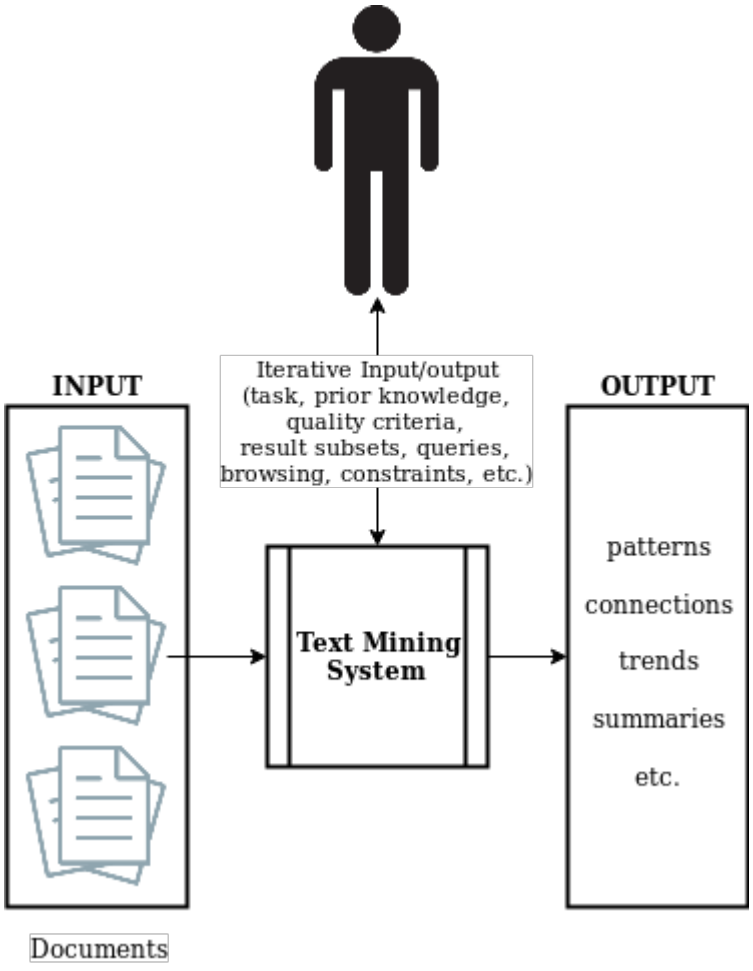


Figura 5: Modelo simple de entrada/salida para sistema de minería de texto.

Desde un aspecto práctico, los sistemas de TM siguen el mismo modelo que el de los de DM y, por lo tanto, pueden dividirse en cuatro etapas principales: (a) tareas de preprocesamiento, (b) operaciones de *minería de texto*, (c) visualización de los resultados obtenidos, y (d) refinamiento de las técnicas empleadas.

A continuación, se hará una breve mención sobre las tareas que involucra el preprocesamiento de texto. Dichas tareas permiten limpiary transformar los documentos para representarlos adecuadamente. La representación a utilizar dependerá de las tareas de text and datamining (TDM) que sean aplicadas posteriormente.

2.7. Recolección y selección

Cuando se trata de sistemas de TM, antes que todo deben conseguirse los documentos que conformarán su entrada de datos. Las tres preguntas clave son las siguientes: cuáles documentos se van a procesar,cómo puede accederse a ellos y cuáles herramientas se pueden utilizar para tal fin.

Como se mencionó antes, en cualquier sistema de TM, el recurso principal es el corpus de texto. Existen corpus diseñados para el desarrollo de este tipo de sistemas, muchos de los cuales pueden accederse importando librerías específicas de determinados lenguajes de programación. Tal es el caso de *Python* que a través del paquete *nltk.corpus* pueden utilizarse funciones específicas para leer archivos devarios corpus en una variedad de formatos.

La mayoría de los sistemas presentados en la literatura utilizan colecciones conocidas como las DUC (NIST, 2002) para evaluar su desempeño. Este

corpus en particular contiene docenas de textos individuales. Tanto esta colección como tantas otras vienen preparadas para ser utilizadas en la realización de determinados experimentos de prueba. A pesar de eso, están algo "controladas" y no representan la realidad que implica desarrollar este tipo de sistemas y ponerlos en práctica en casos concretos con documentos reales. Por esta razón, en esta tesis se decidió utilizar documentos no convencionales en las dos soluciones propuestas. En este caso, Internet fue la fuente principal paraacceder a ellos.

La recopilación automatizada de datos de Internet es casi tan antiguacomo la propia Internet. Esta recopilación puede conseguirse por medio de la utilización de APIs ofrecidas por los dueños de los datos o biena través de técnicas específicas que permiten hacerse de los datos por medio del procesamiento de las páginas de los sitios web.

Si bien cada vez son más las aplicaciones que ofrecen APIs paraacceder a sus datos (como por ejemplo Twitter y Facebook), estas a veces ponen restricciones permitiendo acceder únicamente a una parte de la información total. Por esta razón se acostumbra a conseguir dichos datos a través de mecanismos que combinan *web scraping* con un *web crawler*.

Su combinación permite interactuar con un sitio web de forma automatizada solicitando información al servidor web y manipulandola respuesta. Al fin y al cabo, se está navegando por Internet saltandode un enlace a otro de manera automática, recopilando información y almacenándola adecuadamente para su posterior uso.

Al desarrollar este tipo de programas, se termina apreciando todas las pequeñas cosas que los navegadores hacen por nosotros (Mitchell, 2015). La

Web, sin su capa de formato HTML y estilo CSS resulta caótica al procesarla. El procedimiento habitual tiene tres pasos básicos:

(a) recuperar datos HTML a partir de una Uniform Resource Locator (URL), (b) analizar los datos para almacenarlos (c) y, opcionalmente, continuar con otra página repitiendo el proceso.

Este proceso puede verse complicado cuando los sitios web cargan parte del contenido de sus páginas utilizando programas *JavaScript*. En estos casos, el HTML disponible para su procesamiento es el devuelto por el servidor web inmediatamente después de haberlo solicitado y el contenido esperado no se carga hasta tanto el *Javascript* no se ejecute (lo que implica una o varias comunicaciones con el servidor).

Sin embargo, el código *Javascript* es ejecutado siempre que haya de por medio un navegador web. Tanto en programas que realicen *web scraping* como en un *web scraper* no existe tal navegador. En estos casos deben usarse recursos de programación específicos para automatizarla interacción con un navegador web como si la estuviera llevandoa cabo el usuario y así poder acceder a los contenidos que cargan dinámicamente por código. En el caso de *Python*, por ejemplo, se tienea *Selenium* WebDriver[4], que permite remotamente (desde el script) ejecutar un navegador web con o sin interfaz gráfica para manipular el recorrido por el sitio realizando los clicks necesarios y esperando la carga de contenidos específicos.

[4] https://pypi.org/project/selenium/

2.8. Preprocesamiento del corpus

Tanto la obtención de resúmenes automáticos como la solución a cualquier otro problema de TM, son procesos tan complejos que requieren ser divididos en módulos o etapas. La primera fue la captura de los documentos a ser tratados. La segunda de ellas tiene que ver conla representación del texto en algún formato concreto que pueda ser adecuado para los algoritmos que se aplicarán luego (Hernández Oralloet *al.*, 2004).

En esta etapa, un documento de texto plano se transforma en un objeto con características lingüísticas mínimas como palabras, oracioneso párrafos. Pero para conseguir esta representación del documento deben realizarse previamente un conjunto de tareas que constituyenel preprocesamiento básico en la mayoría de los sistemas de TM: eliminación de ruido, segmentación, tokenización, normalización y reducción de léxico del texto.

Si bien se describirán a continuación cada una de ellas por separado, la mejor manera de verlas es como un todo tal como se muestra enla figura 6 (Mayo, 2017). Es imprescindible tener un conocimiento amplio de las tareas que pueden realizarse en cada caso y los efectos que cada una de ellas provoca. Por suerte, existen paquetesespecíficos en la mayoría de los lenguajes de programación, creados para facilitar el desarrollo de cada una de las subetapas que implicael preprocesamiento de texto.

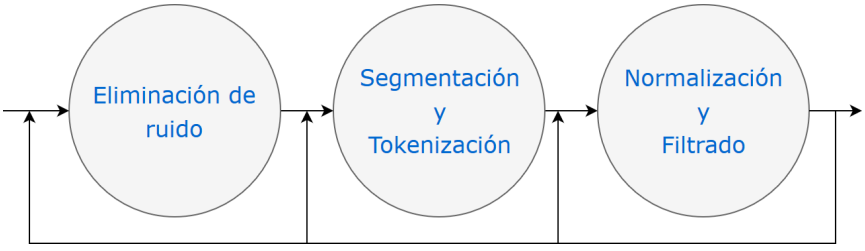


Figura 6: Relación entre tareas de preprocesamiento de texto.

2.8.1. Eliminación del ruido

Suponiendo que el corpus se obtuvo capturando páginas web de Internet, se puede asumir que el texto recopilado está envuelto de etiquetas HTML o XML. La eliminación del ruido ocurre siempre una vez recopilados los documentos. En ese momento, se tiene control sobre el proceso y puede lidiarse con semejante ruido: eliminar encabezados y pies de página, etiquetas HTML/XML y cualquier tipo de metadato o especie de marcado que pudiera haber en el documento, entre otras cosas. En este punto, son de suma utilidad las expresiones regulares,los analizadores de HTML y XML junto con bibliotecas específicas para extraer datos de archivos con esos formatos.

En el anexo B, se describe el proceso utilizado para obtener datos en bruto de PLOS Medicine y posteriormente construir un corpus. Allí se describirá el preprocesamiento del texto cuya presencia de ruido será tratada. No obstante, esta subetapa, a diferencia de las otras menos específicas, puede variar dependiendo de cómo se adquieran y ensamblen los datos.

Como se puede imaginar a partir de la figura 6, el preprocesamiento de texto no se trata de un proceso lineal cuyos pasos deben aplicarse exclusivamente en un orden específico. Sin embargo, pareciera que la eliminación de ruido debiera realizarse antes que las otras subetapas del preprocesamiento. Resulta natural que deba eliminarse el marcado de cualquier texto en formato JSON, por ejemplo, antes de proceder conla tokenización.

2.8.2. Segmentación

La segmentación es el proceso que divide el texto crudo de un documento en componentes significativos según el caso: capítulos, secciones, párrafos, oraciones, palabras e incluso sílabas. Las subtareas a realizar dependerán del objetivo del sistema a desarrollar. Noes lo mismo reconocer palabras clave de un texto que resumir automáticamente un documento.

La forma más común y básica de particionar el texto es el proceso de tokenización, que usualmente convierte una secuencia de caracteres en una secuencia de unidades básicas de procesamiento conocidas como "tokens". Estos pueden ser palabras, números, etc. Esto fue lo que el autor de esta tesis realizó en (Villa Monte *et al*., 2011) y (Lanzarini *et al*.,2011b) para identificar los términos más relevantes de un conjunto de correos electrónicos.

A continuación se muestra un ejemplo sencillo de partición en palabras de una cadena de caracteres:

«smoking causes lung cancer by damaging cells»

⇓

«smoking» «causes» «lung» «cancer» «by» «damaging» «cells»

Desde un punto de vista visual, un *token* puede definirse como la secuencia de caracteres de un texto que es tratada como una unidad indivisible para su procesamiento posterior. Generalmente, se aceptan como tokens todas las palabras delimitadas a la izquierda y a la derecha

por un espacio en blanco una vez que los signos de puntuación han sido removidos. Cada mención de una misma palabra en un documento se trata como un token separado. Sin embargo, la tokenización puede generar como tokens n-gramas de cualquier longitud si fuera necesario.

Un n-grama es una subsecuencia contigua de n elementos (palabras o caracteres) construidos a partir de una secuencia dada (una oración, por ejemplo). Pueden construirse a través de una "ventana deslizante" de longitud n. Las secuencias formadas por dos elementos se conocen como bigramas. Las formadas por tres, trigramas, y así sucesivamente. Debe tenerse en cuenta que el número de n-gramas que resultan de un texto es mayor que el número de palabras. A modo de ejemplo, a continuación se muestra el resultado de dividir en n-gramas la cadena de caracteres del ejemplo anterior de dos maneras distintas:

bigramas de palabras	«smoking causes» «causes lung» «lung cancer» «cancer by» «by damaging» «damaging cells»
trigramas de caracteres	«smo» «mok» «oki» «kin» «ing» «ng » «g c» « ca» «cau» «aus» «use» «ses» «es» «s l» « lu» «lun»…

Si bien la tokenización parece una tarea sencilla, debe realizarse cuidadosamente. Durante esta etapa se producen dificultades al reconocer los límites de las palabras. Los tokens son utilizados en tareas posteriores y los límites de las palabras a veces son ambiguos debido a particularidades de los distintos idiomas. No realizar correctamente el reconocimiento de tokens puede condicionar los resultados finales. Algunos de los inconvenientes son los siguientes:

- faltas de ortografía (olvidar el espacio en blanco luego de un signo de puntuación);
- excepciones en el uso de signos de puntuación como separadores (el punto y la coma para delimitar decimales, el punto en acrónimosy abreviaturas, los dos puntos al indicar horarios, la barra utilizadasen fechas, etc.),
- el uso de guiones en números de teléfono o en palabras compuestas;
- las mayúsculas que indican el principio de una oración o un nombrepropio;
- las direcciones de correo electrónico, las URLs, las citasbibliográficas, etc.,
- expresiones multi-palabras a ser tratadas como un solo token («América Latina», «a priori», «por lo tanto», etc.).

Estos son solo algunos ejemplos que requieren especial atención y deben considerarse si quiere tenerse una mejor representación de los documentos. Casos como los de «América Latina» pueden resolverse a través del reconocimiento de entidades nombradas o incluso casos máscomplejos como el siguiente:

> «***Washington*** *was born to a prosperous family of slaveholding planters in colonial Virginia.»*
>
> «***Washington*** *is one of the wealthiest and most liberally progressive states in the country.»*

Esta subtarea consiste en la ubicación de secuencias de palabras dentro de un texto que refieren a una entidad específica del mundoreal mediante un nombre propio. El resultado de este proceso es una clasificación de las entidades nombradas en categorías tales como el nombre de una persona, una ubicación, una organización, etc. También pueden incluirse otras construcciones textuales que no son estrictamente entidades nombradas, como fechas, números identificatorios, valores monetarios, etc. En Jurafsky and Martin (2018),se detalla una lista completa de los tipos y ejemplos de entidades nombradas que pueden encontrarse.

Dado que el proceso general de tokenización debe ejecutarse antes derealizar cualquier otro tipo de procesamiento del texto, es importante que sea un proceso rápido y preciso. Para eso, se utilizan algoritmos determinísticos basados en expresiones regulares que se compilan en autómatas de estado finito muy eficientes. Sin embargo, si se utilizan directamente las expresiones regulares para tokenizar el texto, se consigue mayor control sobre el proceso, pero llevaría mucho tiempoprogramar todos los casos uno por uno.

Existen varias librerías ya desarrolladas que resuelven la tokenización dependiendo del lenguaje. Lo que mejor resulta es combinar dichas librerías con la aplicación de ciertas expresiones regulares que permitan manejar casos particulares como las URLs, las direcciones de correo electrónico, etc. Como se verá a continuación, las expresiones regulares permiten buscar un patrón en el texto y devolverla parte del mismo que coincide con dicho patrón (Friedl, 2002).

Expresiones regulares

Las expresiones regulares (REs por sus siglas en inglés) datan de los años 70, cuando se las comenzó a utilizar en sistemas UNIX. Hoy son uno de los recursos imprescindibles de cualquier lenguaje de programación. Tienen una sintaxis específica y no poseen un ambiente de desarrollo propio. Por esta razón, se ofrecen en línea sitios en los cuales pueden desarrollarse expresiones regulares y testearse sobre casos de prueba. Los hay algunos bastante completos como *RegExr* [5] o más acotados como *pythex*[6].

Las REs tienen una curva de aprendizaje pronunciada. Pueden resultar difíciles de dominar y complejas de entender si no se escriben con cuidado. Hay quienes afirman que cuando se piensa la soluciónde un problema con expresiones regulares, se consigue tener dos problemas por resolver en vez de uno.

El uso de expresiones regulares es más común de lo que parece. Cualquier programador experimentado ha usado algún tipo de expresión regular. Por ejemplo, al usar el * o el signo de interrogación ? para encontrar archivos.

Una expresión regular es un patrón de texto que consta de caracterescomunes (por ejemplo, letras de la A a la Z o números del 0 al 9) y caracteres especiales conocidos como metacaracteres. Así, por ejemplo, el patrón «journal.pmed.*.xml» permite capturar todo lo que comience con "journal.pmed", siga con cero, uno o más caracteres de cualquier

[5] https://regexr.com/
[6] https://pythex.org/

valor, y termine con ".xml". Esto podría ser útil si se quiere recuperar ciertos archivos dentro de un directorio.

Las expresiones regulares son mucho más complejas que el patrón anterior y, a su vez, más poderosas. Su empleo está normalmente asociado, aunque no solamente, a operaciones de sustitución. Una expresión regular define la forma que debe tener una cadena de texto ypermite entre otras cosas (López and Romero, 2014):

- Comprobar si una entrada respeta un patrón dado. Buscar la presencia de un patrón en un pedazo de texto.Extraer porciones específicas de un texto.
- Reemplazar porciones de texto.
- Dividir un texto en partes más pequeñas.

Así, a través de expresiones regulares, el programador puede realizar muchas de las tareas que han sido mencionadas a lo largodel capítulo. Por ejemplo, puede definirse una expresión regular para detectar las URLs que generalmente traen problemas al tokenizar el texto por el uso mayormente del punto. De la misma manera pueden definirse expresiones regulares para detectar números, fechas, citasbibliográficas, etc.

Durante el preprocesamiento del texto, suele ser necesario reemplazar ciertas expresiones o incluso eliminarlas para podersegmentar correctamente el texto. Este es el caso de expresiones en

inglés como “et al.”, “i.e.” o “e.g.”, las cuales pueden detectarse con expresiones regulares.

Otro de los usos está relacionado con la búsqueda de ciertos términosen los textos o la detección de expresiones que involucran más de una palabra formando una cierta estructura, como se verá en el capítulo 4.

Otra utilización de las expresiones regulares se hace presente durantela captura de documentos. En este caso, para automatizar el recorrido sobre sitios web permitiendo la descarga de ciertos contenidos es habitual buscar dentro del HTML de sus páginas determinados enlaces o ciertos componentes que contienen lo que se busca.

Existen otros casos en los cuales las expresiones regulares pueden facilitar la tarea de un operador de un sistema. En el ámbito judicial de la provincia de Buenos Aires, la carga de información contextual esdeterminante para alcanzar un expediente electrónico (Maestropiedra, 2018). Estos datos se desprenden de los propios documentos judiciales que aportan las partes. A pesar de ello, la tarea de carga la lleva a cabo un agente judicial en forma manual y a través de expresiones regulares podría detectarse el contenido de acuerdo a los metadatos que son necesarios cargar.

2.8.3. Normalización

Una vez tokenizado el texto, lo siguiente que debe realizarse es su normalización. El objetivo radica en reducir las distintas formas deinflección derivadas de una palabra o variaciones del token a una formacomún. Por razones obvias propias del lenguaje, en los documentos se utilizan diferentes formas de una “palabra”. Por ejemplo, «investiga», «investigan» e

«investigamos» son todas conjugaciones del verbo «investigar». Además, «investigable», «investigador» e «investigación» también son palabras relacionadas con dicho verbo. Incluso, se utilizan en mayúsculas si se la utiliza en un nombre propio como "Instituto de Investigación en Informática".

El uso de la mayúscula en algunos casos puede definir el significado de un término. En el caso del inglés, por ejemplo, "Bob" puede haber sido usado como verbo al comienzo de una oración o para referirse a una persona por su nombre.

Otra manera de normalizar el texto es por medio del truncado y la lematización de palabras.

El truncado, también llamado *stemming*, es un método a través del cual se reduce una palabra a su correspondiente raíz morfológica (o stem en inglés). Permite reagrupar palabras distintas que comiencen con una misma secuencia de caracteres, como por ejemplo, "bibliotecas"y "biliotec*ario*" las cuales se reducen a "bibliotec". Aunque hay varios algoritmos de *stemming*, el más común y utilizado es el de Porter (Porter,1980).

En cambio, la lematización reemplaza cada 1-grama de palabra por su correspondiente lema o entrada en el diccionario. El lema de una palabra es su forma base, es decir, aquella palabra que representa por convenio a todas las que derivan de ella. Por ejemplo, "quisiera" y "querría" son derivaciones del verbo "querer". En formas verbales busca el infinitivo y en sustantivos su forma singular. Pero, a diferencia del *stemming*, la lematización requiere el uso de rotuladores gramaticales como lo es *POS Tagger*[7]. Por ejemplo, en el

[7] https://nlp.stanford.edu/software/tagger.shtml

idioma inglés, "saw" puede convertirse en "see" o quedar como está según si se lo reconoce como sustantivo o verbo.

El *stemming* es considerado, por algunos autores, como una especie de lematización básica ya que resulta ser más sencilla y económica que la obtención de lemas. El *stem* de una palabra intenta representar palabras con significado similar a partir de la poda de sufijos (para el inglés por ejemplo la "s" en los sustantivos, la "ing" en los verbos, etc.).

Otra opción algo más compleja que el *stemming* o la lematización consiste en unificar las palabras que se consideren sinónimos. Por ejemplo, los verbos "comprobar" y "verificar" son utilizados por igualla mayoría de las veces. Esto requiere de información adicional que permita identificar el grado de sinonimia entre palabras según su sentido. Este tipo de cosas se consigue utilizando recursos adicionales como *WordNet* [8]. Tanto *WordNet* como también *POS Tagger* forman parte de la solución propuesta en el capítulo 4.

2.8.4. Filtrado

Esta tarea tiene por objetivo remover los tokens de determinado tipoo aquellos que no cumplan ciertas condiciones, tal como se explicará a continuación.

[8] https://wordnet.princeton.edu/

La primera manera de filtrar tokens se basa en la existencia de dos clases de palabras. Por un lado, las palabras que por sí solas tienen un significado y, por el otro, las que poseen escasa o nula información de contenido. En el primer caso, generalmente se incluyen los sustantivos y verbos pero, en algunos casos, también los adjetivos y adverbios.En el segundo caso, se incluyen todas las palabras consideradas "no útiles", como artículos, pronombres, preposiciones y conjunciones. Incluso, pueden llegar a incluirse algunas otras palabras comunes que generalmente no contienen gran cantidad de información por utilizarsecon mucha frecuencia.

Para este tipo de filtro suele aplicarse una *stoplist* formada por las palabras de paro o *stop-words* en inglés. Esta lista indica todas las palabras que serán removidas. Dichas palabras dependen del idioma y, en principio, deberían ser independientes del dominio. Existenlistas ya preparadas para cada lenguaje, aunque suelen construirse manualmente incluyendo palabras específicas que no se desean mantener.

Normalmente, las *stop-words* representan entre el 20 % y el 30 % del total de palabras contenidas en un texto típico en inglés (Torres Moreno,2014). Sin embargo, no toda palabra de paro debe ser borrada.En algunos casos, cuando se quiere capturar estilo de redacción o analizar respuestas libres de encuestas, por ejemplo, los adverbios, las negaciones y hasta los adjetivos suelen ser extremadamente importantes y por eso se los debe conservar (Bécue-Bertaut, 2010).

El filtrado también puede realizarse removiendo palabras demasiado frecuentes o utilizando umbrales de frecuencia mínimos. En general,se seleccionan aquellas palabras cuya frecuencia es superior a cierto umbral. También puede determinarse *a priori* el número de palabras distintas a conservar. Existen muchas estrategias en este aspecto.

2.9. Resúmenes de texto

Como ya se mencionó anteriormente, la creciente cantidad de datos textuales disponibles genera la necesidad de avanzar en el diseño de algoritmos que aprendan patrones interesantes a partir de los datos. Los resúmenes de documentos son uno de los temas de investigación centrales en este sentido.

En cuanto a la definición de resumen, la misma varía de un autor a otro según el término utilizado para referirse a este. Si bien parece haberun acuerdo en definir al resumen como la representación reducida y objetiva de los contenidos de un documento, se debe distinguir entre distintos conceptos asociados al resumen. El NISO (2015) ha publicado una guía para preparar resúmenes en la cual distingue de forma clara los significados de *abstract*, *extract*, *structured abstract* y *summary*.

Hace algunos años, leer un libro saltándose párrafos, páginas o incluso secciones completas era algo menos común. Hoy lo es mucho más por la abundante información textual disponible. Hoy son pocoslos que leen libros completos, noticias, revistas, artículos, etc.

Tanto la *Real Academia Española* como el *Diccionario de Cambridge* coinciden en que el resumen es la acción y efecto de obtener una serie dedeclaraciones

breves, claras y precisas que indican las ideas principalesy esenciales sobre algo.

La generación automática de resúmenes de texto es el proceso mediante el cual se crea utilizando una computadora una "versión reducida" de uno o más documentos con el contenido relevante (Torres Moreno, 2014). Omitir oraciones o incluso párrafos completos sin sufrir pérdida de información es algo difícil de conseguir en forma automática.

Disponer de resúmenes automáticos ofrece muchos beneficios. Un resumen, por ejemplo, ayuda al lector de un documento a decidir si debeconsultar el texto completo. En el caso de un bibliotecario, a través de su lectura consigue una visión general del documento que lo ayudará a determinar los términos de indexación que faciliten su acceso.

Un documento es todo elemento susceptible de ser resumido. Esto no solo aplica a materiales escritos como un libro o una revista, sino también a aquellos materiales que no posean texto, como un video o una imagen. Sin embargo, los resúmenes de documentos no textuales quedan fuera del alcance de esta tesis y al mencionar "resumen" no se hará referencia a ellos.

Entonces, un documento de texto es una unidad de datos textualque usualmente, aunque no necesariamente, se corresponde con algún documento del mundo real. Pueden resumirse informes, monografías, reportes, e-mails, libros, comunicados, historias clínicas, trabajos

científicos, cartas, expedientes, entre otros. En todos los casos, la lecturadel resumen facilita la lectura del documento original. Su extensión jamás iguala ni mucho menos supera la del documento original.

Solo teniendo en cuenta el ámbito científico se tiene disponible una inmensa cantidad de documentos de texto. La cantidad total de documentos es inimaginable. En los últimos años, la difusión deinformación médica a través de Internet creó la necesidad de desarrollar técnicas capaces de descubrir, acceder y compartir el conocimiento de literatura médica (Li and Wu, 2006). Esta es una de las razones por las cuales en esta tesis se decidió utilizar documentos relacionados con la medicina.

Si bien entre los enfoques existentes para producir resúmenes con documentos biomédicos el enfoque extractivo es uno de los más utilizados (Mishra *et al*., 2014), en esta tesis se propone no solo abordarla construcción del resumen por extracción sino también desde una perspectiva algo diferente.

2.9.1. Tipos de resúmenes automáticos

A pesar de haberse comenzado a investigar hace muchos años, la obtención automática de resúmenes no ha dejado de ser un tema relevante. Desde entonces ha recibido contribuciones científicas de todotipo (Spärck Jones, 2007) y lo continúa haciendo hoy en día (Gambhir and Gupta, 2017).

Existen diferentes enfoques para generar automáticamente resúmenes. Estos están relacionados con varios aspectos del resumen:

si fue utilizado un único documento o varios, si se aplicó un umbral para controlar su tamaño e incluso si intervino el ser humano durante el proceso. En Lloret y Palomar (2012) se muestran en una tabla los tipos de resumen que existen. Pese a que son varios, el más importantees el que establece dos tipos principales de resúmenes automáticos: losextractivos y los abstractivos.

Los resúmenes extractivos están formados por “partes” del documento que fueron seleccionadas apropiadamente. Los resúmenes abstractivos, por otro lado, están formados por las “ideas” desarrolladasen el documento, sin hacer uso de las frases exactamente como aparecenen el documento original. En Hahn y Mani (2000) se detallan claramente las diferencias entre ambos enfoques.

En la literatura, la mayor parte de la atención se ha puesto en seleccionar del documento, mientras que generar un nuevo texto ha recibido menos atención de alguna manera (Saggion and Poibeau, 2013). El extractivo es el enfoque más utilizado, ya que no tiene que reescribir el texto ni tampoco tiene que garantizar la coherencia narrativa de lo seleccionado (Gupta and Lehal, 2010). Este tipode resumen ofrece tres ventajas: (1) el tamaño del resumen puedecontrolarse, (2) el contenido del resumen se obtiene con precisión, y (3) puede ubicarse fácilmente en el documento fuente.

Un resumen extractivo está formado por un conjunto de porciones de texto (desde palabras sueltas hasta párrafos enteros) literalmente copiadas de la entrada (Mani, 2001a), a las que llamaremos a partir de ahora “sentencias”, haciendo referencia a oraciones únicamente.

Este enfoque extrae partes de un documento sin requerir un análisis semántico complejo (Hahn and Mani, 2000). Sin embargo, resulta indispensable asignar una puntuación a cada parte del documento. Esta puntuación permite ordenar las partes de un documento de mayor a menor en una lista cuyas primeras posiciones son las más relevantes (Edmundson and Wyllys, 1961). En la sección 2.10 se describirán las métricas más conocidas para ponderar las partes de un documento.

En general, obtener un resumen extractivo puede considerarse un problema de clasificación de dos clases. A cada parte del documento se la etiqueta como "correcta" si forma parte del resumen, o "incorrecta"si no está incluida en él (Neto *et al.*, 2002). Trabajos recientes consideran la tarea de producir resúmenes extractivos como un problema de optimización, donde una o más funciones objetivo se formulan para seleccionar las "mejores" oraciones del documento que formarán parte del resumen (Vázquez *et al.*, 2018). Sin embargo, los documentos en este tipo de trabajo se caracterizan por un conjunto de métricas definido a priori, y su selección no forma parte del proceso de optimización, como en Meena y Gopalani (2015), por ejemplo. Esta es la característica clave del método desarrollado en el capítulo 3.

En cambio, para construir un resumen por abstracción se necesitan recursos de conocimiento lingüístico específicos tales como ontologías, tesauros y diccionarios para comprender el contenido (Mani, 2001b). Este enfoque resulta más complejo de llevar a cabo. Generalmente, utiliza algunas pocas tareas de preprocesamiento que no "destruyen" (en el sentido de transformar demasiado) el documento, ya que las palabras y su contexto resultan fundamentales. Este es el enfoque seguido en el capítulo 4 para la construcción de un grafo causal apartir del texto. Una vez preprocesado el documento, se crea una representación intermedia que lejos está de ser el documento original.

2.10. Representación de documentos

En general, la tarea de resumir automáticamente se puede dividir en dos grandes fases: (a) Construcción de la representación del texto; (b) Generación del resumen, que puede incluir la extracción de oraciones existentes o la

construcción de nuevas oraciones. La primera de ellas comienza con las tareas de preprocesamiento detalladas en la sección 2.8 y termina de explicarse en esta sección. Con relación a la segunda etapa, los siguientes dos capítulos presentarán una estrategia cada uno para resumir texto: el capítulo 4, siguiendo un enfoque más bien abstractivo; y el 3, uno extractivo.

La representación de los documentos depende del problema a resolver. No es lo mismo querer extraer palabras clave, clasificar documentos, resumirlos o incluso analizar sentimientos. Varios autores consideran distintas clasificaciones para los tipos de características quepueden utilizarse.

En general, pueden verse como estáticas o dinámicas. Las características estáticas son eligidas antes del procesamiento de los documentos. Consideran al documento como una serie de caracteres, palabras o incluso oraciones a partir de la cual se extraen estadísticas como el número total de caracteres, el número de letras mayúsculas,el tamaño de las palabras, la riqueza del vocabulario, la frecuencia de determinados signos de puntuación, el uso de la voz pasiva, el número total de oraciones, la longitud promedio de los párrafos, la presencia de ciertas palabras clave, etc. Las dinámicas, en cambio, derivan automáticamente del procesamiento de los documentos, por lo que no se puede definir *a priori* cuáles serán exactamente. Variarán de acuerdo a la colección de documentos considerada. Las características dinámicas más comunes son las palabras que ocurren en la colección de documentos y que no hayan sido filtradas. Incluso, también suele utilizarse los n-gramas de caracteres y de palabras de variada longitud.

Este último tipo de característica da lugar a la construcción de diferentes representaciones de texto. Una de ellas, quizás la más utilizada, es la representación de bolsa de palabras (BoW por sus siglas en inglés). En dicha

representación, cada documento se representa como un vector cuya dimensión es igual al número de palabras. Cada palabra constituye una componente del vector y representa una característica, la cual puede ser booleana (si la palabra aparece o no en el documento) o numérica (igual al número de apariciones de la palabra en el documento).

La figura 7 muestra la representación de bolsa de palabras utilizandopesos booleanos para los siguientes tres supuestos documentos:

«D1: Trabajo mucho en mi nuevo trabajo.»
«D2: Me gusta mucho el trabajo que hago.»
«D3: Me lleva minutos llegar al trabajo.»

	al	el	en	gusta	hago	llegar	lleva	me	mi	minutos	mucho	nuevo	que	trabajo
D1	0	0	1	0	0	0	0	0	1	0	1	1	0	1
D2	0	1	0	1	1	0	0	1	0	0	1	0	1	1
D3	1	0	0	0	0	1	1	1	0	1	0	0	0	1

Figura 7: Representación de bolsa de palabras utilizando pesos booleanos.

Si se utilizara la frecuencia de los términos para los pesos, larepresentación de los documentos 2 y 3 sería la misma. En cambio, en el documento 1, el término «trabajo» ocurre dos veces, debiendo valer 2 sucomponente. Si bien en ambos casos fueron utilizadas palabras, puedenutilizarse bi-gramas de palabras o tri-gramas de caracteres por ejemplo.

Por otro lado, los objetos o instancias de la representación utilizada corresponden a cada uno de los documentos. En el ejemplo, cada documento está compuesto por una única oración cada uno. Si se tratarade documentos más extensos, cada uno de ellos podría representarse por separado y cada instancia correspondería a una oración del documento.

Como se ha visto en la figura 7, los documentos son representados por una lista de valores que corresponden a los pesos de cada término del léxico del documento. Interpretando esta lista como un vector de n componentes, tantos como términos distintos tenga el documento, cada documento se puede considerar un vector en un espacio n-dimensional.

El *modelo de espacio vectorial* (VSM por sus siglas en inglés) es una de las variantes surgidas del Information Retrieval (IR) para recuperación de documentos (Peña *et al.*, 2002). Desarrollado por Salton (1971) para el sistema SMART, fue pionero en muchos de los conceptos usados en los motores de búsqueda modernos (Olivas, 2011). VSM representa cada documento como un punto en un espacio (Salton, 1971). De la misma forma que un punto en el plano (definido por los valores de sus coordenadas) puede considerarse el vector que va desde el origen hasta el citado punto, con los documentos sucede lo mismo (véase figura 8). Así, documentos semejantes entre sí están ubicados cerca en el espacio n-dimensional.

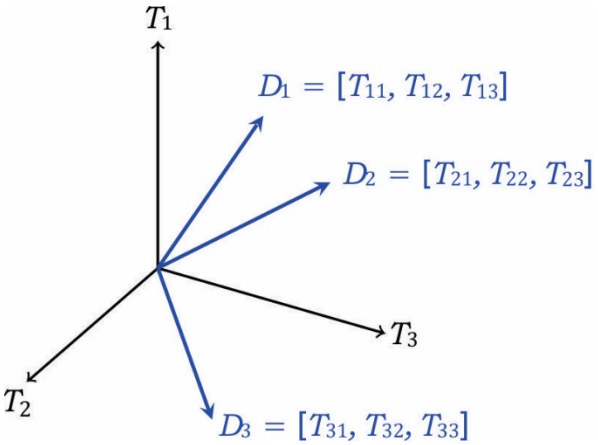


Figura 8: Representación del Modelo de Espacio Vectorial para tres documentos de tres términos.

En IR se cree que BoW permite estimar la relevancia de un documento respecto a una consulta, representando tanto los documentos como las

consultas como bolsa de palabras (Salton *et al*., 1983). Esta hipótesis se basa en la creencia que el vector fila en la matriz "documento-término" de la figura 9 captura, en alguna medida, aquello de lo que trata el documento. Si dos documentos tienen vectores filas similares, estos tendrán algún aspecto de su significado similar.

Como se vio anteriormente, en BoW cada documento es representado como una bolsa de palabras en la cual el orden noes importante. Cada bolsa es implementada como un vector con la frecuencia de ocurrencia en el documento de cada uno de los términos

	t_1	t_2	$\cdots$	t_k	$\cdots$	t_m
d_1						
d_2						
$\vdots$ d_j $\vdots$				w_{jk}		
d_n						

Figura 9: Matriz "documento-término" correspondiente al VSM. Las filas representan cada uno de los documentos y las columnas los términos diferentes que ocurren en ellos. w_{jk} es el peso del término k para el j-ésimo documento.

del vocabulario. Se le llama "vocabulario" o léxico al conjunto de términos diferentes que surge de la colección de documentos. Así, lasbolsas de todos los documentos de la colección constituyen una matriz "documento-término" de dimensión $n \times m$, siendo n el número de documentos en la colección y m la cantidad de términos que ocurrenen esos documentos.

Como se muestra en la figura 9, la importancia de un término está determinada por su peso w_{jk}. Generalmente, su peso lo determina su frecuencia. Las frecuencias de los términos ayudan a describir el contenido del documento (Sparck Jones, 1972).

Si se utiliza TF en la ponderación de los términos, el peso se incrementa proporcionalmente al número de veces que aparece el término en el documento. La forma más simple de usar la frecuencia como indicador de importancia en la entrada de datos es la probabilidad de aparición del término t_k (Nenkova and McKeown, 2012), calculada como el cociente entre el número de ocurrencias del término t_k en el documento d_j y T, el total de términos del documento:

$$tf\,(t_k, d_j) = \frac{(t_k, d_j)}{T} \qquad (1)$$

siendo T menor que m (cantidad total de términos diferentes que ocurren en el corpus D de n documentos) o a lo sumo igual.

En cambio, utilizando TF-IDF, la importancia general de un término se decrementa proporcionalmente a sus ocurrencias en la colección entera. La frecuencia de documento inversa se basa en que términos comunes no son buenos para discriminar documentos (Salton and Buckley, 1988) y se calcula de la siguiente manera:

$$tf - idf\,(t_k, d_j, D) = tf\,(t_k, d_j) \cdot \log \frac{n}{df\,(t_k, D)} \qquad (2)$$

donde $df\,(t_k, D)$ es el número de documentos del corpus D que contienen el término t_k y será siempre menor o igual que n.

A lo largo de esta sección se habló de palabras y términos casi indistintamente, pero no son lo mismo. En muchos casos, las palabras no son usadas directamente para la representación de documentos, sino los "términos" que se obtienen a partir de ellas. Una palabra es una unidad léxica que tiene un significado fijo y una categoría gramatical.En cambio, al usar "término" se hace referencia a toda aquella entidad que constituye una unidad atómica de significado en un texto. Según las decisiones que haya tomado el programador del sistema durante la etapa de preprocesamiento, el término podrá ser una palabra, su stem, su lema, un n-grama de caracteres, un n-grama de palabras o incluso frases, entre otras alternativas.

Debe tenerse en cuenta que la vectorización de documentos producematrices que pueden llegar a ser inabordables por muchos de los algoritmos de aprendizaje automático existentes. Las matrices resultantes son muy voluminosas, dispersas y extremadamente ruidosas. Esto significa que la información relevante es solo una pequeña parte del volumen total de datos disponibles (Torres Moreno, 2014). Por esta razón, los mecanismos de filtrado y normalización vistosen la sección 2.8 son indispensables para reducir la complejidad del léxico y, por lo tanto, la dimensión de esta matriz. Una vez construidala matriz también suele reducirse su dimensionalidad utilizando diferentes estrategias como a*nálisis de componentes principales* o *análisis de correlación* (Lebart *et al*., 2000).

Existen muchas variantes de VSM que difieren en el tipo de matriz, término y ponderación utilizados. Estas variantes están fuera del alcance de esta tesis, pero en Jurafsky y Martin (2018) puede consultarse un detalle completo de cada una de ellas.

2.11. Ponderación de sentencias

En la sección anterior, se vio la representación de los documentos independientemente del problema a resolver. Sin embargo, al igual que sucede con cada acción que se realice durante el preprocesamiento, la representación concreta depende estrechamente del problema. No será la misma representación la que deba usarse si se quiere categorizar documentos por tema que si se lo quiere hacer por autor. Incluso, tampoco será la misma representación si se van a extraer palabras claves de un documento o si se lo piensa resumir.

En sus orígenes, VSM se propuso para representar únicamente documentos, pero rápidamente fue aplicado también a sentencias. Este modelo se ha utilizado en clasificación de documentos, filtrado de información y también resumen automático de texto. Varios algoritmos de resumen automático utilizan este modelo definido con m términosy n documentos, adaptándolo a un solo documento con m términos y p sentencias. No obstante, cuando de resúmenes extractivos se trata, la representación predominante es la matriz que establece la relación entre una sentencia y un conjunto de características particulares.

Estas características, tal como se adelantó en la sección 2.9.1, permiten clasificar el contenido de un documento a través del puntaje asignado a cada sentencia (Nenkova and McKeown, 2012). Generalmente la puntuación de una sentencia indica su importancia y esta se relaciona con la forma que tiene, las palabras que usa, su ubicación dentro del documento, etc. Cada una de ellas permite seleccionar las “mejores” sentencias para producir el resumen (aquellas con su valor más alto). Dicho resumen puede variar según la característica utilizada.

2.11.1. Enfoques clásicos más utilizados

En la literatura, existe abundante bibliografía relacionada con los resúmenes extractivos. En su mayoría, los trabajos relacionados utilizan métricas de *scoring* para caracterizar las partes de un documento y construir su representación, como es el caso de Litvak *et al*. (2010b).

Cada métrica analiza una característica determinada del documento y permite aplicar cierto criterio de clasificación a su contenido. A continuación, se hace un repaso de las más utilizadas.

Luhn (1958) desarrolló el primer algoritmo que obtuvo un resumen simple ponderando las oraciones a partir de la distribución de la frecuencia de palabras. En ese mismo año, Baxendale (1958) usó la posición de las oraciones. Ambos fueron quienes hicieron las primeras propuestas de indexación automática. Unos años después, basándose en el trabajo de Luhn, Edmundson (1969) propuso utilizar la presencia de ciertas palabras y las coincidencias con las palabras que conforman los títulos de un documento.

Muchos años después, se usó la frecuencia de los términos y la frecuencia de sentencia inversa (ISF por sus siglas en inglés), una adaptación de la medida TF-IDF usada en IR (Larocca Neto *et al*., 2000).

Al año siguiente, Gong and Liu (2001) propuso el uso de *análisis de semántica latente* (LSA por sus siglas en inglés) para resumir texto sin utilizar recursos léxicos como *WordNet*.

En 2004, se comenzaron a desarrollar modelos basados en grafos para la extracción de oraciones (Mihalcea, 2004) y también se presentó un método basado en la proporción de palabras clave presentes en las oraciones y en el documento (Fatma *et al*., 2004). Tres años después, Vanderwende *et al*.

(2007) puso en práctica la idea de usar la frecuenciade palabras promedio para la selección de oraciones.

Estas son solo algunas de las métricas que existen. La tabla 1 muestra detalladamente el conjunto de métricas que comúnmente se utiliza y

Tipo	Fórmula	Referencia
Posición	i i^{-1} $\max(i^{-1}, (p - i + 1)^{-1})$	(Baxendale, 1958)
Longitud	$\|terms(s_i)\|$ $\sum_{t \in s_i} \|\{char : char \in t\}\|$	(Nobata et al., 2001)
Keywords	$\frac{\|keywords(c)\|^2}{\|c\|}$ $\sum_{k \in keywords(s_i)} \text{tf}(k)$ $\frac{\|terms(s_i) \cap keywords(d_j)\|}{\|keywords(d_j)\|}$	(Luhn, 1958) (Edmundson, 1969) (Fatma et al., 2004)
Frecuencias	$\frac{\sum_{t \in s_i} \text{tf}(t)}{\|terms(s_i)\|}$ $\sum_{t \in s_i} \text{tf}(t) \cdot \text{isf}(t)$	(Vanderwende et al., 2007) (Larocca Neto et al., 2000)
Títulos	$\frac{\|terms(s_i) \cap terms(titles)\|}{\min(\|terms(s_i)\|, \|terms(titles)\|)}$ $\frac{\|terms(s_i) \cap terms(titles)\|}{\|terms(s_i) \cup terms(titles)\|}$ $\frac{s_i \times titles(s_i)}{\|s_i\| \times \|titles(s_i)\|}$	(Edmundson, 1969)
Cobertura	$\frac{\|terms(s_i) \cap terms(d_j - s_i)\|}{\min(\|terms(s_i)\|, \|terms(d_j - s_i)\|)}$ $\frac{\|terms(s_i) \cap terms(d_j - s_i)\|}{\|terms(d_j)\|}$ $\frac{s_i \times d_j - s_i}{\|s_i\| \times \|d_j - s_i\|}$	(Litvak et al., 2010b)

Tabla 1: Detalle de métricas que más se consideran en la literatura para representar documentos yobtener resúmenes extractivos.

menciona en la literatura. Por cada tipo de métrica se indica la manera de calcular cada una de ellas. En todos los casos, $d_j = \{s_1, s_2, \dots, s_p\}$ es el j-ésimo documento de la colección, compuesto por p sentencias a las que se le calcularán cada una de las métricas. s_i representa la i-ésima sentencia del documento d_j. i es un número entero entre $(1, p)$ asignadosecuencialmente a cada sentencia de principio a fin según la ubicación dentro del documento. $|\ |$ indica cardinalidad. $terms(s_i) = \{t : t \in s_i\}$. $keywords$ y $title$ devuelven una lista de términos correspondiente a las palabras clave y a la unión de todos los títulos del documento respectivamente.

Las métricas de posición miden la cercanía de la sentencia con el final, el principio y los extremos del documento.

Las de longitud se emplean para penalizar las sentencias del documento demasiado cortas. Miden su longitud en cantidad de términos o contando sus caracteres.

En el caso de la métrica de Luhn, se calcula sobre una secuenciade palabras (dentro de la sentencia) determinada por las palabras clave del documento. c es la secuencia más larga entre dos palabras clave dentro de una sentencia. En las siguientes dos métricas, una suma las frecuencias de todas las palabras clave del documento que se encuentren en la sentencia y la otra calcula la proporción de palabras clave del documento que la sentencia tiene.

En las otras métricas, $tf\ (t_k)$ es la frecuencia del término t_k calculada como indica la ecuación 1 e $isf\ (t_k)$ la frecuencia de sentencia inversa del término t_k calculada como:

$$isf\,(t_k) = log\,\left(\frac{p}{sf\,(t_k)}\right) \quad (3)$$

donde $sf\,(t_k)$ es el número de sentencias del documento d_j que incluyenel término t_k.

En cuanto a las métricas, tanto de título como de cobertura, cuentan la cantidad de términos comunes entre la sentencia s_i y otro texto. En el primer caso, corresponde a los títulos del documento y, en el segundo,al documento excluyendo la sentencia en cuestión. En ambos casos, para calcularlas se tienen tres variantes según la medida de similitudutilizada: *overlap*, *jaccard* o *coseno*. En el caso de esta última no se utilizan las cadenas de términos sino los vectores formados por el número de ocurrencias de cada término. Por ejemplo, para la i-ésima sentencia se conforma el vector $s_i = ((t_1, s_i), c(t_2, s_i), \dots, c(t_m, s_i))$.

Existen otras métricas basadas en grafos cuyos nodos representan las oraciones del documento y cuyos arcos las relaciones entre ellas (Litvak et al., 2010a). El peso del arco está determinado por la cantidad de palabras que tienen en común dos oraciones. Luego, aplicando determinados umbrales, algunos arcos son eliminados y las oraciones de los nodos con mayor cantidad de enlaces serán las más relevantes. El grafo simplemente es utilizado para ponderar cada sentencia y no se lo utiliza como representación del documento a partir de la cual extraerposteriormente un resumen, como se lo verá en el capítulo 4.

Para profundizar tanto en las métricas mencionadas como en otros métodos menos citados puede consultarse Torres Moreno (2014). Por

otro lado, en el anexo B se presenta un diseño de base de datos para almacenar de manera estructurada los documentos permitiendo calcular el conjunto de métricas utilizado.

2.11.2. Representación basada en métricas

Basándose en la representación de Salton *et al*. (1975) y en las métricasantes descritas, en la mayoría de los trabajos que tratan resúmenes extractivos, los documentos se modelan como vectores n-dimensionales utilizando un conjunto de características numéricas obtenidas de calcular n métricas.

En este tipo de representación se considera al documento como una serie de palabras en sentencias. Por lo tanto, a partir de un documento se genera una matriz S de p filas (sentencias) y n columnas (métricas). La vectorización de un documento lo transforma en un conjunto de p vectores "sentencia" de la forma $S_i = (s_{i1}, s_{i2}, \dots, s_{in})$. Esto produce la matriz "sentencia-métrica" de la ecuación 4.

$$S = \begin{bmatrix} s_{11} & s_{12} & \dots & s_{1n} \\ s_{21} & s_{22} & \dots & s_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{p1} & s_{p2} & \dots & s_{pn} \end{bmatrix} \quad (4)$$

Cada sentencia estará representada por su correspondiente vector S_i. Los vectores de características luego son utilizados para obtener resúmenes automáticos aplicando algoritmos más sofisticados (Nenkova and McKeown, 2012).

2.11.3. Selección de información relevante

Cuando el ser humano resume, trata de seleccionar la información relevante poniendo en juego una serie de criterios y estrategias derelevancia que dependen en algunos casos del tema en cuestión o del tipo de documento, pero por sobre todo de los objetivos del lector (Pinto *et al*., 2005). Diseñar un programa que seleccione frases representativas y significativas de los documentos automáticamenterequiere instrucciones precisas (Cremmins, 1996).

Para formar un resumen automático, se debe seleccionar la mejor combinación de oraciones. La preferencia por una sentencia está dada por la puntuación que se le asigne. Se trata de un valor entero positivo proporcional al grado de importancia estimado. Aquellas sentencias que reciban cero como puntaje serán interpretadas como irrelevantes, mientras que las que reciban los valores más altos serán las más significativas.

La puntuación se consigue, o bien a través de una métrica (y en ese caso se tienen tantos puntajes como métricas se hayan calculado), o bien combinando diferentes métricas para conseguir un único puntaje para la sentencia. En Oliveira *et al*. (2016), los autores evaluaron individualmente el desempeño de algunas técnicas de puntuación de oraciones y también aplicando diferentes estrategias de combinación.

En este caso, se espera que la combinación lineal de ellas represente elcriterio del ser humano al resumir. Por lo tanto, el problema a resolver consiste en hallar ciertos coeficientes tales que, aplicados a los valores de las métricas para cada sentencia, permitan aproximar un puntaje similaral que le pondría el humano y así construir un resumen lo más parecidoa la realidad.

2.12. Conclusiones

Obtener conocimiento nuevo a partir de la información almacenadano es una tarea sencilla y mucho menos si se trata de información no estructurada. El objetivo principal del TM es obtener nueva informacióna partir de fuentes de datos en formato texto. Constituye junto con el NLP y el ML entre otros, el proceso de KDT.

Se comenzó el capítulo introduciendo la problemática actual y contextualizando cada uno de los conceptos que implica el desarrollo de sistemas de TM. Fueron desarrolladas las etapas del KDT haciendo hincapié en las diferencias que tiene con el conocido KDD.

En particular, se explicaron las tareas que involucra el preprocesamiento de texto. Esta etapa depende del problema a resolver, del criterio del programador del sistema y/o de los expertos que intervienen en su desarrollo. Del conocimiento absoluto y la supervisión de cada una de las tareas que se realicen dependerá la representación de los documentos.

Se presentaron los dos tipos de resúmenes automáticos más importantes y las características de cada uno de ellos. De todos los enfoques que podían utilizarse para construir un resumen automático, la última parte del capítulo estuvo dedicada a explicar en detalle la representación vectorial que puede conseguirse a partir de los documentos a través de la caracterización de sus sentencias, utilizando un conjunto de métricas. Dicha representación será utilizada en el próximo capítulo y difiere completamente de la utilizada en el capítulo 4.

CAPÍTULO 3

Resumen utilizando una técnica de optimización mediante cúmulo de partículas

La generación de resúmenes automáticos de documentos continúa siendo una tarea difícil de resolver y más cuando el resumen a construir debe ser lo más parecido posible al resumen que el ser humano construiría. Esta tesis presenta un método alternativo que, utilizando la *optimización mediante cúmulo de partículas* (PSO por sus siglas en inglés), permite construir el resumen deseado.

El método genera automáticamente resúmenes extractivos de documentos al ponderar adecuadamente las características de puntuación de las sentencias. La particularidad del método aquí propuesto está en la identificación de aquellas características que están más cerca del criterio utilizado por el individuo al resumir y el descubrimiento de los coeficientes de ponderación correspondientes. Elmétodo propuesto combina una representación binaria y una continua de las partículas, haciendo uso de una variación original de dicha técnica.

Hallar mediante una técnica de optimización la mejor solución posible o la solución óptima de un problema concreto no siempre es una tarea sencilla. Todo dependerá de la complejidad del espacio de búsqueda. Para aplicar una técnica de optimización poblacional como PSO, se debe definir qué información almacenará cada individuo, cuál será la inicialización de la población y cuál será el desplazamiento que las partículas harán dentro del espacio de búsqueda, entre otras cuestiones. Este capítulo brindará todos los detalles del método propuesto aplicado a la obtención de resúmenes.

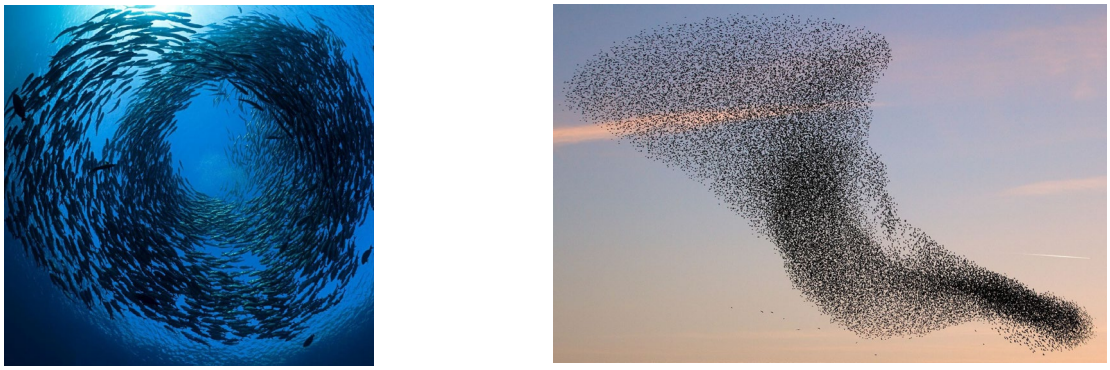
3.1. Técnicas de optimización

Cuando los problemas no son demasiado complejos, existen técnicas para hallar su solución, como lo es la búsqueda del óptimo utilizando información del gradiente. Sin embargo, a medida que la complejidad del espacio de búsqueda aumenta, el costo computacional de dichas técnicas se incrementa notablemente, haciendo su aplicación inviable. En estos casos, se aceptan soluciones alternativas encontradas en un tiempo razonable. Se trata de soluciones que se aproximan a la soluciónóptima y que cumplen con una cota de error establecida.

Cuando la solución exacta es difícil de obtener, las estrategiasde búsqueda aproximadas mediante técnicas de optimización han demostrado ser sumamente efectivas. Entre dichas técnicas se distinguen las basadas en una única solución y las que utilizan grupos de soluciones para aproximar el óptimo. En el primer caso,se perfecciona sucesivamente la solución actual. En el segundo, en cambio, un conjunto de soluciones denominado población es mejorado a través de procesos biológicos.

En biología, adaptación es el proceso realizado por un organismoque ha evolucionado durante un período de tiempo con el objetivode acomodarse a las condiciones de su entorno (Futuyma, 2006). La naturaleza ejemplifica a diario el éxito de los procesos adaptativos. Los seres humanos los han imitado en numerosos contextos y disciplinas con el único objetivo de "mejorar" su desempeño a la hora de resolver un problema. Por lo tanto, no es de extrañar que, explícita o implícitamente, se hayan utilizado ciertos comportamientos biológicos para la construcción de modelos matemáticos complejos y que se hayan usado muchas metáforas y términos procedentes de la genética, la etología, e incluso de la etnología o la psicología para justificar decisiones y/o acciones (Clerc and Kennedy, 2002).

La inteligencia de cúmulo (o SI por sus siglas en inglés) basa la búsqueda de la mejor solución en la "inteligencia colectiva". El objetivo es utilizar el conocimiento adquirido por los individuos de la población para mejorar el desempeño del grupo. Según el enfoque empleado, podrán utilizarse individuos existentes para generar nuevos individuos, como lo hacen los algoritmos genéticos, o permitir que los individuosse desplacen en el espacio de soluciones observando el comportamientode su entorno y el suyo propio. Esto último corresponde al SI que será desarrollado en este capítulo, por tratarse de la estrategia en la que se basa el método propuesto. La optimización realizada de esta forma



(a) Peces (b) Aves

Figura 10: Ejemplos de Inteligencia de Cúmulo en la naturaleza.

se denomina *optimización mediante cúmulo de partículas* (PSO por sus siglas en inglés).

Las siguientes secciones describen los tipos de esta técnica que fueron utilizados en el método que permite obtener una solución óptima aproximada a través de una población de tamaño fijo. Sus individuos tienen la capacidad de desplazarse por el espacio de soluciones continuo y discreto recordando su mejor desempeño y observando su entorno.

3.2. Optimización mediante cúmulo de partículas

El algoritmo metaheurístico basado en población conocido como PSO fue propuesto por Kennedy and Eberhart (1995). Al igual que otras técnicas de optimización, su funcionamiento tiene una inspiración biológica imitando el comportamiento social de algunos grupos de animales como las aves y los peces (véase la figura 10). Estos grupos utilizan la inteligencia colectiva para resolver distintas situaciones.

Este comportamiento no es exclusivo de los animales, ya que los seres humanos también lo utilizan. Por ejemplo, si un grupo de personas se encuentra atrapado en un incendio, tenderán a buscar una salida rápida. Quienes no puedan hallarla, observarán el comportamiento de los demás y seguirán a la mayoría aun sin saber si la direcciónelegida es la correcta. Como puede observarse, según este enfoque, el conocimiento actual de cada individuo se ve afectado por el entorno. Para más detalles sobre SI o los aspectos filosóficos de PSO se puede consultar Kennedy y Eberhart (2001).

En (Kennedy and Eberhart, 1995), la población está compuesta por un cúmulo de tamaño fijo de individuos. Cada individuo de la población, llamado "partícula", es una posible solución al problema que permanentemente busca mejorarse a través de un proceso de adaptación que tiene en cuenta tres factores esenciales: (I) conocimiento actual sobre su entorno (su capacidad para resolver el problema indicado por su valor de aptitud o fitness), (II) conocimiento histórico o experiencias anteriores (memoria o conocimiento cognitivo),y (III) conocimiento sobre individuos situados en su vecindario (cooperación o conocimiento social).

Los dos primeros hacen referencia a los aspectos culturales de los individuos del cúmulo y, el último, a su capacidad de observar al grupo e imitar comportamiento, brindándole un aspecto social. En su versión original, cada partícula está en continuo movimiento, explorando elespacio de búsqueda, y nunca muere. De esta forma, se consigue mejorar las soluciones evolucionando toda la población a lo largo de

Algoritmo 1 Pseudocódigo del algoritmo básico de PSO

```
inicializar las variables necesarias
crear la población de partículas
repeat
    ajustar el valor del factor de inercia
    for all partícula de la población do
    calcular su fitness
        if el fitness es mejor que el de su mejor partícula then
            actualizar su mejor partícula y guardar el fitness
        end if
    end for
    for all partícula de la población do
        recuperar la mejor partícula del vecindario
        actualizar la velocidad y modificar la posición de la partícula
    end for
until alcanzar la condición de terminación
return la solución de la mejor partícula de la población
```

las iteraciones del algoritmo. Los movimientos de las partículas se encuentran acotados dentro del espacio de búsqueda. Dicho espacio se encuentra definido de antemano y no se permite que las partículas se desplacen fuera de él. El algoritmo 1 describe el proceso de búsqueda yoptimización llevado a cabo por el PSO básico.

La figura 11 ilustra el movimiento de una partícula en el espaciode soluciones. A medida que las partículas pierden velocidad, la información referida a su mejor desempeño anterior y a la mejor solución del vecindario comienza a tomar importancia, provocandosu estabilización.

Con respecto a la mejor partícula del vecindario, si se trata de un PSOglobal, se tiene en cuenta toda la población. En cambio, si se trata de uno local, la mejor partícula se consigue del entorno inmediatamente próximo a la partícula. El tamaño de dicha vecindad influye en la velocidad de convergencia del algoritmo, así como en la diversidadde los individuos de la población. A mayor tamaño de vecindad, la convergencia del algoritmo es más rápida,

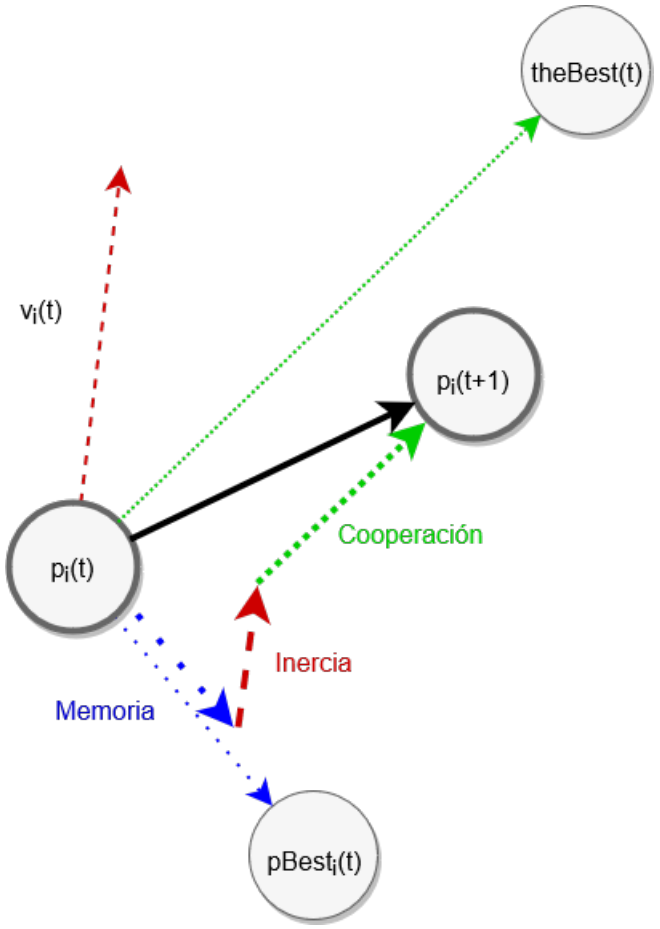


Figura 11: Movimiento de una partícula del instante t al instante $t + 1$ siendo $p(t)$ su posición actual, $p(t + 1)$ su siguiente posición, $v_i(t)$ la velocidad actual (inercia), $pBest_i(t)$ la mejor solución hallada hasta el momento (memoria) y $theBest(t)$ la mejor solución del vecindario (cooperación).

pero la diversidad de los individuos es menor. En las próximas secciones, a medida que el método propuesto sea desarrollado, se podrán entender el resto de los aspectos del algoritmo 1.

Desde su creación se han desarrollado diferentes versiones de esta conocida técnica de optimización (del Valle *et al.*, 2008). Originalmente, se definió para operar sobre un espacio de búsqueda continuo, por lo cual debían tenerse en cuenta consideraciones especiales para trabajar en espacios discretos. Por esta razón, años más tarde Kennedy and Eberhart (1997) definieron una versión binaria del método. Uno de los principales problemas que tuvo dicha versión fue la dificultad para cambiar de 0 a 1 y de 1 a 0 una vez

estabilizado. Esto impulsóel desarrollo de diferentes versiones binarias del PSO que buscaron mejorar su capacidad exploratoria.

3.3. Método propuesto basado en PSO

Usar PSO para generar un resumen extractivo que combine diferentes métricas de puntuación requiere utilizar ambos tipos de PSO mencionados anteriormente. Por un lado, se debe seleccionar el subconjunto de métricas a utilizar (parte discreta) y, por el otro, se debe establecer la relevancia de cada una de estas métricas (parte continua). Por este motivo, el método PSO que se usará aquí combina una versión binaria y una continua. Este principio se utilizó con anterioridad para encontrar reglas de clasificación a partir de información de instituciones financieras que permitieran mejorar la distribución de los créditos (véase el anexo A).

3.4. Algoritmo de optimización utilizado

Para moverse en un espacio n-dimensional, cada partícula p_i dela población está formada por: (a) un individuo binario $BinInd$ y su mejor individuo $BestBinInd$, ambos con el formato $BinInd_i$ = ($binInd_{i1}$, $binInd_{i2}$, …, $binInd_{in}$); (b) un individuo continuo $RealInd$ y su correspondiente mejor individuo $BestRealInd$, ambos con el formato
$RealInd_i$ = ($realInd_{i1}, realInd_{i2}$, …, $realInd_{in}$); (c) el valor de aptitud fit_i

correspondiente al individuo y el de su mejor individual $fitBestInd_i$;y (d) tres vectores de velocidad, $V1$, $V2$ y $V3$, todos con el formato $V1_i = (v1_{i1}, v1_{i2}, \ldots, v1_{in})$.

Como se puede ver, la partícula tiene una parte binaria y otra continua. Las velocidades $V1$ y $V2$ se combinan para determinar ladirección en la que la partícula se moverá en el espacio discreto, y $V3$ seusa para mover la partícula en el espacio continuo. $BinInd$ almacena la ubicación discreta de la partícula, y $BestBinInd$ almacena la ubicación de la mejor solución encontrada hasta el momento. Tanto $BinInd$ y $BestBinInd$ como $RealInd$ y $BestRealInd$ contienen la ubicación de lapartícula y la de la mejor solución encontrada en los espacios binarioy continuo respectivamente. fit es el valor de aptitud del individuo, y $fitBestInd$ es el valor correspondiente a la mejor solución encontrada poréste. En la sección 3.7, se describirá el proceso utilizado para calcular laaptitud de una partícula a partir de sus dos posiciones (una por cada espacio).

Luego, cada vez que la i-ésima partícula se mueve, su posición actualcambia de la siguiente manera.

Parte binaria

$$v1_i(t+1) = wBin.v1_{ij}(t) + \varphi 1.rand1_{ij}.(2.bestBinInd_{ij} - 1) + \varphi 2.rand2_{ij}.(2.theBestBinInd_{ij} - 1) \quad (5)$$

donde $wBin$ representa el factor de inercia, $Rand1$ y $Rand2$ son valores aleatorios con distribución uniforme en $(0, 1)$, $\varphi 1$ y $\varphi 2$ son valores constantes que indican la importancia asignada a las respectivas soluciones, $bestBinInd_{ij}$ y $theBestBinInd_{ij}$ corresponden al j-ésimo dígito de los vectores binarios $BestBinInd$ y $TheBestBinInd$ de la i-ésima partícula, y $TheBestBinInd$ representa la posición binaria de la partícula
con el mejor valor de aptitud dentro del entorno de la partícula p_i (sise trata de un PSO local) o de todo el cúmulo (si es global). Comose muestra en la ecuación 5, además de considerar la mejor solución
encontrada por la partícula, también se tiene en cuenta la posición de la mejor partícula vecina. Por lo tanto, el valor $theBestBinInd_{ij}$ corresponde al j-ésimo valor del vector $BinInd_k$ de la partícula p_k con un valor de aptitud $f\,it_k$ mayor que su aptitud ($f\,it_i$).

Cabe señalar que, como se discutió en Lanzarini *et al*. (2011a) y a diferencia del método binario descrito en Kennedy y Eberhart (1997), el movimiento del vector $V1_i$ en las direcciones correspondientes a la mejor solución encontrada por la partícula y la mejor de su vecindario no dependen de la posición actual de la partícula. Luego, cada elemento del vector de velocidad $V1_i$ se calcula con la ecuación 5 y se controla conla ecuación 6,

$$v1_{ij}(t) = \begin{cases} \delta 1_j & si\ v1_{ij}(t) \geq \delta 1_j \\ -\delta 1_j & si\ v1_{ij}(t) \leq -\delta 1_j \\ v1_{ij}(t) & si\ no \end{cases} \tag{6}$$

$$\delta 1_j = \frac{limit1_{j\ upper} - limit1_{j\ lower}}{2} \tag{7}$$

donde $v1_{ij} \in (-limit1_j, limit1_j)$ debido a los límites que mantienen los valores de las variables dentro del rango establecido. Luego, el vector
$V1$ se usa para actualizar los valores del vector de velocidad $V2$, como se muestra en la ecuación 8,

$$v2_i(t+1) = v2_{ij}(t) + v1_{ij}(t+1) \tag{8}$$

El vector $V2_i$ se controla de forma similar al vector $V1_i$ cambiando $limit1_{j\ upper}$ y $limit1_{j\ lower}$ por $limit2_{j\ upper}$ y $limit2_{j\ lower}$, respectivamente. Esto generará $\delta 2_j$ que se usará como en 6 para limitar los valores de $V2_i$. Luego, la nueva posición de la partícula se calcula usando la ecuación 10, aplicando la función sigmoide (ecuación 9),

$$sig(x) = \frac{1}{1 + e^{-x}} \tag{9}$$

$$binInd(t+1) = \begin{cases} 1 & si\ rand_{ij} < sig(v2_{ij}(t+1)) \\ 0 & si\ no \end{cases} \tag{10}$$

donde $rand_{ij}$ es un número aleatorio con distribución uniforme en (0, 1).

La aplicación de la función sigmoide en la ecuación 10 cambia radicalmente la forma en que se usa el vector de velocidad paraactualizar la posición de la partícula.

Parte continua

$$v3_i(t+1) = wReal.v3_{ij}(t) + \varphi 3.rand_3 ij.(bestBinInd_{ij} - realInd_{ij}) + \varphi 4.rand4_{ij}.(theBestRealInd_{ij} - realInd_{ij}) \quad (11)$$

siendo

$$realInd_i(t+1) = realInd_{ij}(t) + v3_{ij}(t+1) \quad (12)$$

donde, al igual que en la parte binaria, $wReal$ representa el factor de inercia, $Rand_3$ y $Rand_4$ son valores aleatorios con distribución uniforme en (0, 1), y $\varphi 3$ y $\varphi 4$ valores constantes que indican la importanciaasignada a las respectivas soluciones. En este caso, $TheBestRealInd$ corresponde al vector $RealInd$ de la misma partícula desde la que se tomó el vector $BinInd$ para ajustar $V1_i$ con el vector $TheBestBinInd$ en la ecuación 6. Tanto $V3_i$ como $RealInd_i$ están controlados por $limit3_{j\ upper}$, $limit3_{j\ lower}$, $limit4_{j\ upper}$ y $limit4_{j\ lower}$, de manera similar a como se controlaron los vectores de velocidad $V1$ y $V2$ en la parte binaria.

Debe notarse que, aunque el procedimiento seguido para actualizar los vectores $V2$ y $realInd$ es el mismo (véanse ecuaciones 8 y 12), los valores de $V2$ se utilizan como argumento en la función sigmoide (ecuaciones 9 y 10).

Esto permite obtener un valor dentro del intervalo (0, 1) que equivale a la probabilidad de que la posición de la partículatome el valor 1. Por lo tanto, se podrán obtener probabilidades dentro del intervalo ($si(limit2_{j\ lower})$, $sig(limit2_{j\ upper})$). Siendo que la función sigmoide para valores extremos produce valores de probabilidad muy similares, cercanos a 0 o 1, permite reducir la posibilidad de cambio enlos valores de las partículas, estabilizando la población.

3.5. Representación de los individuos

Como se vio en la sección 2.11.2, para representar los documentosse utilizan las dieciséis métricas más consideradas en la literatura (detalladas en la tabla 1). Cada sentencia de cada documento del corpusse convierte en un vector numérico cuya dimensión está dada por el número de métricas a usar, en este caso, dieciséis. Por lo tanto, cada documento estará representado por un conjunto de estos vectores cuya cardinalidad coincide con el número de sentencias que contiene en total.

Usar PSO para resolver un problema específico requiere tomar dos decisiones importantes: qué información incluir en la partícula y cómo calcular su valor de aptitud. Durante el proceso evolutivo, las partículas compiten entre sí buscando la mejor solución. En el caso del resumen automático de documentos, consiste en encontrar los coeficientes que, aplicados a los valores de las métricas de cada sentencia, obtienen una selección similar a la establecida por el usuario.

Siguiendo las indicaciones detalladas en la sección anterior, cada partícula está formada por cinco vectores y dos valores escalares. La dimensión de los vectores está determinada por la cantidad de métricasa usar. El vector binario $BinInd$ determina las métricas a considerar según el valor de cada posición. Si la j-ésima posición del vector es 1 significa que la métrica j se considera y si vale 0 significa que no. El vector $RealInd$ incluye los coeficientes que ponderarán la participaciónde cada métrica en el cálculo del puntaje. Los tres vectores restantes sonvectores de velocidad y se manipulan como fue descrito en la sección anterior.

3.6. Aspectos fundamentales del método

El método propuesto comienza con una población de N individuos ubicados aleatoriamente dentro del espacio de búsqueda según los límites preestablecidos. Sin embargo, la parte binaria no se inicializa igual que la continua.

Durante el proceso evolutivo, los individuos se mueven tanto a través del espacio discreto como del continuo según lo visto en la sección 3.4. Algo que debe tenerse en cuenta es cómo modificar el vector de velocidad cuando se usa la función sigmoide (ecuación 9).

PARTE CONTINUA. En la versión continua de PSO, el vector develocidad inicialmente adquiere valores más altos para facilitar la exploración del espacio de soluciones, pero estos se reducen más tarde (normalmente, en forma proporcional al número máximo de iteraciones a realizar) para permitir que la partícula se estabilice buscando la solución en un área

específica identificada como "prometedora". En este caso, el vector de velocidad representa la inercia de la partícula y es el único factor que impide que sea fuertemente atraída, ya sea por sus experiencias previas o por la mejor solución encontrada por la población.

PARTE BINARIA. Por otro lado, al usar la representación binaria dela partícula, aunque el movimiento sigue siendo real, la función sigmoide se ocupa de binarizar el resultado e identificar la nueva posición de la partícula. Para tener capacidad exploratoria, dicha función debe comenzar evaluando valores cercanos a 0, donde hayuna mayor probabilidad de cambio. En el caso particular de la función expresada en la ecuación 9, cuando x es 0, devuelve un resultado de 0.5. Este es el mayor estado de incertidumbre cuando la respuesta esperada es binaria (un 0 o un 1). Luego, a medida que su argumento se aleja de 0, ya sea en dirección positiva o negativa, su valor se estabiliza. Por lo tanto, a diferencia de lo que se hace en la parte continua, cuando se trabaja con PSO binario se debe aplicarel procedimiento opuesto, es decir, comenzar con una velocidad cercana a 0 y luego aumentar o disminuir su valor.

Debido a las razones explicadas anteriormente, en la ecuación 5 se usa un factor de inercia diferente al utilizado en la ecuación 11. Se utiliza $wBin$ para actualizar el vector de velocidad $V1$ y $wReal$ para $V3$. Ambos factores w ($wBin$ y $wReal$) se actualizan dinámicamente segúnla ecuación 13,

$$w = wStart - (wStart - wEnd) * \frac{ite}{maxIte - 1} \qquad (13)$$

donde $wStart$ es el valor inicial de w y $wEnd$ su valor final, ite es la iteración actual, y $maxIte$ es el número total de iteraciones. El uso de un factor de inercia variable facilita la adaptación de la población. Un valor de w superior al comienzo de la evolución permite a las partículas realizar grandes movimientos y alcanzar diferentes posiciones en el espacio de búsqueda. A medida que el número de iteraciones avanza, elvalor de w disminuye, lo que les permite realizar ajustes más precisos.

El algoritmo propuesto utiliza el concepto de elitismo, que preserva al mejor individuo de cada iteración. Una vez que la población termina de moverse, si el mejor actual es peor que el de la generación anterior entonces se lo reemplaza por aquel (quien tuvo el mejor desempeño enla iteración anterior).

El proceso adaptativo finaliza cuando se alcanza el número máximo de iteraciones (indicado antes de iniciar el proceso) o cuando el mejor fitness no cambie (o cambie solo ligeramente) durante un porcentaje determinado del número total de iteraciones.

3.7. Diseño de la función de aptitud

El aprendizaje del criterio utilizado por una persona al resumirun texto requiere que dicha persona resuma un conjunto de documentos previamente. Comúnmente se resaltan aquellas partes del texto consideradas importantes asignando etiquetas. Cada parte deldocumento será de un tipo: “positiva”, si se encuentra en el resumen,o “negativa”, en caso contrario.

Independientemente del problema que se deba resolver, uno de los aspectos más importantes de una técnica de optimización es su función de aptitud. Por plantearse la obtención del resumen como un problema de clasificación se utilizará la matriz de confusión para medir el rendimiento de la solución encontrada por cada partícula.

Entre las métricas más populares utilizadas en ML se decidió utilizar el llamado *coeficiente de correlación de mathews* (MCC por sus siglas en inglés) ya que en este caso puntual *recall*, *precision* y *F-measure* no son útiles para diferenciar la calidad de diferentes soluciones eneste problema. Sucede que las tres tienen el mismo valor al no incluir *true negatives* en su cálculo y ser la suma de *true positives* y *false negatives* igual a la suma de *true positives* y *false positives*. En cambio, MCC considera todas las celdas de la matriz de confusión y, por lotanto, maximiza la precisión global del modelo de clasificación. Como resultado, no debe calcularse el promedio de las matrices de confusión correspondientes a cada documento de entrenamiento. Los valores de MCC oscilan entre (−1, 1), donde 1 corresponde al modelo perfectoy −1 al peor. Finalmente, la aptitud del individuo se calcula de la siguiente manera:

$$fitness(p_i) = \sum_{d \in corpus} \frac{|summary(d)|}{|all\ summaries|} \cdot MCC \qquad (14)$$

Como se puede ver en la ecuación 14, la matriz de confusión utilizada para calcular el valor de MCC debe construirse por cada partícula y cada documento. Construir esta matriz implica reconstruir la solución representada por la partícula en función de sus vectores $binInd$ y $realInd$. El primer vector permitirá identificar las características más representativas del criterio aplicado por el usuario, y el segundo permitirá ponderar cada uno de ellas. Si bien ambos representan la posición de la partícula en cada espacio, del vector continuo solo se usan las posiciones indicadas por el binario.

Por otro lado, como los documentos no tienen el mismo tamaño ysus resúmenes tampoco, el MCC de cada documento se multiplica porel cociente entre la cantidad de sentencias del resumen (indicado por lapartícula) y la cantidad total de sentencias si se suman los resúmenes de todos los documentos (usando lo indicado por el usuario).

Dado que se calculan varias métricas para cada una de las sentenciasde cada documento, se espera que su combinación lineal represente el criterio aplicado por el usuario, tal como se expresa en la ecuación 15.

$$scor(RealInd_i, S_k) = \sum_{j=1}^{n} (realInd_{ij} * s_{kj}) \qquad (15)$$

donde $\sum_{j=1}^{n}(realInd_{ij}) = 1$, siendo $realInd_{ij}$ el coeficiente que el individuo i usará para ponderar el valor de métrica j en la sentencia

k indicada como s_{kj} (véase la ecuación 4). *score* $(RealInd_i, S_k)$ retorna unnúmero entero positivo proporcional a la importancia estimada de la sentencia. Dado que cada coeficiente corresponde al valor de $realInd_{ij}$ para el individuo y está dentro del intervalo $(-limit4_j, limit4_j)$, antes de usarlo para los cálculos, debe ser escalado a $(0, 1)$ usando dichos límites.

De no hacerlo, se permitiría que los valores de las métricas sean restados para ajustar el puntaje e incluso sumados más de una vez. Sin embargo, PSO requiere valores tanto positivos como negativos para mover las partículas dentro del espacio de búsqueda. Una vez que los valores han sido escalados, se dividen por el total para identificar las métricas que tienen una mayor influencia en los cálculos de puntuación.

Cada partícula evoluciona para encontrar coeficientes que, cuando se multiplican por los valores de cada métrica para todas las sentencias, permiten aproximar el resumen producido por el usuario. Una vez que se calculó el puntaje de todas las oraciones en el documento, se pueden ordenar de mayor a menor. Aquellas oraciones que tienen 0 de puntaje se interpretarán como irrelevantes, mientras que aquellas que reciban valores más altos serán más significativas. La preferencia del usuario para una sentencia del documento estará determinada por la puntuación que le asigna la combinación lineal. Luego, el resumen automático del documento se obtendrá considerando las mejores t oraciones, siendo t un umbral definido *a priori*.

Cabe señalar que la evaluación del rendimiento del individuo nose limita a todos los componentes del vector $BinInd$ cuyo valor es 1, sino que el individuo binario se utiliza para generar combinaciones de métricas. Todas las combinaciones posibles se generan al seleccionara lo sumo una métrica

por cada tipo. El único caso en el que no se usaría ninguna métrica es cuando todas las posiciones del vector son 0. Cuando hay un solo 1 en las posiciones del vector correspondientes a un tipo de métrica, la métrica correspondiente será la única de ese tipo que participe en las combinaciones. Este procedimiento no solo permite reducir la dimensionalidad, sino que también ayuda a evitar incoherencias y redundancias entre las métricas incluidas en el criterio de resumen. Por ejemplo, no tendría sentido utilizar simultáneamente dos medidas de posición, una que asigne un mayor peso a las oraciones que se encuentran al principio del documento y otra que haga exactamente lo mismo con las oraciones del final. Eneste caso, el método debe seleccionar la métrica de posición que asigna valores altos a las oraciones ubicadas en cualquiera de los extremos deldocumento.

Después de evaluar todas las combinaciones posibles de métricas, se selecciona aquella que consiga el mayor valor de aptitud. Como resultado, el vector $BinInd_i$ se convierte en el vector $FitInd_i$. Dicho vector mantiene el valor indicado en $realInd_{ij}$ únicamente en las métricas que resulten de la combinación ganadora (las demás posiciones tendrán 0). Para evitar afectar excesivamente la manera en que funciona la técnica de optimización, cada elemento que no participó de la combinación ganadora recibirá una reducción del 2 % en $v1_{ik}$ y una reducciónde 25 % en $v2_{ik}$. Por lo tanto, la posibilidad de que las dimensiones descartadas se seleccionen en el siguiente movimiento de la partícula se reduce. Sin embargo, no se anulan por completo, permitiendo continuar explorando cerca de la solución que está siendo propuesta actualmentepor la partícula.

Finalmente, después de evaluar el valor de aptitud de cada partícula, el vector $FitInd_i$ indicará las métricas a usar y los pesos que formanel criterio aplicado por el usuario (del cual se obtuvo el valor de aptitud almacenado en fit_i), tal como se indica a través de la leyenda "selection of metrics" en la figura 12. Aunque el valor de aptitud delapartícula coincide con los valores indicados por $FitInd$, la partícula sigue moviéndose de la manera convencional utilizando los tres vectores de velocidad completos.

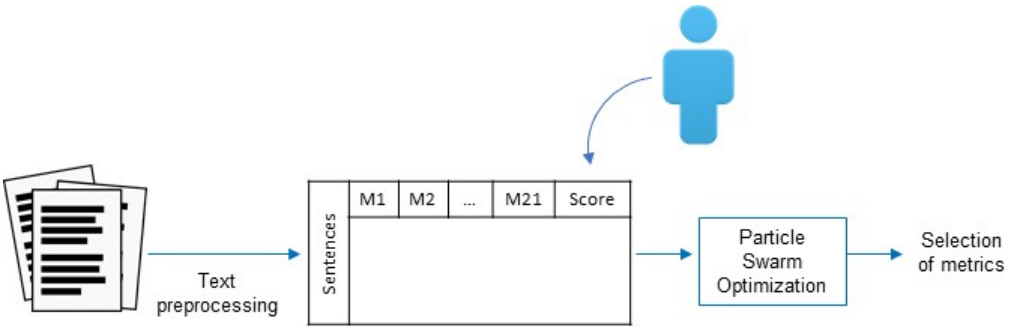


Figura 12: Selección de métricas utilizando PSO.

3.8. Experimentación y resultados

Para evaluar la calidad del resumen automático producido por el método propuesto, el resumen obtenido de cada documento se comparó con el esperado (producido por un ser humano). Para hacer esto, se utilizaron artículos de investigación publicados en una revista médica conocida en idioma inglés. Los documentos fueron descargadosde forma gratuita del sitio web de *PLOS Medicine* en formato *XML* y representados como indica el anexo B.

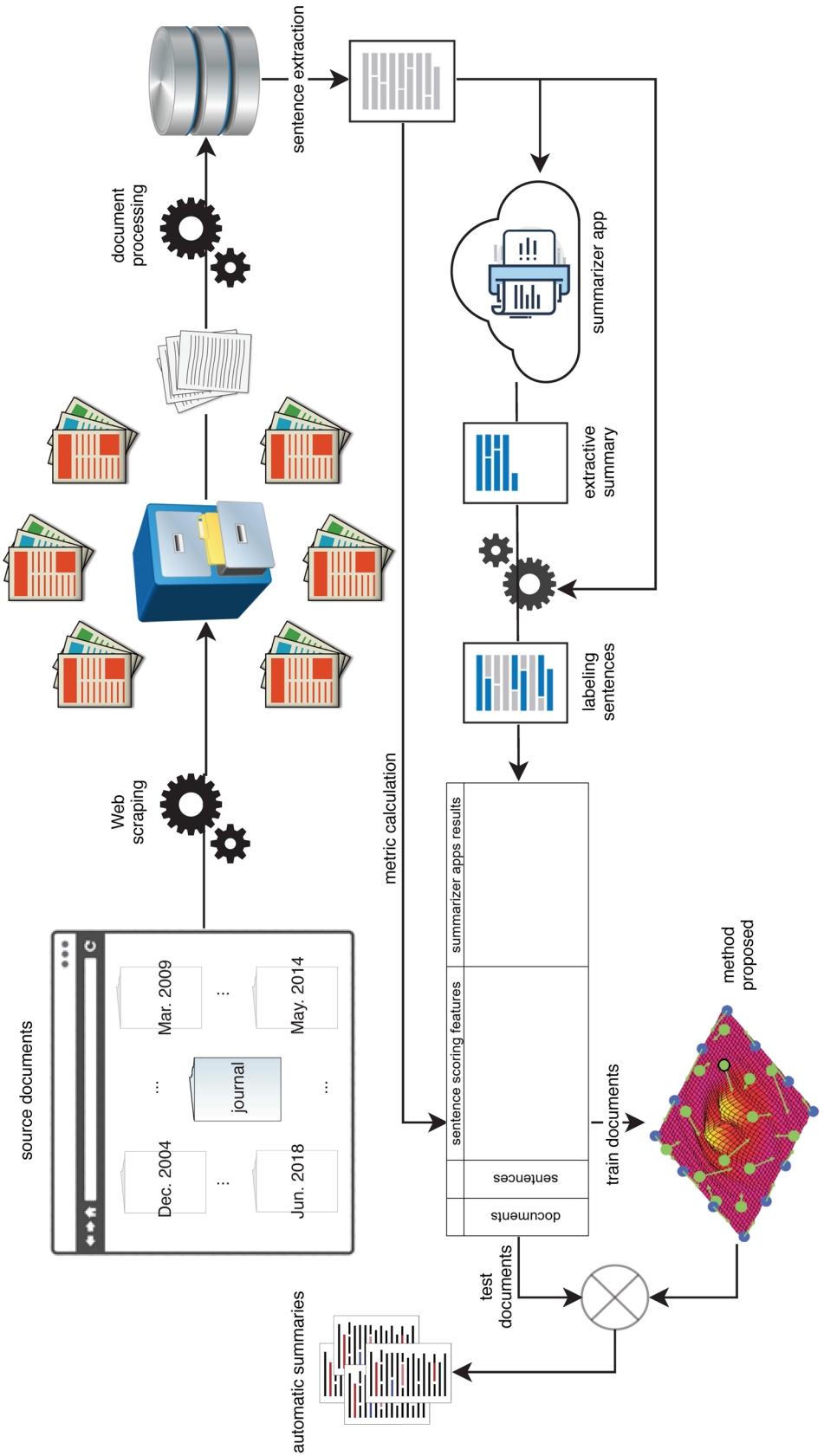


Figura 13: Metodología propuesta para el proceso de generación de resúmenes.

La figura 13 muestra la metodología propuesta. En cada documento se descartaron algunas secciones (“Referencias”, “Agradecimientos”, etc.) así como también todas las figuras y tablas. Luego, se identificaron títulos y párrafos. Cada párrafo se dividió en oraciones usando “.” como delimitador, excepto cuando se lo usó en números y abreviaturas.Luego las oraciones se dividieron en palabras, se eliminaron las *stop-words* y, finalmente, se aplicó *stemming*. Una vez que se completaron todos estos pasos de preprocesamiento, se calculó para cada oración cada una de las métricas descritas en la tabla 1. Es importante luego de calcular las métricas escalar sus valores por documento entre (0, 1). El rango de sus valores varían de documento en documento.

Como se indica en la sección 3.7, una vez que se completa el proceso evolutivo, los resúmenes se crean utilizando los coeficientes de la mejorcombinación de métricas seleccionada por la partícula de la población con la mejor aptitud.

Para conseguir el conjunto de documentos resumidos por elusuario, se utilizó una aplicación web que resume automáticamente documentos. Después de analizar varias aplicaciones de resumen disponibles en Internet, se decidió utilizar *Tools 4 noobs*[9], ya que fuela única que cumplió con los siguientes requisitos: (1) cada sentencia devuelta corresponde con una sentencia del documento, (2) a todaslas sentencias les asigna puntaje, (3) la interfaz web pudo integrarseal desarrollo.

El corpus utilizado consistió en 3.322 artículos publicados entre octubre de 2004 y junio de 2018. Dado el volumen de documentos, se realizó el entrenamiento utilizando los artículos publicados en un mes, y el testeo los

[9] https://www.tools4noobs.com/summarize/

del mes siguiente. El porcentaje del documento a resumirse estableció en 10 % en todos los casos. Este porcentaje fue seleccionado en base a los resultados obtenidos en Villa Monte *et al.* (2016).

Por otro lado, dado que el resultado depende de la inicializaciónde la población, se realizaron 30 ejecuciones independientes para cada método, utilizando un máximo de 100 iteraciones.

La población inicial se inicializó aleatoriamente con una distribuciónuniforme en el caso de la parte continua y en 0 para la parte binaria. Los valores de $limit1$, $limit2$, $limit3$ y $limit4$ fueron los mismos para todas las variables. En el caso de $limit1$ y $limit3$, estos fueron (0; 1) y (0; 0.5), respectivamente, mientras que $limit2$ y $limit4$ tuvieron un valor de (0; 6) en ambos casos. Por lo tanto, los valores de los vectores de velocidad $V1$y $V3$ se limitaron a rangos $(-0.5, 0.5)$ y $(-0.25, 0.25)$, mientras que los de $V2$ y $realInd$ ambos entre $(-3, 3)$.

En cuanto al conocimiento social de cada partícula, se utilizó PSO global. Respecto al tamaño de la población, el mismo fue de 10 partículas en todos los casos. Sin embargo, se podría haber usado una estrategia de población variable como la definida en Lanzarini *et al*. (2008). De esta forma, se comenzaría con una población de tamañomínimo y se ajustaría la cantidad de partículas durante todo el proceso adaptativo, mejorando la eficiencia y eficacia del algoritmo. La variacióndel tamaño de la población se basa en la modificación del algoritmo 1, permitiendo que las partículas evolucionen y mueran durante el proceso de adaptación (agregando y/o eliminando partículas en función de su capacidad para resolver el problema).

Los resultados obtenidos con el método propuesto se comparan con los obtenidos aplicando el método en Villa Monte et al. (2016) al mismocorpus y

utilizando los mismos valores de los parámetros de la parte continua del método propuesto.

La figura 14 muestra el nivel de participación de las métricas para losdos métodos evaluados, ordenados en forma decreciente según el valor de su coeficiente promedio indicado en la tabla 2. Estos coeficientes sonlos que se usan para ponderar el valor de cada métrica al obtener un puntaje de cada sentencia. Por ejemplo, si se observan los tres primerosvalores en la columna "Método propuesto" de la tabla 2, se puede ver que las métricas correspondientes son "tf", "d_cov_j" y "len_ch",cuyos coeficientes promedio son 0.15, 0.13 y 0.13, respectivamente. Porlo tanto, el criterio indicado en la ecuación 15 hace el mismo énfasis entre "d_cov_j" y "len_ch". Sin embargo, si se observa la figura 14, se puede ver que el nivel de participación de "len_ch" es mayor que el de "d_cov_j". Esto se debe a que la primera ha sido seleccionada más veces por la técnica de optimización. En cambio, la métrica "tf" tiene el coeficiente promedio más alto para el método propuesto en la tabla 2, ytambién el nivel más alto de participación en la figura 14.

La figura 15 muestra cómo la precisión de cada uno de los métodos evoluciona a medida que se agregan nuevas métricas. Esto se hace enel orden indicado en la tabla 2. Como se puede ver, el comportamiento

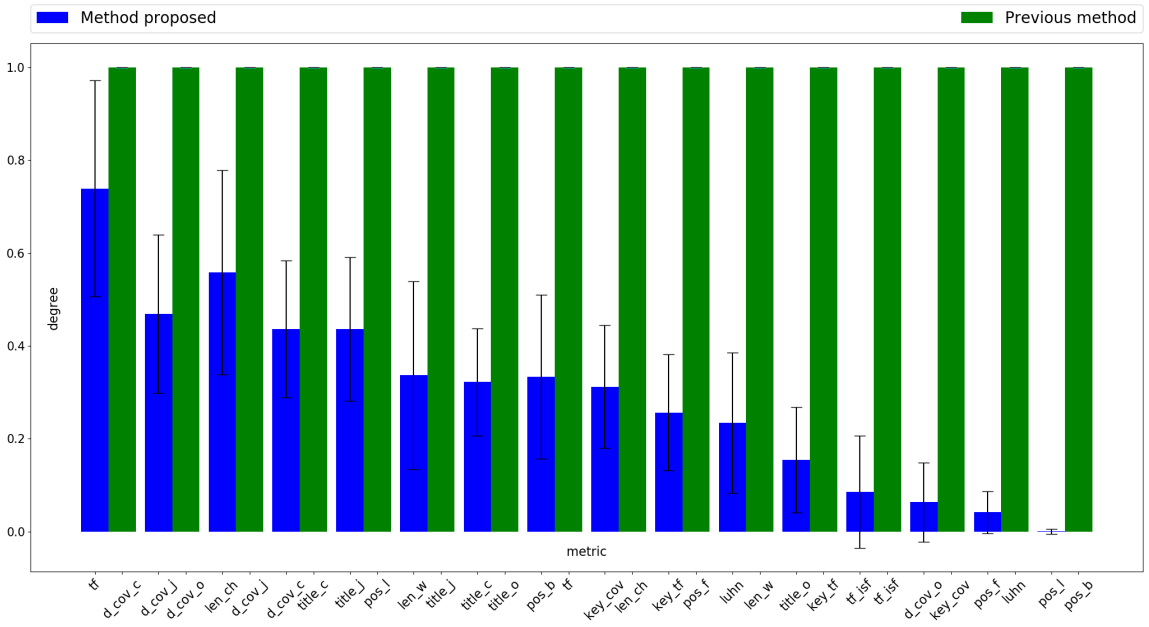


Figura 14: Nivel de participación de las métricas ordenadas en orden descendente por valor de coeficiente.

1lightgray

Métodopropuesto		Método	previo
tf	0.15 ± 0.06	0.97 ± 0.02	d_cov_c
d_cov_j	0.13 ± 0.06	0.97 ± 0.03	d_cov_o
len_ch	0.13 ± 0.06	0.96 ± 0.03	d_cov_j
d_cov_c	0.12 ± 0.05	0.96 ± 0.03	title_c
title_j	0.08 ± 0.04	0.95 ± 0.03	pos_l
len_w	0.08 ± 0.05	0.95 ± 0.03	title_j
title_c	0.06 ± 0.03	0.95 ± 0.03	title_o
pos_b	0.06 ± 0.04	0.95 ± 0.03	tf
key_cov	0.05 ± 0.03	0.95 ± 0.03	len_ch
key_tf	0.04 ± 0.03	0.95 ± 0.03	pos_f
luhn	0.04 ± 0.03	0.94 ± 0.04	len_w
title_o	0.03 ± 0.02	0.92 ± 0.04	key_tf
tf_isf	0.01 ± 0.02	0.92 ± 0.04	tf_isf
d_cov_o	0.01 ± 0.02	0.92 ± 0.04	key_cov
pos_f	0.01 ± 0.01	0.90 ± 0.05	luhn
pos_l	0.00 ± 0.00	0.89 ± 0.05	pos_b

Tabla 2: Coeficientes promedio obtenidos con cada métrica. Estos valores corresponden a la media y desviación correspondiente.

del método propuesto aquí es más estable. Además, después de agregarla cuarta métrica, la precisión se vuelve notablemente mejor que la queobtuvo el método en Villa Monte *et al.* (2016). Aunque el valor máximose observa con la participación de 7 métricas, 4 serían suficientes para

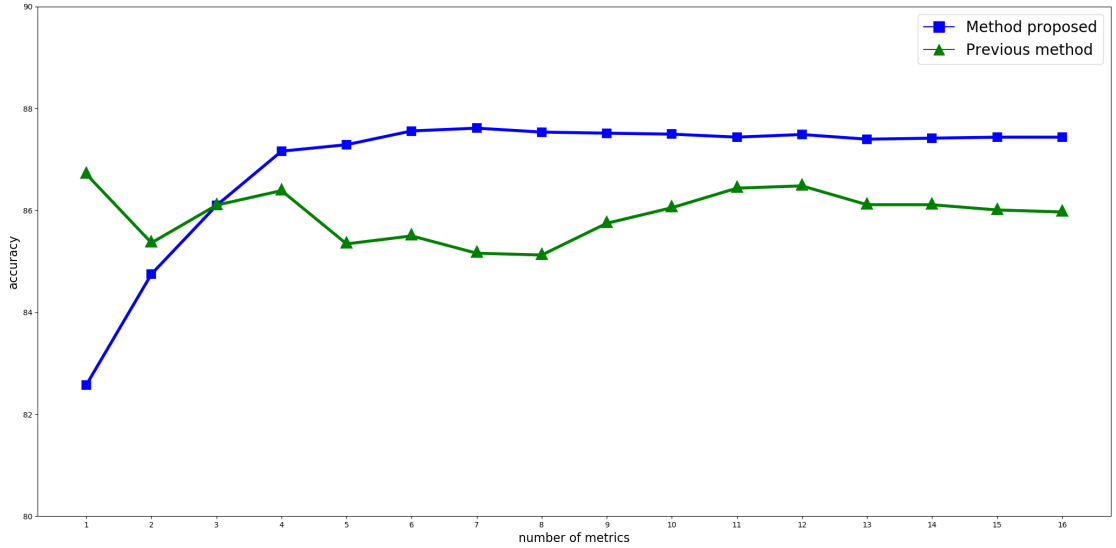


Figura 15: Evolución de la precisión a medida que se agregan nuevas métricas para calcular el puntaje.

obtener un buen rendimiento. También se debe tener en cuenta que, utilizando el método descrito en Villa Monte *et al.* (2016), incluso sila precisión resultante es mayor para las dos primeras métricas, las restantes arrojan un resultado inferior en comparación con el método propuesto, sin lograr superar el valor más alto de 87.61 %.

Finalmente, el método propuesto es capaz de identificar la importancia que cada métrica tiene al simular el criterio del usuario. Esto es evidente a partir de la estabilidad lograda en la precisión una vez incorporada la cuarta métrica, tal como se observa en la figura 15, así como en la magnitud de los coeficientes enumerados en la tabla 2.

3.9. Conclusiones

En este capítulo, se ha descrito en detalle un nuevo método para obtener resúmenes orientados al usuario mediante la representación de sentencias basada en métricas de puntuación y la aplicación de una técnica mixta de

optimización discreta-continua. La estrategia permite encontrar automáticamente, a partir de documentos de entrenamiento etiquetados por el usuario, las métricas y los pesos óptimos que se utilizarán para resumir los documentos aplicando el mismo criterio.

El capítulo es un claro ejemplo de los puntos fundamentales que son necesarios definir para aplicar una técnica de optimización en la resolución de un problema: representación a utilizar, mecanismo para medir el desempeño de las soluciones halladas, operadores para modificar y mejorar las soluciones existentes, criterios de terminación y valores adecuados de los parámetros que controlan la técnica. Cada uno de estos puntos requiere de un estudio minucioso de la situación. En el anexo A se describirá cómo la técnica presentada en este capítulo puede utilizarse para obtener reglas de clasificación.

Los resultados obtenidos de aplicar el método propuesto a documentos confirman que las métricas seleccionadas producen una precisión adecuada, siendo ponderadas según lo indicado por la mejor solución obtenida utilizando la técnica de optimización propuesta. Las pruebas realizadas arrojaron mejores resultados que los establecidos previamente para un amplio conjunto de artículos científicos de una conocida revista médica.

CAPÍTULO 4

Resumen mediante grafos causales y concomponentes temporales

En el capítulo 2 se diferenciaron dos de los enfoques más utilizados para resumir automáticamente documentos de texto. Sin embargo, existen otros menos convencionales que buscan extraer determinados patrones textuales para construir un resumen a partir de grafos.

Este tipo de enfoque generalmente utiliza el grafo para representar el contenido de un conjunto de documentos. En dicho grafo se identifican las partes más relevantes de los documentos que permiten construir oraciones que constituirán finalmente el resumen. En esta dirección,se han realizado muchos estudios utilizando relaciones conceptuales, centrándose principalmente en aspectos semánticos y no tanto en la causalidad.

La causalidad cumple un rol importante en cualquier toma de decisiones. Proporciona información que permite elegir una determinada acción que pueda conducir a un resultado deseado.

Generalmente, se caracteriza por una relación que sigue el esquema “A causa B”, donde A es la causa y B el efecto. Sin embargo, no solo se trata de una cuestión de afirmación causal, sino también de expresión condicional. La causalidad y la condicionalidad están fuertemente relacionadas.

En este capítulo se presentarán aspectos básicos de una relación causal, varios conceptos relacionados y las principales áreas de aplicación. Se tratará la representación de sentencias causales y condicionales exponiendo sus principales diferencias y similitudes. Se proporcionarán los recursos necesarios para extraer de varios documentos de texto este tipo de sentencias y luego convertirlas en un grafo equivalente, en el cual puedan apreciarse las principales relaciones que describen el texto mostrando vínculos “causa-efecto” entre sus nodos. Además, se estudiarán dichas relaciones haciendo hincapié en restricciones temporales que afectan su interpretación, señalando los muchos beneficios que pueden obtenerse de su aplicación en el área de la salud.

4.1. Introducción a la causalidad

La causalidad ha jugado y sigue jugando un papel importante enla cognición humana. La idea de que el conocimiento causal es una característica esencial de nuestra comprensión del mundo es muy antigua. Desde los tiempos de Aristóteles, la causalidad fue motivo de estudio. Debido a su importancia, la causalidad ha sido abordada desde los años 90 a través de diversos estudios. Filósofos, científicos, físicos, matemáticos y hasta informáticos han explorado este campo.

En la práctica científica es muy común proporcionar contenido causal. La causalidad está sumamente involucrada con una de las tareas más importantes de la ciencia empírica: la explicación. La lógica causal ofrece un vocabulario y reglas para explicar y predecir procesos complejos en términos de vínculos "causa-efecto". La causalidad no solo es importante para contestar preguntas sino también para acompañar con una explicación la respuesta. Por otro lado, las causasson relevantes en la toma de decisiones porque los efectos de una decisión pueden determinarse por sus causas. Desde ese punto de vista, la causalidad permite generar conocimiento científico objetivo (Kantet al., 1999; Mill, 1843) y describir fenómenos (Mach, 1976).

La física, la ingeniería y la medicina tienen una larga historia contribuyendo al conocimiento humano. Gran parte del conocimiento proporcionado por esas disciplinas es causal y, desde sus comienzos, selo ha almacenado en formato texto. Por ese motivo resulta interesante desarrollar mecanismos para localizar y representar el contenido causalde documentos permitiendo establecer relaciones entre conceptos ocultos a simple vista.

4.2. Fundamentos básicos de una relación causal

Describir relaciones de causas y efectos es fundamental en todas las ciencias. Tradicionalmente, se las ha utilizado en física para analizar eventos y demostrar hipótesis (Bunge, 1979). En las ciencias empíricas,la causalidad es una forma útil de generar conocimiento y proporcionarexplicaciones.

La causalidad es una relación en la cual la ocurrencia de una entidad B de cierta clase depende de la ocurrencia de una entidad A de otra clase.

Generalmente, por “entidad” se entiende un fenómeno, un hecho, una característica, una situación o un evento, entre otras cosas. En este tipo de relación, “A” representa una causa y “B” un efecto. Entre ambos existe una clara relación de dependencia, ya que la causa provoca un efecto, y el efecto se deriva de la causa, considerándola su consecuencia. Se expresa no solo utilizando el término “cause” sino también “produce”, “bring about”, “issue”, “generate”, “result”, “effect” o “determine”, entre otros (Kim, 1995).

La causalidad puede ser un proceso directo cuando A causa B y Bes un efecto directo de A; o un proceso indirecto cuando A causa Ca través de B, y C es un efecto indirecto de A. Tratándose de textola fuente de información, para extraer las relaciones causales directas se necesita tener un amplio conocimiento del lenguaje que permitadefinir correctamente una serie de patrones de búsqueda específicos (véase la lista de la figura 18). En cambio, no ocurre lo mismo conlas relaciones causales indirectas. Este tipo de relaciones no se puede identificar a través de patrones. A pesar de ello, extrayéndose relaciones causales directas y representándolas adecuadamente en forma de grafo (como el de la figura 20), a través del recorrido de sus caminos pueden establecerse relaciones indirectas que no eran obvias y no podríanhaberse obtenido de otra manera.

Para considerar como causal una relación deben cumplirse, según Sowa (2000), tres supuestos descritos por Born (1949):

- *Evidencia* ⇒ Causas y efectos se demuestran unos a otros.
- *Precedencia* ⇒ Las causas deben ocurrir con anterioridad a sus efectos o, al menos, en forma simultánea.

- *Contigüidad* ⇒ Las causas se encuentran contiguas a los efectos inmediatos o, de lo contrario, se conectan por medio de otros elementos causa-efecto intermedios.

Por otra parte, además de las anteriores características, existen otrasdos que serán tratadas en las próximas secciones:

- *Imperfección* ⇒ Las causas, los efectos y los vínculos causa-efecto nosiempre son precisos.
- *Temporalidad* ⇒ Las asociaciones entre causas y efectos suelen acompañarse de restricciones de tiempo.

La causalidad no se relaciona únicamente con afirmaciones causales. Está relacionada con expresiones condicionales de la forma "Si A entonces B", un formato similar al de una regla de clasificación (véase la sección A.1). En las declaraciones condicionales, la causalidad generalmente surge de la relación de vinculación entre el antecedente yla consecuencia.

Si bien la causalidad y la condicionalidad tienen una zona de intersección en la que comparten similitudes, no toda sentencia condicional es causal. Por ejemplo, expresiones como «If Marseille isin France, two is an even number» no son, en general, causales. Hay otras expresiones condicionales pero con significado causal como «If somebody drinks two grams of cyanide, he will die», que son más bien casi lo mismo que decir «Drinking two grams of cyanide causes death». Por lo tanto, se deben considerar las oraciones condicionales como portadoras de un significado causal.

Si bien, la forma típica de causalidad es “A causa B” y la de condicionalidad “Si A entonces B”, existen otros formatos. Expresiones como “B is due to A”, “A produces B”, “B is necessary for A”, “B if A”, etc. son alternativas para expresar causalidad y condicionalidad. Estas formas también deben tenerse en cuenta. Independientemente de su formato, una relación causal se define siempre por un elemento inicial (o causa A), un elemento final (o efecto B) y la relación entre ambos.

4.3. La causalidad y el tiempo en Medicina

Las sentencias causales constituyen una parte principal de cualquier explicación médica. Este tipo de sentencias están presentes al detallar las causas de una enfermedad o al informar los efectos de un tratamiento. Con frecuencia, las Ciencias de la Salud muestran la causalidad como un proceso complejo que involucra la evolución de una causa anterior en causas intermedias a lo largo del tiempo antes dealcanzar el efecto final.

Como se dijo anteriormente, las explicaciones causales relacionan causas y efectos y, en Medicina, a menudo se las restringe temporalmente. El tiempo juega también un papel importante en diferentes escenarios médicos (Keravnou, 1996):

- *Prevención* ⇒ «Smoking for a long time causes lung cancer» Evaluar factores de riesgo en pacientes que se comportan de ciertamanera.
- *Pronóstico* ⇒ «25 % of patients with septic shock will die within 15 days»
  Predecir la probabilidad de supervivencia de una persona.

- *Tratamiento* ⇒ «Omeprazole should be taken before aspirin»
  Administración de medicamentos durante ciertos períodos detiempo o bajo ciertas restricciones temporales.
- *Diagnóstico* ⇒ «According to the symptoms that he presents, he hasflu»
  Identificación del origen o causa de las enfermedades.

La toma de decisiones basada en el tiempo cubre tres áreas de la atención al paciente, donde la administración de medicamentos duranteciertos períodos de tiempo o bajo ciertas restricciones temporales resulta fundamental:

(I) *Medicación tiempo-dependiente en forma crítica*. Existen algunos medicamentos que son muy sensibles a su administración en el tiempo. El ISMP[10] publicó una lista actualizada con algunos medicamentos que deben proporcionarse en un período de tiempo específico para lograr los resultados esperados[11]. Si no se lo hace, pueden causarle al paciente un daño grave, ya seaporque no son efectivos o porque provocan un daño adicional. Los errores en la administración de este tipo de medicamentos no solo generan un problema de salud, sino también financiero. Los problemas causados por la ineficacia de los tratamientos pueden generar reclamos que demandan compensaciones o gastos extrahospitalarios (Guo *et al*., 2011).

(II) *Cronoterapia*. La toxicidad y eficacia de algunos medicamentos dependen del momento en que se administran (Lucio, 2018). Por ejemplo, la administración de corticoides una vez al día por la mañana reduce la supresión adrenocortical, pero si la dosis se divide en cuatro

[10] https://www.ismp.org/
[11] https://www.ismp.org/sites/default/files/attachments/2018-08/highAlert2018-Acute-Final.pdf

administraciones, el efecto es otro. En cronoterapia, las recomendaciones de administración según el tiempo para medicamentos recetados regularmente son un tema relevante para los profesionales de la salud y contribuyen a la eficacia terapéutica (Martiny *et al*., 2015).

(III) *Interacción farmacológica dependiente del tiempo*. La administración de dos o más medicamentos, uno de los cuales es incompatible conel otro, puede provocar una disminución de la absorción, afectar su metabolismo o producir daños graves (incluida la muerte) en caso de incompatibilidad absoluta (Rodvold and Kraus, 2010). La administración de medicamentos por separado por un intervalo de tiempo adecuado (con frecuencia, 2-4 horas) puede evitar esosinconvenientes (Van der Sijs et al., 2009).

Una parte esencial de la medicina es el estudio del origen de las enfermedades (etiología) y el diagnóstico en función de las causas (Thagard, 1998). Con frecuencia, el diagnóstico de una enfermedadse relaciona con el momento en que aparecen los síntomas. Esto se puede verificar en textos de dominio médico, como formularios deinscripción, correos electrónicos, informes o historias clínicas. En el campo de la medicina, los textos escritos en lenguaje natural son una fuente primordial de información donde encontrar referencias de tiempo (Augusto, 2005).

El lenguaje natural es el medio que utilizan los pacientes para expresar sus dolencias y los médicos su conocimiento sobre enfermedades. Este lenguaje incluye frecuentemente alusiones a aspectos temporales y, a menudo, carece de precisión porque el paciente ignora u olvida datos sobre los cuales se le interroga.

Los humanos han inventado calendarios y relojes para medir y controlar el tiempo. Sin embargo, tal como existe el tiempo objetivoo cronológico medido por dispositivos, también existe el tiempo subjetivo, a través del cual los humanos experimentan la misma duración de diferentes maneras Bardon (2013).

En los últimos años, el reconocimiento automatizado de conceptos médicos, independientemente de ser causales o no, ha cobrado mayor relevancia teniendo en cuenta su aplicación en la recuperación de documentos médicos, entre otras (Li and Wu, 2006). Para ello es imprescindible la identificación, extracción y representación de determinadas frases del documento. A continuación se tratarán estas tres tareas en base a la información causal temporal que se desea.

4.4. Representación de información temporal

El diagnóstico médico, en su mayoría, se basa en asociaciones causales que implican dependencias temporales. Con frecuencia, los síntomas se prolongan en el tiempo y así la enfermedad no es vista como una imagen estática sino progresiva. Por otro lado, su tratamiento también depende del tiempo. Algunos medicamentos deben ser programados rigurosamente para conseguir los beneficios previstos.

Los diagnósticos médicos y los tratamientos realizados pueden recuperarse de los informes médicos y las recetas. Inspeccionando este tipo de documentos en formato texto se pueden observar algunas características particulares del léxico utilizado.

El tiempo utiliza un léxico específico. Las oraciones temporales incluyen con frecuencia referencias relacionadas con el calendario (año, mes, día) o momentos del día (mañana, tarde, noche). También utilizan conjunciones y preposiciones como "by", "until", "before", "since", "past", "next". Además, hay frases causales como "secondary to" o "because of", que también denotan influencia causal temporal.

Asimismo, con frecuencia dichas expresiones temporales son borrosas, como se muestra a continuación:

«A little before the admission, ...»
«Few days ago he received a very high dose.»

Este tipo de oraciones incluyen:

Cuantificadores difusos	«He was bleeding most of the previous days.» «He was discharged a few days later».
Atenuantes lingüísticos	«He was admitted very early.» «It was rather sedentary than active.»
Adjetivos temporales	«His temperature will fall in the subsequent 24 hours.» «During the previous 24 hour period, ...»
Adverbios temporales	«He hasn't eaten anything lately.» «Anesthetics provide pain relief for several hours afterward».

Existen otras palabras como «occasional», «abruptly» o «still» que también denotan temporalidad difusa (Zhou and Hripcsak, 2007).

La representación de información temporal tiene como objetivo estudiar y formalizar las relaciones entre entidades temporales. Una posible aproximación se consigue analizando la duración. Es así que Allen (1983) definió una ontología de relaciones temporales basada en tres tipos

básicos: (a) Intervalo-Intervalo, (b) Instante-Instante y (c) Instante-intervalo.

4.5. Relaciones de tiempo de Allen

Según Allen (1984), la ontología del tiempo se realiza a partir de instantes e intervalos. Un instante de tiempo es un elemento únicoque denota un evento instantáneo. En cambio, un intervalo es un par ordenado de instantes de tiempo, siendo el primero de ellos anterior al segundo (o menor que el segundo en alguna escala de tiempo).

Relación	Símbolo	Inversa	Ilustración
A before B	<	>	AAA BBB
A occurs at the same time as B	=	=	AAA BBB
A meets B	m	mr	AAA BBB
A overlaps B	o	or	AAA BBB
A during B	d	dr	AAA BBBBBB
A starts B	s	sr	AAA BBBBBB
A finishes B	f	fr	AAA BBBBBB

Tabla 3: Relaciones entre intervalos.

Relación	Símbolo	Inversa	Ilustración
tp before B	<	>	■ BBB
tp during B	=	There isn't	BBB■BBB
tp starts B	s	There isn't	■ BB
tp overlaps B	o	There isn't	■

Tabla 4: Relaciones entre instantes e intervalos de tiempo.

Las relaciones temporales básicas son (I) entre intervalos de tiempo, (II) entre instante e intervalo de tiempo, y (III) entre instantes de tiempo. Las tablas 3, 4 y 5 ilustran en detalle dichas relaciones.

Relación	Símbolo	Ilustración
p_i occurs at the same time as p_j	=	
p_i before p_j	<	■
p_i inverse before p_j (after)	>	

Tabla 5: Relaciones entre instantes de tiempo.

Como se dijo anteriormente, las relaciones de tiempo pueden ser precisas o difusas. La fuzzificación de las relaciones de Allen puede conseguirse utilizando valores o intervalos difusos. En la próxima sección se describirá cómo.

4.6. Borrosificación de la ontología de Allen

Los instantes e intervalos temporales de la ontología de Allen son elementos de tiempo precisos que no generan confusión. Por desgracia, las sentencias causales van acompañadas frecuentemente de restricciones temporales difusas. Esto se nota especialmente en informes médicos o en frases como las siguientes:

«A drug causes a therapeutic effect if it is administrated **a little before** the meal.

«To lose weight you can not eat protein **at night**.»

---

siendo en el primer caso “a little before” un instante difuso y, en el segundo, “at night” un intervalo difuso (porque el período nocturno varía según las diferentes estaciones y culturas). Frases como las anteriores sugieren que debe suavizarse la representación de las relaciones de Allen para conseguir instantes e intervalos de tiempo difusos.

Según Dubois *et al*. (2003), un instante de tiempo difuso p es un punto que no tiene asociado en el tiempo un momento específico. «He arrived at about 5 o’clock» o «he is almost 15 years old» son ejemplos. Los puntos de tiempo difusos contrastan con los precisos como «five

(a) «five o’clock» (b) «about five o’clock»

Figura 16: Representación (a) precisa y (b) difusa de un instante de tiempo.

o’clock», por ejemplo. La figura 16 representa ambos tipos de instantes: el caso preciso, representado con una función de un solo paso, y el borroso, con una función de caída suave (una función triangular).

Como puede verse en la figura, un punto de tiempo difuso se modela a través de una función de distribución de posibilidad, $\pi_p : T \rightarrow (0, 1)$. Entonces, $\forall t \in T, \pi(t) \in (0, 1)$ es la estimación numérica de la posibilidad de que p sea exactamente t.

Si una acción se refiere a más de un instante t, se debe proporcionar un intervalo. Dubois *et al*. (2003) describe un intervalo difuso comoun par de

conjuntos difusos)A, B (y (A, B). El conjunto), B (es un conjunto de puntos de tiempo que están más o menos con certeza entreA y B, y el conjunto (A, B) es un conjunto de puntos de tiempo que probablemente están entre A y B. Estas medidas de proximidad puedenser calculadas por:

$$\mu_{(A,B)}(t) = sup_{s \le t \le s'} min(\pi_A(s), \pi_B(s')) = (A, +\infty) \cap (-\infty, B)$$

$$\mu_{)A,B(}(t) = min(\mu_{)A,+\infty)}(t), \cap(\mu_{(A-\infty B(}(t)) = (-\infty, A)^C \cap (A, +\infty)^C$$

La figura 17 ilustra un ejemplo de intervalo de tiempo difuso (con distribuciones sin superposición) como es el de la oración «between about 3 and about 6».

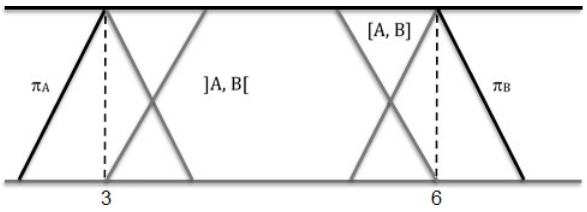


Figura 17: Representación del intervalo difuso «between about 3 and about 6».

Desde el comienzo de la inteligencia computacional, fue relevante tener modelos de información temporal y determinar su uso en argumentos lógicos o inferencias causales. En AI, las inferencias se hacen generalmente con lenguaje natural, el cual involucra, especialmente en el campo de la medicina, restricciones vagas.

Para recuperar sentencias causales con restricciones temporales parece apropiado tener un modelo que: (a) distinga elementos primitivos

epistemológicamente fértiles, (b) sea capaz de integrarse con otro tipo de conocimiento y (c) que coopere con tareas de razonamiento.Como el tiempo es algo frecuentemente difuso, es conveniente que el modelo permita su fuzzificación. La lógica temporal de Allen y su fuzzificación según Dubois *et al*. permiten su aplicación a textos en la recuperación de expresiones causales con restricciones temporales y su representación mediante grafos. En las próximas secciones se abordaráel tema como una serie de procesos que deben ser ejecutados en orden para lograr tal fin.

4.7. Extracción y análisis de relaciones causales

Para analizar el comportamiento de las sentencias condicionales y causales en documentos de texto, se requiere contar con un proceso capaz de detectar, extraer y clasificar las sentencias con base en ciertos patrones estructurados. Pero antes es necesario dividir cada frase en sus componentes básicos (verbos, adverbios, pronombres, sustantivos, etc.).

La detección de sentencias condicionales y causales es un proceso difícil de realizar debido a la complejidad inherente del lenguaje, la gran variedad de combinaciones lingüísticas, contextos y diversidad designificados. Además, lidiar con problemas reales a través de solucionescomputacionales introduce un grado de complejidad mayor.

La gramática inglesa, siendo bastante estándar en comparación con otros idiomas, posee 48 estructuras sintácticas que expresan contenidocondicional y causal. Algunas expresiones se basan en el tiempo verbal, como en la frase «By playing well, we'll win the match». Algunas otras dependen de ciertos adverbios y conjunciones que se usan para formar la frase, como por ejemplo, «The car does not start because it has no petrol». Ambos casos permiten

generar patrones que se utilizarán para la extracción del contenido causal de un documento y la generación de conocimiento.

La figura 18 muestra 20 de los patrones sintácticos usados comúnmente para expresar causalidad en el idioma inglés. El listado completo de estructuras junto con la explicación detalla de cada caso

1: if + present simple + future simple
2: if + present simple + may/might
3: if + present simple + must/should
4: if + past simple + would + infinitive
5: if + past simple + might/could
6: if + past continuous + would + infinitive
7: if + past perfect + would + infinitive
8: if + past perfect + would have + past participle
9: if + past perfect + might/could have + past participle
10: if + past perfect + perfect conditional continuous
11: if + past perfect continuous + perfect conditional
12: if + past perfect + would + be + gerund
13: for this reason, as a result
14: due to, owing to
15: provided that
16: have something to do, a lot to do
17: so that, in order that
18: although, even though
19: in case that, in order that
20: on condition that, supposing that

Figura 18: Estructuras condicionales y causales implementadas.

puede consultarse en Puente (2010). Si bien en la mencionada figura se muestra una serie de patrones conceptuales que conforman oraciones condicionales en inglés, estas mismas estructuras podrían ser la entradade un proceso capaz de reconocerlas dentro de documentos de texto.

Transcribiendo las estructuras de aquellos patrones que no presentan algún tipo de ambigüedad, puede construirse un programa capazde filtrar oraciones condicionales y causales dentro de un texto que coincidan con cualquiera de

los patrones seleccionados. En Puente*et al*. (2010) se explica en profundidad el algoritmo utilizado para crear un autómata de 4 estados que, mediante un analizador morfológicoy sintáctico, puede detectar y clasificar las oraciones en base a dichos patrones. Dicho analizador será el punto de partida para los procesos posteriores.

Tabla 6: Comparación de resultados obtenidos de la extracción y clasificación de sentencias causales por cada tipo de documento analizado.

Tipo de texto	Científicos	Médicos	Novelas	Noticias	Evangélicos
Recall	0.65	0.76923	0.32431	0.57893	0.5
Precision	0.8387	0.90909	0.54544	0.7857	0.7
F-measure	0.7323	0.8333	0.40672	0.6667	0.5833

De la realización de experimentos con textos pertenecientes a diferentes ámbitos, se descubrió que el comportamiento del algoritmo variaba según el tipo de texto analizado. Por esta razón, fueron analizados manualmente algunos documentos pertenecientesa diferentes categorías (50 páginas por cada una), y se calcularon medidas estándar como *recall* (calculado como el número de oraciones causales correctamente clasificadas por el sistema, dividido por el número de oraciones causales clasificadas manualmente), *precision* (calculada como el número de oraciones causales correctamente clasificadas por el programa, dividido por la cantidad total de oraciones recuperadas) y *F-measure* (combinación de *recall* y *precision*). Los resultados obtenidos, mostrados en la tabla 6, demostraron mayor desempeño en textos científicos y médicos. Por tal motivo, fueron utilizados documentos de este tipo en los experimentos finales. En especial, se usaron documentos relacionados con el cáncer de pulmón.

4.8. Creación del grafo causal

Las sentencias finalmente extraídas por el proceso descrito en la sección anterior permiten luego crear una base de conocimiento causal sobre un determinado tema. Esto es posible siempre que sean procesados documentos de una misma temática. Si se trataran páginas web, podrían conseguírselos a través de la realización de una simple búsqueda en Internet y posterior recopilación del contenido de los sitios arrojados como resultado.

Una vez producida la base de conocimiento (mediante la extracción y clasificación de sentencias causales y condicionales) en un determinado ámbito y sobre un tema en particular, el usuario deberá introducir una pregunta. Dicha pregunta permitirá seleccionar aquellas oraciones que se encuentren directamente relacionadas con ella. En Sobrino *et al*. (2014) fue desarrollado, combinando un analizador léxico y un etiquetado gramatical, un programa capaz de seleccionar el concepto principal de la consulta de entrada y así poder determinar si el usuario está preguntando por causas o por consecuencias.

A continuación, se presenta un ejemplo en el cual se establece la relación entre dos conceptos: «smoking» y «lung cancer». Como fuentesde información confiables en las cuales buscar «cancer», se utilizaron principalmente documentos de los siguientes sitios web: (a) Mayo Clinic[12], (b) American Society of Clinical Oncology[13], (c) Centers for Disease Control and Prevention[14] y (d) eMedicineHealth[15] .

Si el usuario preguntase, por ejemplo, «What provokes lung cancer?», un Part-Of-Speech (POS) Tagger permitirá remarcar la cláusula «lung cancer»

[12] https://www.mayoclinic.org/
[13] https://www.asco.org/
[14] https://www.cdc.gov/
[15] https://www.emedicinehealth.com/

etiquetando como sustantivo tanto «lung» como «cancer». Además, se detectaría tanto la palabra «provokes» como el pronombre interrogativo «what», permitiendo suponer que el usuario está preguntando por la causa del cáncer de pulmón.

Una vez que la cláusula ha sido seleccionada y aislada del resto delas palabras, otro programa extrae aquellas oraciones en las que están contenidos estos conceptos. El conjunto en el cual buscar es el archivo creado con las oraciones condicionales y causales filtradas previamentea través de los patrones vistos en la sección 4.7. El conjunto de oracionesque resulte de este segundo filtrado servirá para construir el resumen final.

Pero antes, las oraciones obtenidas a partir del concepto «lung cancer» son almacenadas en una base de datos, a la cual accede un proceso de graficación. Luego, una vez que todas las frases hayan sido procesadas rastreando los conceptos asociados a causas o consecuencias, se construye el grafo causal de la figura 20 que las representará. El esquema del proceso completo de obtención del grafo se muestra en la figura 19.

Los grafos causales son una forma gráfica de mostrar las dependencias causales que posee la información no estructurada como lo es el texto. La causalidad implica una transferencia de la causa al efecto que en el grafo se denota mediante una flecha dirigida. Dicha flecha conecta el nodo “causa” con el nodo “efecto” en el grafo causal.

Como podemos ver en el grafo de la figura 20, hay cuatro nodoscon la palabra «smoking», o palabras relacionadas, y otros cuatro con las palabras «lung cancer». El resto de los nodos han sido recuperados durante el proceso. Cada nodo representa un concepto

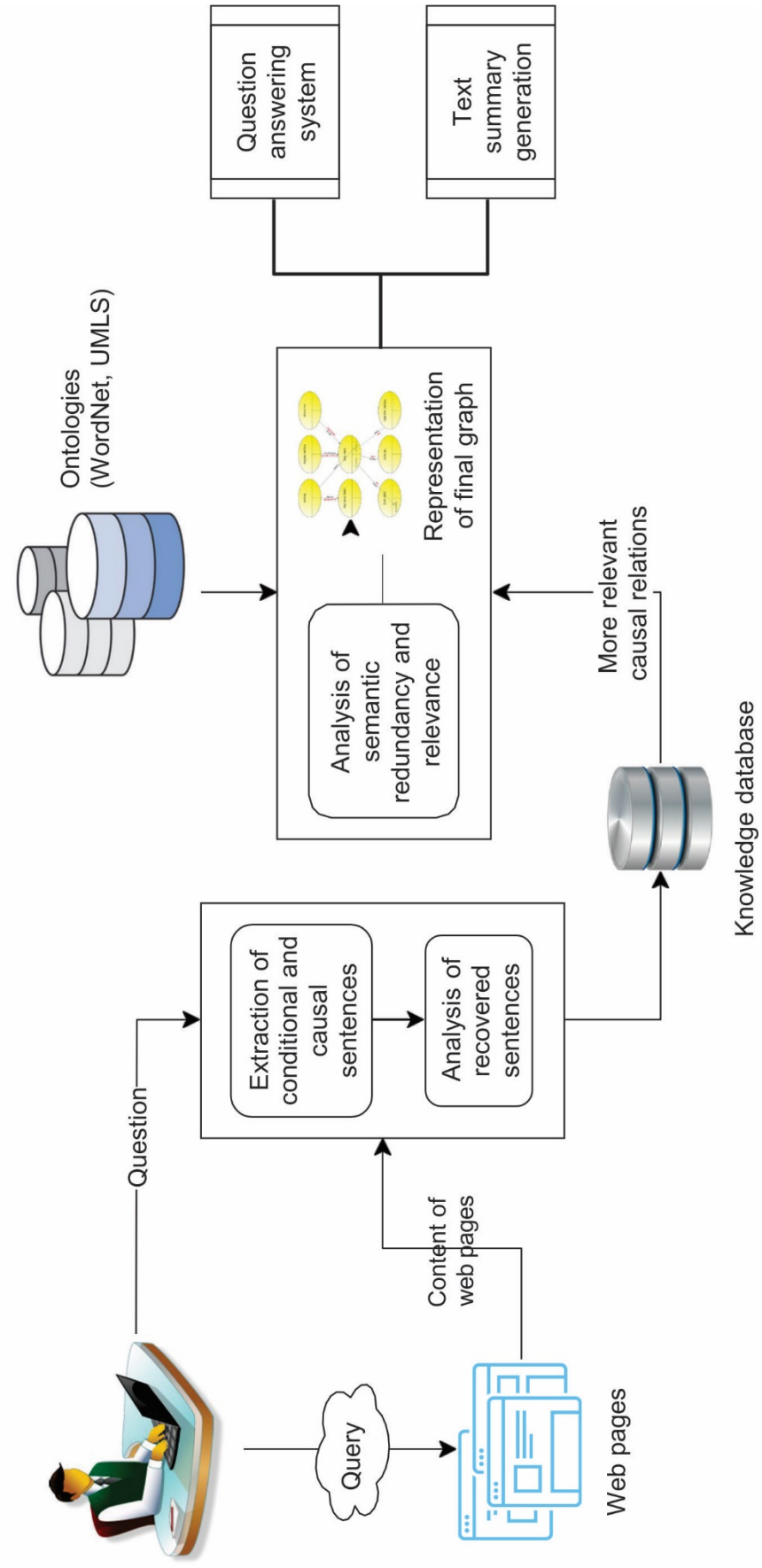


Figura 19: Proceso para obtener un grafo causal a partir de texto plano.

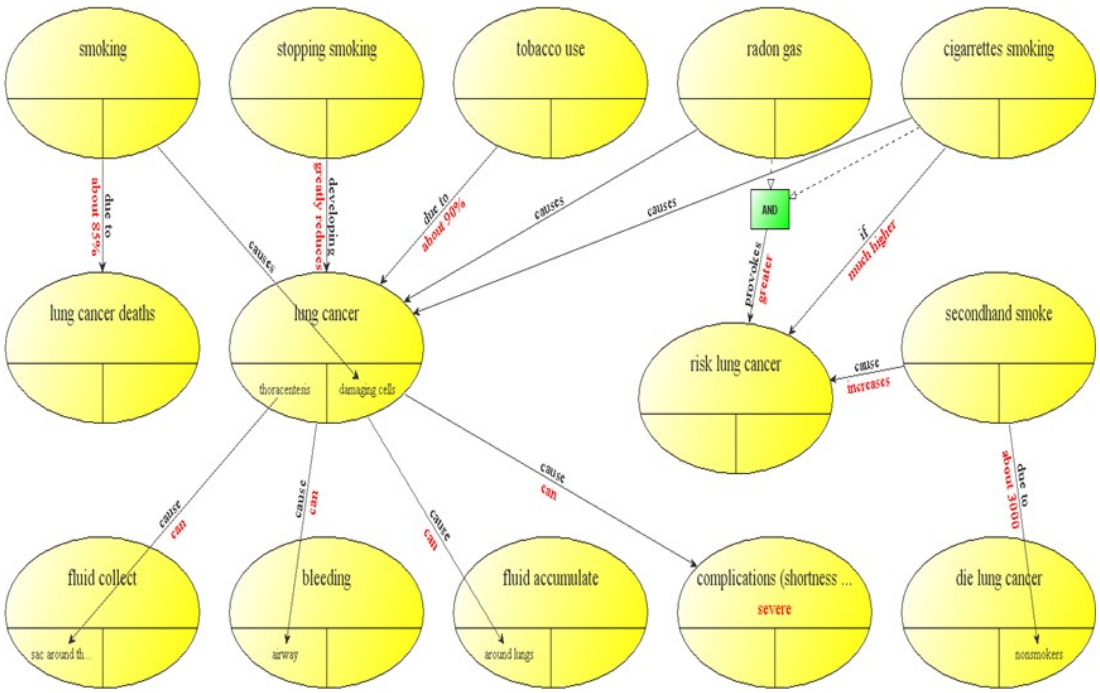


Figura 20: Grafo causal relacionado con «lung cancer» construido de forma automática.

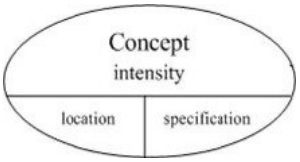


Figura 21: Representación gráfica de un nodo con sus modificadores.

mediante una elipse que contiene una cadena de texto más tres posibles modificadores: ubicación, especificación e intensidad. La representación gráfica de esta estructura puede verse en la figura 21.

Las relaciones entre nodos están representadas por arcos que llevan el tipo de conector causal y, opcionalmente, la intensidad de la relación. La intensidad (o cuantificador), si lleva, estará marcada en rojo y el tipo de conector causal en negro. Una relación está vinculada con un modificador del nodo si la flecha llega hasta dentro de la celda modificadora. De lo contrario, la flecha apunta al borde de aquel.

Entonces, habiendo obtenido del texto un grafo como el de la figura 20, de su lectura pueden obtenerse relaciones ocultas a simple vista entre, por

ejemplo, el nodo «smoking» y algún otro nodo. La figura 22 muestra concretamente cómo se pueden proporcionar automáticamente a partir del grafo, algunos vínculos causales entre factores y efectos que permiten responder la pregunta planteada al comienzo.

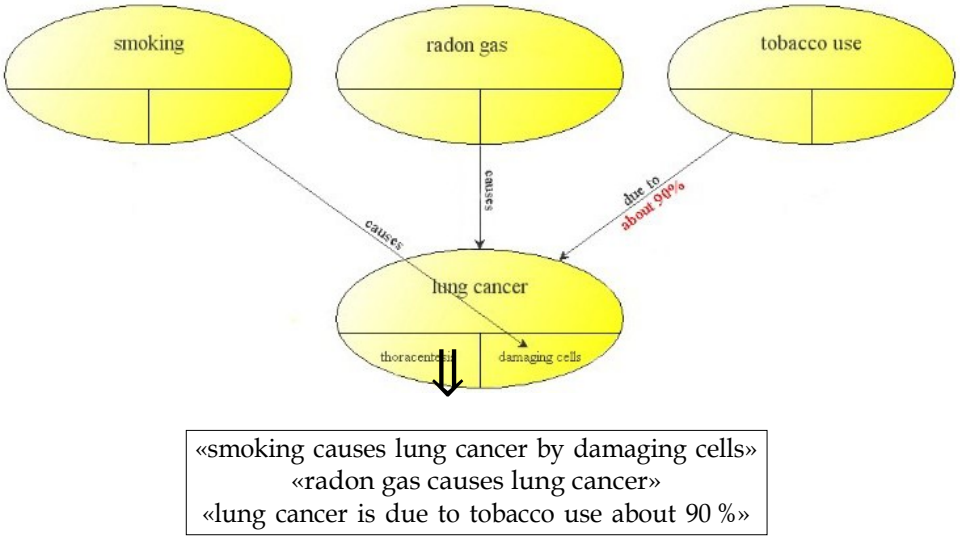


Figura 22: Ejemplo de respuesta leyendo un grafo causal.

4.9. Anotaciones de temporalidad en el grafo

Para introducirle restricciones de tiempo al grafo, se debe buscar por cada nodo aquellas oraciones que tengan modificadores temporales.Las palabras más comunes que indican tiempo son: «after that», «time», «hour», «minute», «day», «before», «now», «today», «tomorrow», «yesterday», «always», «ever», entre otras. El analizador morfológico detectará dichas palabras en las sentencias. De un conjunto de 1214 sentencias causales relacionadas con «lung cancer», se obtuvieron 370 con indicadores de tiempo.

El siguiente paso consiste en poder identificar si el indicador detiempo viene asociado al nodo antecedente o al consecuente. Para ello, una vez detectado el modificador en cuestión, se utiliza el mismo programa que en el paso anterior, donde a partir de la formulación de una pregunta el sistema reconocía si estaba preguntándose por la causao por el efecto. Por ejemplo, en la oración «If, for some reason, surgery is not an option with early stage 1 and 2, treatment is usually radiation therapy» la restricción de tiempo se encuentra modificando la causa. Eneste caso, la tabla 7 indica la salida del POS Tagger de Stanford[16].

root (ROOT-0 , therapy-23)	case (stage-14 , with-12)
mark (1-15 , If-1)	amod (stage-14 , early-13)
case (reason-5 , for-3)	nmod (not-9 , stage-14)
det (reason-5 , some-4)	advcl (therapy-23 , 1-15)
nmod (1-15 , reason-5)	cc (1-15 , and-16)
nsubj (1-15 , surgery-7)	conj (1-15 , 2-17)
cop (1-15 , is-8)	nsubj (therapy-23 , treatment-19)
neg (1-15 , not-9)	cop (therapy-23 , is-20)
det (option-11 , an-10)	advmod (therapy-23 , usually-21)
nmod:npmod (not-9 , option-11)	compound (therapy-23 , radiation-22)

Tabla 7: Salida del POS Tagger de Stanford para la oración «If, for some reason, surgery is not an option with early stage 1 and 2, treatment is usually radiation therapy».

En este ejemplo puede apreciarse que el modificador temporal «early» se encuentra asociado a la palabra «stage». Al introducir todas las palabras que componen la sentencia el programa, etiquetándolas gramaticalmente, es capaz de localizar el modificador y determinar si está afectando al antecedente o consecuente, tal como puede verse en lasiguiente salida de programa:

"> *Line number : 17914 If, for some reason, surgery is not an option with early stage 1 and 2, treatment is usually radiation therapy. Modificator found:*

[16] https://nlp.stanford.edu/software/tagger.shtml

early , context: with early stage 1 and 2. Modificador found: usually , context:is usually radiation therapy"

Una vez localizado en el grafo el nodo que se encuentra afectado por el modificador de tiempo, la restricción debe anotarse. En la sentencia «If you stop smoking before a cancer develops, your damaged lung tissue gradually starts to repair itself», la causa debe ocurrir "antes" de que el cáncer comience a desarrollarse. Por lo tanto, la restricción de tiempo está asociada a la causa y la restricción es dejar de fumar "antes"si el efecto se desea lograr. La figura 23 representa dicha restricción en una línea de tiempo.

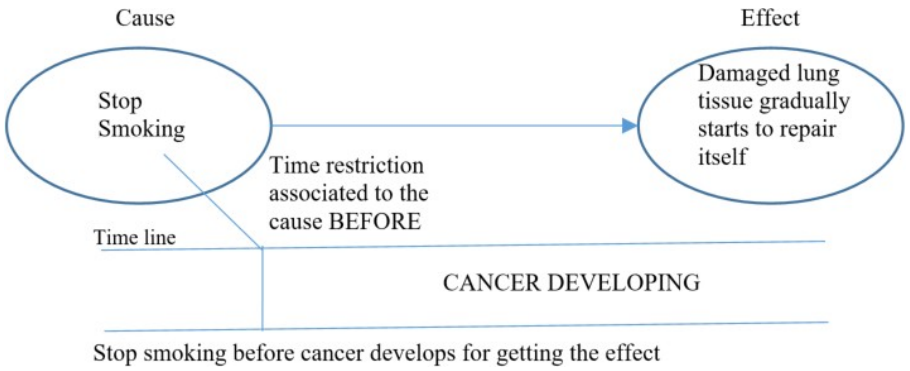


Figura 23: Causa según restricción temporal «before» representada en línea de tiempo.

Cuando los modificadores de tiempo son difusos, las restricciones temporales se representan mediante funciones de pertenencia difusas triangulares o trapezoidales (véase la sección 4.6). Por ejemplo, en la oración «Some lung cancers are found early by accident as a result of tests for other medical conditions» el modificador «early» afectaal efecto y denota tiempo difuso. Como puede verse en la figura 24,su representación mediante una línea decreciente ilustra que «early» registra valores cada vez más bajos a medida que se aleja del principio.

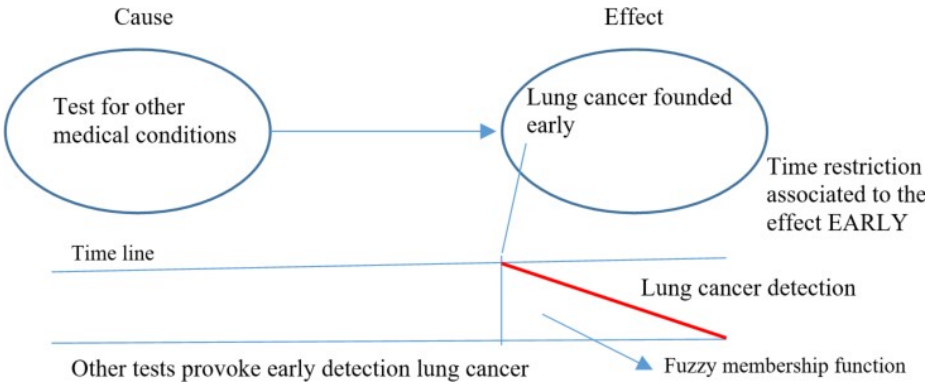


Figura 24: Efecto según restricción temporal difusa «early» representado en línea de tiempo.

Entonces, a partir de estos esquemas se analiza cuál de ellos le corresponde a cada modificador de tiempo encontrado en una oración. El siguiente paso consiste en modificar el grafo causal original con este tipo de representación del tiempo. Para ello, se realiza la búsqueda de aquellas oraciones causales que se encuentren relacionadas con un nodo que esté marcado por un modificador de tiempo. Por ejemplo, las sentencias relacionadas con «smoking» serían las siguientes: (I) «If you stop smoking **before** a cancer develops your damaged lung tissue gradually starts to repair itself», (II) «Although decades have passed since the link between smoking and lung cancers became clear smoking is **still** responsible for most lung cancer deaths» y (III) «Even if you have already been diagnosed with lung cancer there are **still** benefits to stopping smoking». Para completar el grafo, se utilizará la celda inferior derecha del nodo. En dicho espacio, si corresponde, se establece la restricción de tiempo indicando la o las palabras que la denotan. En cuanto a las probabilidades y cuantificadores difusos, se indican en la parte inferior de la flecha de la relación.

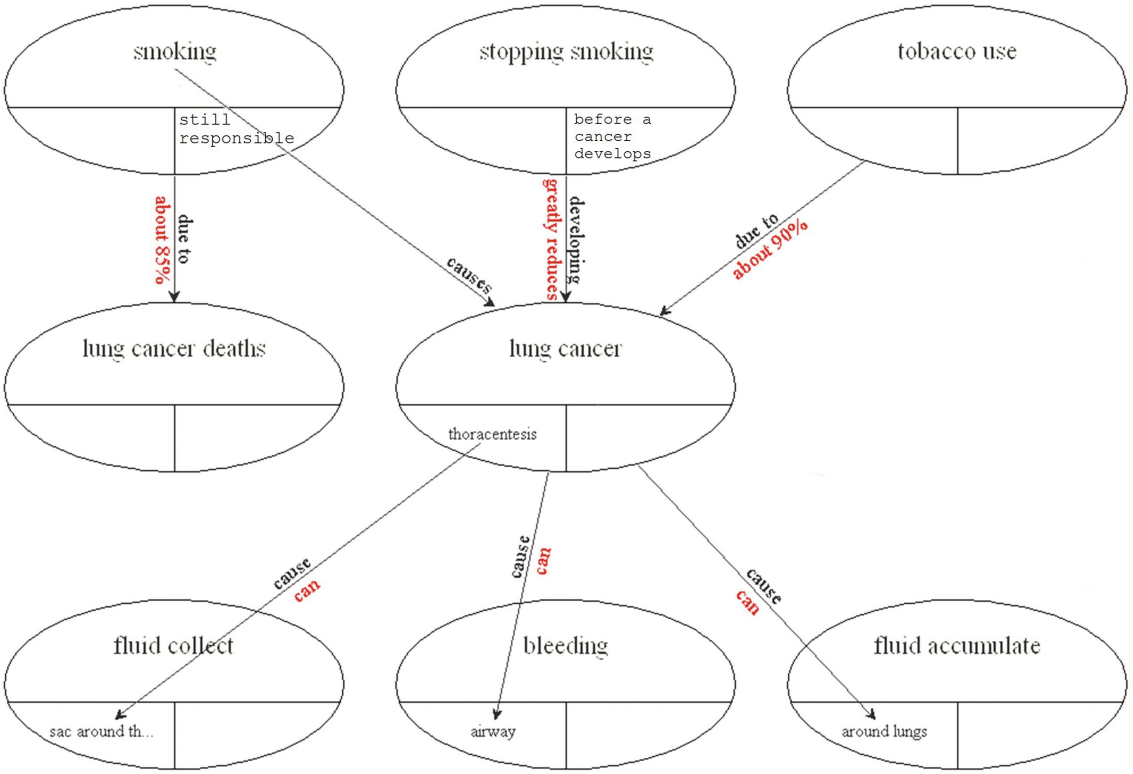


Figura 25: Grafo causal obtenido a partir de restricciones temporales.

Finalmente, como resultado del procedimiento descrito anteriormente, se obtiene un grafo causal como el presentado enla figura 25. En el grafo pueden observarse los nodos con restricción temporal según lo indicado por las oraciones que fueron encontradas relacionándose con el concepto en cuestión. En el caso que un nodo tuviese oraciones relacionadas con diferentes modificadores de tiempo,prevalecerá el más fuerte. En el ejemplo, los modificadores de tiempo encontrados relacionados con el "hábito de fumar" («smoking») son «still» y «before». Sin embargo, en el caso particular de "dejar de fumar"(«stopping smoking») pareciera que «before» se impone por sobre «still» y por eso únicamente se lo representará en el nodo en cuestión.

4.10. Un enfoque práctico sobre el tema

Encontrar aplicaciones concretas a modelos teóricos no es una tarea sencilla. A partir del grafo presentado en la sección anterior, puede generarse un resumen que incluya información causal con componentes temporales. A continuación se presentan los pasos principales que involucra este desafío.

La primera tarea consiste en "limpiar" el grafo quitando todos los nodos redundantes. Son eliminados aquellos nodos cuyos términos poseen una relación lingüística de sinonimia, hiperonimia o meronimia.Ese es el caso, por ejemplo, de «smoking» y «tobacco use». Para realizaresta tarea, se aplica un proceso de limpieza que lee los conceptos del grafo almacenados en una base de datos y a través de una ontología como *WordNet*[17] obtiene diferentes grados de similitud entre los términos. En este punto, además de *WordNet* es útil utilizar recursosespecíficos del dominio como puede ser UMLS[18] en este caso.

Encontrar palabras polisémicas (aquellas con múltiples significados) es otro de los problemas que se debe enfrentar. Por ejemplo, en un contexto de cirugía, «bleeder» puede significar

- el sangrado de un vaso sanguíneo

  «Emergency panendoscopy revealed no active bleeder in the visiblefield, but a lot of fresh blood was gushing from the afferent loop of the jejunu»

- estar relacionado con una persona que sufre hemofilia.

[17] https://wordnet.princeton.edu/
[18] https://www.nlm.nih.gov/research/umls/

«Bleeder is a hemophiliac person who bleeds freely or is subject to frequent hemorrhages»

Para generar un resumen consistente o la respuesta a una preguntaes fundamental manejar en el grafo adecuadamente la polisemiade palabras eligiendo el sentido correcto. Para ello, se recopilan los sentidos de cada una de las palabras del grafo y el texto original donde aparecían estos conceptos. Luego, un programa se encarga de encontrar el significado de estos términos, así como el grado de similitud de los conceptos comparados. Para calcular su similitud, se utiliza *WordNet::Similarity*, que proporciona medidas de similitud comola longitud de la ruta, (Leacock and Chodorow, 1998; o Wu and Palmer, 1994, entre otros).

En la figura 26, se muestra el resultado del proceso de limpieza. A partir de dicha salida los conceptos que son redundantes se eliminan según el grado de similitud pero conservando los factores temporales que podrían llegar a ser relevantes. Los nodos con restricciones de tiempo se consideran "especiales", por lo que permanecen en el grafofinal.

Como se describe en Puente *et al*. (2013b), el grafo obtenido se puede interpretar como un resumen si se sigue el proceso de construcción de lafigura 19 hasta el final. La principal diferencia con Puente *et al*. (2013b),en este caso, es la consideración del tiempo. En la figura 27 se ilustra cómo materializar este aspecto a través de un ejemplo.

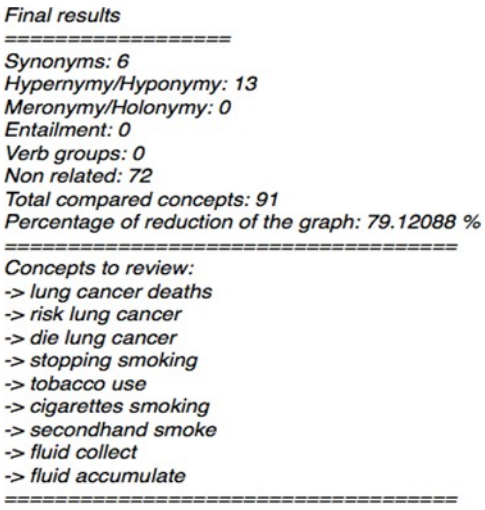


Figura 26: Relaciones de similitud entre conceptos obtenidos por el proceso de filtrado y limpieza.

El proceso descrito a lo largo de las secciones anteriores detectaque la restricción temporal «before» está asociada a la causa «stopping smoking», y luego inserta en su nodo la leyenda «before a cancer develops». Dicha leyenda se interpreta como un prerrequisito temporalque el nodo correspondiente, en este caso a la causa, debe verificarpara poder ejercer su influencia sobre el efecto. Si el requisito previo nose cumple, el otro nodo causal, afectado por la componente temporal «still», se dirige sin restricción negativa hacia «lung cancer». Este ejemplo demuestra que no resulta ser una tarea trivial incorporar al grafo los indicadores de tiempo, ya que la función que cumple cada uno de ellos puede verse afectada por otros nodos que interactúan entre sí. El manejo en estos casos es crucial para construir un grafo que permita generar en un paso siguiente texto significativo.

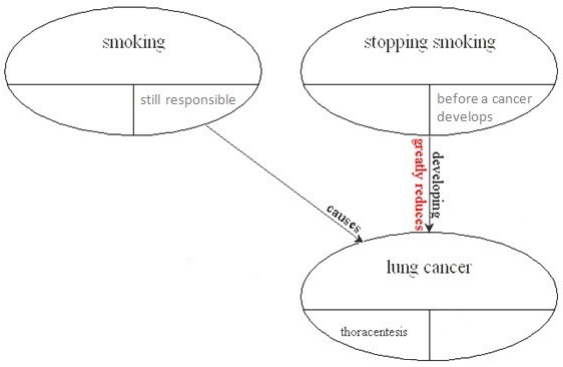


Figura 27: Relaciones causales limitadas por restricciones temporales.

4.11. Sistema de alerta para una adecuada administración de medicamentos

Como se dijo anteriormente, la causalidad y el tiempo tienen una relevancia extraordinaria en medicina. Es poco probable que durante una visita médica no se hagan menciones causales y, más aún, referencias temporales. Los registros e informes médicos están llenos de este tipo de léxico. Frases tales como «Fever started two days ago», o «Antibiotics should be taken after meals» son algunos ejemplos, siendo «two days ago» y «after» indicadores temporales.

Aunque existen sistemas que recuperan sentencias causales de los textos, no hay muchos que organicen dichas sentencias para resolver un problema concreto y, mucho menos, haciendo uso de información temporal.

Los diagnósticos de enfermedades o la prescripción de medicamentos frecuentemente implican dependencias temporales. En el caso de una enfermedad, generalmente, depende de cuándo apareció el síntoma y, en el de los medicamentos, su uso está limitado a ciertas condiciones temporales.

Cada año, la salud de los pacientes se ve perjudicada por la mala administración de medicamentos. Existen estudios que evidencian un alto porcentaje de medicamentos mal suministrados en cuidados intensivos (Keers *et al*., 2013). Dado que la correcta administración de medicamentos depende del tiempo y, además, se encuentra a cargode seres humanos (enfermeras generalmente), desarrollar sistemas de alerta puede evitar errores, contribuir con la seguridad del paciente y mejorar la gestión de los hospitales.

Algunos medicamentos, por ejemplo, deben tomarse antes o después de las comidas, siendo "antes" y "después" restricciones temporales. Para contribuir al manejo del tiempo en tratamientos médicos que deben tener en cuenta el tiempo se propone el desarrollo de una aplicación llamada "My Medicine", que permite controlar las restricciones temporales indicadas por los médicos al prescribirtratamientos para ciertas enfermedades.

La solución a este problema involucra personas de todas las edades, pero, en especial, aquellas que son mayores y que generalmente sufren pérdida de la memoria. Por esa razón, la aplicación del sistema posee una interfaz sencilla, con pocos componentes para que sea fácil de entender y utilizar.

En primer lugar, el sistema debe tener catalogados todos los medicamentos que controlará. De acuerdo con el ISMP, hay varios medicamentos cuya administración debe realizarse cuidadosamente, yasea porque: (I) el tiempo es clave para obtener un efecto farmacológicoóptimo, (II) algunas drogas requieren administrarse durante ciclos

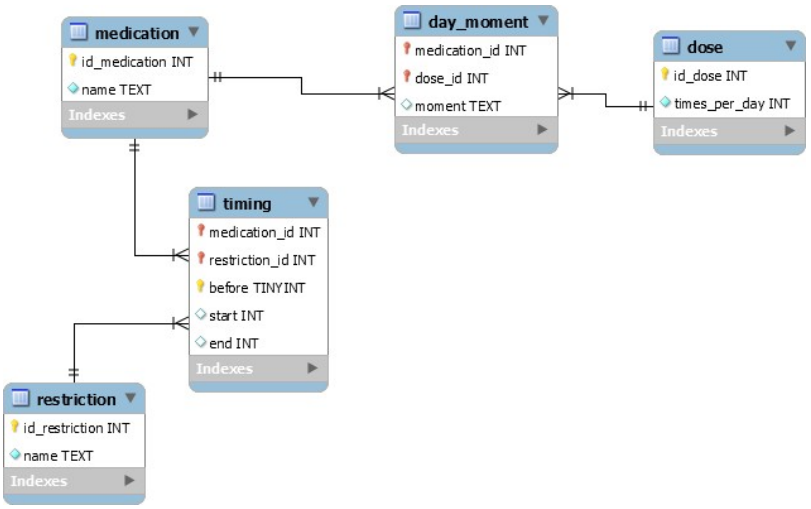


Figura 28: Base de datos utilizada para la administración oportuna de medicamentos.

repetitivos o con cierta frecuencia, o (III) la primera dosis y la de carga deben administrarse de acuerdo con un cronograma. El software detectará los principios activos de cada uno de los medicamentos sensibles al tiempo utilizando un analizador. Una vez detectados los principios activos de cada medicamento, se almacenan en una base de datos registrando además las interacciones con otros medicamentos. Lainformación almacenada a través de la aplicación permite indicar el momento adecuado en el que un medicamento debe tomarse y si este posee interacción con algún otro.

La figura 28 muestra un diseño sencillo de base de datos en el que se almacenan las reglas de gestión del suministro de medicamentos. Aquellos que requieren un momento de administración exacto son de particular interés. Este modelo consta de cinco tablas que almacenan las restricciones necesarias para administrar oportunamente un medicamento programado. En particular, de un medicamento dado se almacena su dosis y restricciones, teniendo en cuenta el momento del día y el tiempo de espera (tablas "day_moment" y "timing"respectivamente).

La estructura de la base de datos se pensó para lograr registrar toda la información suministrada por el ISMP a través de un documento en elque detalla la administración oportuna de medicamentos programados[19]. Una vez analizada en detalle la lista de medicamentos y cargada la información en la base de datos, pueden obtenerse las reglas de controlpara la administración de cada medicamento. Es imprescindible que el proceso de carga se lleve a cabo cuidadosamente.

La información de la base de datos se completa utilizando un analizador morfológico y sintáctico que permite introducir ciertos patrones de búsqueda y recuperar los principios activos de los medicamentos en sitios web. Utilizando, por ejemplo el sitio web dela Clínica Mayo[20], pueden localizarse los principios activos de cada medicamento y comprobarse si es dependiente del tiempo. Una vez localizado un medicamento crítico, se detectan posibles conflictos en el tiempo con otros medicamentos. Estas restricciones son almacenadas en la tabla "restriction" asociada a cada medicamento. Una vezregistrada en la base de datos toda la información necesaria, la interfazde aplicación permitirá controlar los tiempos, como se explica a continuación. La figura 29 sintetiza el proceso descrito completo.

La interfaz de aplicación muestra las restricciones y advertenciasde una manera fácil de interpretar. Primero le hace al usuario una serie de preguntas simples que comienzan con cuál es el nombre del medicamento a tomar (alguno de los que se introdujeron en la base de datos).

[19] https://www.ihs.gov/bcma/includes/themes/responsive2017/display_objects/documents/resources/BCMA_TimelyAdminPolicySample.pdf
[20] https://www.mayoclinic.org/

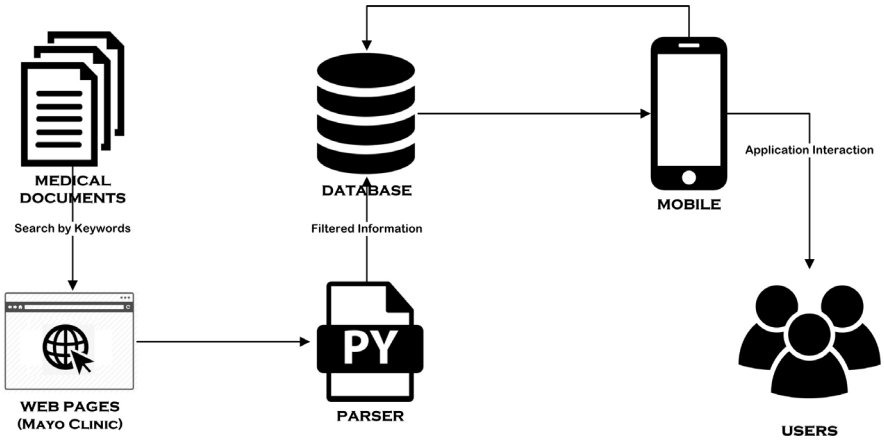


Figura 29: Proceso de control para administrar dosis de medicamentos críticos a lo largo del tiempo.

Teniendo en cuenta el tipo de medicamento, el sistema continúacon otras preguntas en relación con su administración. Por ejemplo, si el medicamento debe tomarse 30 minutos antes de la primera comida, el sistema le preguntará la hora en que generalmente se levanta para poder programar una alarma 30 minutos antes.

Por otra parte, si el usuario tiene que tomar más de un medicamento, la aplicación debe detectar y controlar los posibles conflictos a lo largo del tiempo. Por eso, también le solicita la cantidad de medicamentos a buscar en la base de datos, a fin de detectar posibles incompatibilidadesentre ellos.

Una vez introducida toda la información de configuración, la aplicación notificará al usuario cuando deba tomar el medicamentode acuerdo con las restricciones introducidas. Al mismo tiempo, se mostrará una especie de semáforo con el color adecuado en cada caso para indicar si puede o no tomar el medicamento según la hora que sea.

A continuación, se mostrará un ejemplo completo para lograr terminar de explicar el funcionamiento de la aplicación.

Una de las enfermedades cuyo tratamiento es especialmente sensible al control del tiempo es la deficiencia de hormona tiroidea, tratadacon el principio activo L-tiroxina. Según las restricciones recogidas enla base de datos, el medicamento debe tomarse media hora antes del desayuno. Por otro lado, los suplementos de calcio pueden interferir con el tratamiento del hipotiroidismo según la Clínica Mayo[21]. Esta es una restricción especial que debe ser controlada ya que afecta la absorción de los medicamentos que reemplazan la hormona tiroidea. Según aquelsitio web, hay dos restricciones relacionadas con el calcio que deben considerarse en la aplicación:

- "Don't take calcium supplements or antacids at the same time youtake thyroid hormone replacement"
- "Take any products containing calcium at least four hours before orafter taking thyroid hormone replacement"

Estas dos limitaciones se transforman en nuestra base de datos como "elcalcio no puede administrarse con L-Thyroxine cuatro horas antes".

A continuación se muestra una serie de capturas de pantalla dela aplicación. En la primera (véase la figura 30) hay dos botones:uno para que el usuario indique si desea introducir información("CONFIGURATION PANEL") y otro para verificar si puede ingerir

[21] https://www.mayoclinic.org/diseases-conditions/hypothyroidism/expert-answers/hypothyroidism/ faq-20058536

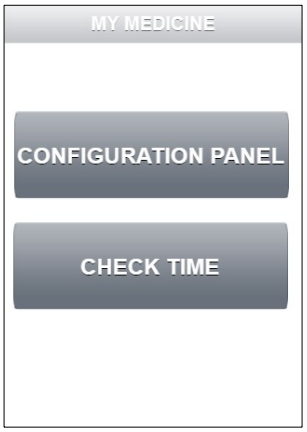


Figura 30: Pantalla inicial de la aplicación “My Medicine”.

(a) Primera medicina (b) Segunda medicina

Figura 31: Pantallas del panel de configuración.

algún medicamento según la información ya ingresada (“CHECKTIME”).

Una vez seleccionado el panel de configuración, el usuario introduce el número de medicamentos que está tomando y cada uno de los nombres. Luego, se le hacen preguntas sobre las restricciones de tiempo,como puede verse en la figura 31 para el caso de L-Thyroxine.

Asociado con el primer medicamento, L-tiroxina, existe una restricción temporal y, antes de programar la alarma, el sistema le solicitará al usuario el horario en el que se despertará. Una vez queel usuario presiona el botón “NEXT”, se muestra otra pantalla en la cual se solicita el siguiente medicamento. En este ejemplo, comoel calcio no tiene preguntas asociadas

para hacerle al usuario, se informa únicamente que existe una restricción de tiempo que debeser controlada. Una vez que el usuario ha terminado de ingresar los medicamentos, hace clic en el botón “FINISH” para guardar los datos introducidos.

En este ejemplo, si el usuario ha introducido “9.30” como hora de despertarse, el sistema programará una alarma media hora antes para alertar que tiene que tomar la L-tiroxina, como se ve en la figura 32a. También, se muestra un semáforo con el color adecuado para indicarle al usuario si puede tomar, en el momento actual, otros medicamentos o no. En este caso, el mensaje aparece en verde, lo que indica que aún no hay conflicto con otros medicamentos.

Si el usuario toma su medicamento y comprueba el sistema a las “10.00”, se mostrará un mensaje que indica la existencia de un conflicto grave con el segundo medicamento introducido (en este caso, Calcium). Por lo tanto, el semáforo se mostrará en rojo advirtiendo que aún no puede tomar el medicamento (véase la figura 32b).

En este mismo ejemplo, si el usuario comprueba la aplicación a las “13:00”, el sistema indicará que el tiempo de conflicto entre los medicamentos está por terminar, mostrándose el semáforo en amarillo(véase la figura 32c). En el caso de largos períodos de tiempo entre la administración de varios medicamentos, como el de cuatro horas como en este caso, el semáforo amarillo aparecerá 30 minutos antes de que seacabe el tiempo.

Figura 32: Pantallas de restricciones temporal de la aplicación.

Una vez expirado el tiempo en el que los medicamentos seencontraban en conflicto (debiendo ocurrir a las “13.30” en este ejemplo), el sistema lanzará otra alarma para indicarle al usuario que yapuede tomar el calcio, poniendo en verde el semáforo (véase la figura 32d).

Con esta sencilla aplicación, el usuario puede controlar a lo largo del día los medicamentos que debe tomar.

4.12. Conclusiones

La causalidad juega un papel fundamental en todos los campos de la ciencia. Es una manera de generar conocimiento y proporcionar explicaciones a través de relaciones entre dos tipos de entidades: causay efecto. La causalidad puede ayudar a mejorar los sistemas de IR proporcionando nuevas relaciones entre conceptos no tratadas con anterioridad.

En medicina, las expresiones causales se encuentran frecuentemente afectadas por limitaciones temporales. Diseñar un sistema que extraiga e interprete sentencias causales capturando elementos temporales en documentos médicos tiene mucha relevancia.

La detección de relaciones causales en los textos es uno de los grandes desafíos del NLP. No existen muchos intentos por extraer mecanismos causales (oraciones encadenadas por relaciones causales) incluyéndoles, además, referencias temporales.

A lo largo del capítulo fue descrito en forma completa el proceso capaz de representar sentencias causales en forma de grafo.El proceso comenzó con la detección, clasificación y análisis de sentencias condicionales y causales a través de ciertos patrones textuales sumamente útiles para la toma de

decisiones. Estos patrones constituyen la información necesaria a partir de la cual se construyeel grafo, al que luego se le incorporan las restricciones de tiempo encontradas.

El grafo es una representación visual que permite captar deforma intuitiva relaciones entre conceptos y también hacer inferencias fácilmente, sin la necesidad de analizar de forma manual los textos originales. A partir de la lectura del grafo construido se pueden automáticamente responder preguntas (Puente *et al*., 2013a) o devolverun resumen (Puente *et al*., 2013b).

En este capítulo, las expresiones causales y los componentestemporales han sido estudiados desde un punto de vista meramente teórico hasta llegar a una solución práctica. En este caso, la aplicación presentada pretende ayudar al paciente durante la administración de medicamentos de acuerdo con las restricciones temporales indicadas por su médico.

CAPÍTULO 5

Conclusiones finales y futuras líneas de trabajo

A lo largo de los capítulos anteriores se han descipto, definidoy utilizado una gran cantidad de temas con el objetivo de analizar información en formato texto. Si bien el foco estuvo puesto en la generación de resúmenes automáticos, se han presentado otras áreasde aplicación. En este capítulo, se hará un recorrido por toda la tesis mencionando lo hecho en cada capítulo, pudiéndose apreciar el grado de cumplimiento de los objetivos establecidos en el capítulo 1, en cada caso. Además, serán presentadas algunas ideas que podrían utilizarse en el futuro para mejorar los métodos aquí propuestos o para aplicar dichos métodos en otras áreas del conocimiento.

5.1. Conclusiones generales

A raíz del rápido aumento en la cantidad de datos generados en formatotexto (como resultado del uso excesivo, disponibilidad y sofisticación de la tecnología), la extracción de conocimiento a partir de informacióntextual se volvió un tema de sumo interés en la actualidad. A medida que crece el volumen de información disponible, su administración se convierte en un problema casi imposible de resolver de forma manualy un problema bastante complejo de resolver en forma automática.

En la actual era de la información, si bien es cierto que resulta imprescindible manejar la mayor cantidad de información posible, no es cierto que toda la información consumida tenga el mismo nivel de relevancia. Esto justifica la necesidad de disponer de sistemas automáticos de síntesis, especialmente en aquellas áreas de la ciencia en las cuales la investigación y divulgación de la información son fundamentales para su desarrollo. En particular en la medicina, por ejemplo, es una práctica habitual condensar información para tomar buenas decisiones.

A la hora de extraer información textual relevante el resumen es clave.Contra lo que se pueda llegar a creer, su construcción no es una tarea sencilla para el ser humano y mucho menos para las computadoras. Después de más de 50 años de investigación alrededor del tema, no existe un único algoritmo que resuma texto. Incluso, no existe el que lo haga de la mejor manera cualquier sea el contexto. Este objetivo está lejos de ser alcanzado. No obstante, la investigación de nuevos algoritmos no ha cesado.

Los usuarios solicitan continuamente sistemas automáticos que produzcan resúmenes personalizados de calidad y que les sirvan para procesar grandes volúmenes de documentos heterogéneos. Cumplircon esta expectativa es un gran desafío. Básicamente, la obtención del resumen automático requiere, por un lado, jerarquizar el contenido de un texto y, por otro, expresarlo de algún modo. Ambas subtareas han sido abordadas en esta tesis a través del desarrollo de dos soluciones muy diferentes pero con varios aspectos en común.

5.2. Representación de la información

A diferencia del conocido proceso de KDD, en el KDT, siendo texto la fuente de información, la etapa de preprocesamiento de la información es aún más *ad-hoc* que cuando se trabaja con información estructurada. Es con vista al resultado final a obtener que se realiza dicha etapa y es el modelo a construir el que condiciona las transformaciones que obligatoriamente deben realizarse sobre los datos de entrada.

Las distintas etapas del proceso no deben recorrerse secuencialmente y en forma aislada. Se requiere tener una visión integral del problema a resolver para seleccionar las técnicas que permitirán representar adecuadamente los documentos de partida. En cuanto a la obtención de resúmenes, no es muy diferente al resto de los problemas de TM. Habiendo dos grandes enfoques, extractivo y abstractivo, sin importar cuál se haya elegido, existe una estrecha relación entre las etapas del KDT. Antes de iniciar el proceso debe conocerse al detalle la naturaleza del tipo de resumen que se espera obtener y obrar en consecuencia.

5.3. Extracción de contenido relevante

El resumen automático de texto ha recorrido un largo camino a partir del trabajo inicial de Luhn en 1958. Desde entonces, muchos enfoques han sido desarrollados con el objetivo de simular el desempeño humano en la construcción de resúmenes.

Sin reparar en los procesos cognitivos que el ser humano realizaal extraer información a partir de los textos, se disparan mecanismos que consisten en

comprender y producir lenguaje natural. A pesar de haberse avanzado en la creación de base de datos léxicas delas cuales pueden obtenerse relaciones semánticas entre palabras,la interpretación del texto continúa siendo uno de los principales problemas que atraviesa el resumen automático. Los métodos basados en extracción evitan el difícil problema de hacer que una computadora entienda el significado de un texto. La extracción se realiza mediante métodos automáticos de análisis estructural del texto (ya sea superficial, intermedio o profundo) que permiten seleccionar de la información original lo considerado relevante para generar el resumen.

En este sentido, se desarrolló un método capaz de identificar el criterio utilizado por el usuario al seleccionar las partes principales de un documento. Esto fue logrado por medio de una variante de PSO desarrollada para que, a partir de los documentos representadosa través de un conjunto de métricas de puntuación, encuentreautomáticamente las métricas a utilizar y su ponderación.

No obstante, el criterio final depende de las métricas utilizadas en la representación de los documentos (que, a su vez, dependen del preprocesamiento realizado), ya que el resultado no es más quela correcta combinación de un subconjunto de las mismas. Como trabajo futuro, se espera ampliar el conjunto de métricas utilizadas para caracterizar los documentos de entrada y así lograr enriquecersu representación. No se descarta el desarrollo de variantes de estas mismas métricas. Además, el tamaño de los resúmenes generados con este método fue establecido de antemano, por lo que las pruebas deberán ampliarse en el futuro para evaluar diferentes valores.

Por otro lado, a futuro se desea incorporar al método propuesto conceptos de lógica difusa de Zadeh (1965) que permitan flexibilizar el criterio del usuario obtenido. El método desarrollado lleva a cabo la combinación de criterios individuales (el de cada métrica) para conseguir un criterio global y único que produzca el ordenamiento del contenido del documento según la relevancia. Dicha combinación no está basada en criterios borrosos, tampoco la valoración de la cada una de las métricas y, ni siquiera, el etiquetado de los documentos de entrenamiento por parte del usuario se lo hace de aquel modo. Todos son valores exactos. Las métricas podrían responder a diferentes lógicas borrosas, las cuales el método se encargaría de combinar para establecer una única lista ordenada del contenido. Así, cada parte del documento tendría varios valores borrosos según cada una de las métricas utilizadas en la representación. Entonces, sería necesario agregar todos los valores borrosos en uno solo utilizando, por ejemplo, operadores OWA de Yager (1988) o alguno derivado de ellos que se ajuste al problema.

5.4. Resumen utilizando grafos

En general, los sistemas de resumen extractivo utilizan el VSM en la representación de los documentos. No obstante, también se utiliza el descubrimiento de ciertas estructuras en los textos que permiten construir representaciones gráficas tipo mapa, en las cuales visualmente se exponen las relaciones entre conceptos del documento que son significativos. Si bien este enfoque está más cerca de la abstracción, no deja de requerir, como primer paso, al menos, la extracción de cierto contenido del documento. Además, necesita no solo tener en cuenta el contenido y estructura del texto, sino

también manejar un buen conocimiento del dominio específico y conocer los intereses personales.

En este sentido, se desarrolló un sistema capaz de extraer el contenido causal y condicional de documentos de texto para crear una base de datos causal relacionada con un tema determinado. Luego, a partir de dicha base, un algoritmo construye un grafo donde se visualizan de forma intuitiva las principales relaciones causales entre los conceptos identificados previamente. A partir de este grafo es posible hacer inferencias. Las sentencias causales en los textos frecuentemente denotan contenido relevante. Por lo general, la causalidad está vinculada a proposiciones o declaraciones generales, que describen un conocimiento sobresaliente sobre un tema determinado. Por lo tanto, aislar e identificar este tipo de contenido es un tema importante.

La causalidad ha sido estudiada, por años, desde muchas perspectivas: filosofía, computación, matemática, etc. Sin embargo, no se encuentran en la literatura muchos casos de aplicación concreta. Por ese motivo, se ha mostrado en esta tesis la relevancia de las sentencias causales influenciadas por limitaciones de tiempo en el área de la medicina. Paraello, se incorporaron anotaciones de temporalidad al grafo y se mostró un ejemplo claro en el que puede transferirse dicho conocimiento al áreade la salud. Se lo hizo a través del diseño de una aplicación que ayuda al paciente a controlar la administración de las dosis de medicamentos indicados por el médico al prescribir el tratamiento. La aplicación, por medio de: (1) la visualización de una especie de semáforo, (2) la programación adecuada de alertas de tiempo, y (3) la identificación de incompatibilidades entre medicamentos, constituye una ayuda efectivapara el paciente.

Como trabajo futuro se abordará la clasificación del léxico temporal para mejorar la representación de las restricciones en el grafo. Por otro lado, se indagará aún más sobre el rol del tiempo en la administración de medicamentos para conseguir la eficacia de éstos teniendo en cuenta la hora del día en que son más eficaces y tolerantes. Se continuará con el diseño de la aplicación, usándola no solo para la administración personal de medicamentos que posean conflictos de tiempo, sino también en la gestión hospitalaria realizada por enfermeros.

5.5. Construcción del grafo usando métricas

Se ha demostrado, en esta tesis, la importancia que posee el contenido causal en la construcción de resúmenes. Sin embargo, el resumen formado por las sentencias causales tal como aparecen en el documentono es mejor que el que puede conseguirse utilizando cualquier métrica extractiva. En Puente *et al*. (2017), se comprobó que, a pesar de que lassentencias causales contienen gran cantidad de información y permiten vincular conceptos, es bastante ambicioso crear un resumen extractivo utilizando solo este tipo de sentencias.

En la sección 4.8 se mostró cómo un grafo causal es creado a partir del texto. Durante aquel procedimiento, las sentencias causales son extraidas y al utilizáselas en la construcción del grafo todas ellas participan por igual, sin tener en cuenta, por ejemplo, su ubicacióndentro del documento.

A través de las métricas que se utilizan en la construcción de resúmenes extractivos, podría conseguirse jerarquizar el contenido causal de un documento y luego permitir pesar los nodos del grafo.De la combinación de enfoques se lograría construir un grafo más preciso y, por lo tanto, obtener

los mejores caminos entre sus nodos constituyendo los antecedentes y los consecuentes de las relaciones causales más relevantes. De esta manera, se podría generar un resumenaún mejor.

5.6. Conclusiones finales

Más de cincuenta años no fueron suficientes para pasar la prueba de Turing (1950) y tampoco lo han sido para escribir un programa que seacapaz de generar resúmenes de calidad humana. Ni siquiera se sabe si será posible alguna vez. Si se quiere producir resúmenes lo más reales posible, en el futuro probablemente se tenga que cambiar totalmente deparadigma.

Hay quienes sostienen que, para muchos de los problemas de TM,el enfoque algorítmico está agotado y que la solución debe basarse en operar sobre la representación del documento. De esta forma, al mejorarla calidad de la representación de la información del documento fuente se podría conseguir mejorar el desempeño de los algoritmos aplicados.

En esta tesis se tuvo por objetivo específico el desarrollo dedos estrategias capaces de resumir documentos de texto en forma automática. El énfasis estuvo puesto en la utilización de dos enfoques diferentes. Por un lado, buscando identificar el criterio del usuario al seleccionar las partes principales de un documento y, por otro, extrayendo de un documento patrones textuales específicos sumamente útiles para la toma de decisiones y representándolos en forma de grafo.

Por todo lo antes expuesto, se considera que los objetivos propuestoshan sido cumplidos satisfactoriamente y, además, quedan definidasnuevas líneas de investigación por donde continuar este trabajo.

Bibliografía

Charu C. Aggarwal. *Data Mining: The Textbook*. Springer Publishing Company, 2015.

James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, 1983.

James F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, 23(2):123 – 154, 1984.

Juan Carlos Augusto. Temporal reasoning for decision support in medicine. *Artificial Intelligence in Medicine*, 33(1):1 – 24, 2005.

Adrian Bardon. *A Brief History of the Philosophy of Time*. OUP USA, 2013.

Phyllis B. Baxendale. Machine-made index for technical literature: An experiment. *IBM Journal of Research and Development*, 2(4):354–361, 1958.

Mónica Bécue-Bertaut. *Minería de textos. Aplicación a preguntas abiertas en encuestas*. Cuadernos de Estadística. La Muralla, 2010.

Harold Borko and Charles L. Berniers. *Abstracting Concepts and Methods*. Academic Press, Inc., 1975.

Max Born. *Natural Philosophy of Cause and Chance*. Oxford Clarendon Press, 1949.

Mario Bunge. *Causality and Modern Science*. Dover Publications, 1979.

Maurice Clerc and James Kennedy. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEETransactions on Evolutionary Computation*, 6(1):58–73, 2002.

Edward T. Cremmins. *The Art of Abstracting*. Info Resources Press, 1996.

Yamille del Valle, Ganesh Kumar Venayagamoorthy, Salman Mohagheghi, Jean-Carlos Hernandez, and Ronald G. Harley. Particleswarm optimization: Basic concepts, variants and applications in power systems. *IEEE Transactions on Evolutionary Computation*, 12(2):171–195, 2008.

Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Didier Dubois, Allel HadjAli, and Henri Prade. Fuzziness and uncertainty in temporal reasoning. *Journal of Universal Computer Science*, 9(9):1168–1194, 2003.

Harold P. Edmundson. New methods in automatic extracting. *Journalof the ACM*, 16(2):264–285, 1969.

Harold P. Edmundson and Ronald E. Wyllys. Automatic abstracting and indexing—survey and recommendations. *Communications of the ACM*, 4(5):226–234, 1961.

Jaoua Kallel Fatma, Jaoua Maher, Belguith Hadrich Lamia, and Ben Hamadou Abdelmajid. Summarization at LARIS Laboratory. In *Proceedings of the Document Understanding Conference*, DUC'04, 2004.

Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth. The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39(11):27–34, 1996.

Ronen Feldman and James Sanger. *Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2006.

Eibe Frank and Ian H. Witten. Generating accurate rule sets without global optimization. In *Proceedings of the Fifteenth International Conference on*

Machine Learning, ICML '98, pages 144–151. Morgan Kaufmann Publishers Inc., 1998.

Alex A. Freitas. A survey of evolutionary algorithms for data mining and knowledge discovery. In *Advances in Evolutionary Computing: Theory and Applications*, pages 819–845. Springer Berlin Heidelberg, 2003.

Jeffrey Friedl. *Mastering Regular Expressions*. O'Reilly & Associates, Inc., 2002.

Douglas J. Futuyma. *Evolutionary Biology*. Sinauer Associates, 2006.

Mahak Gambhir and Vishal Gupta. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, 47 (1):1–66, 2017.

Yihong Gong and Xin Liu. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, pages 19–25, 2001.

Jia-Wen Guo, Sarah Iribarren, Seraphine Kapsandoy, Seneca Perri, and Nancy Staggers. Usability evaluation of an electronic medication administration record (emar) application. *Applied Clinical Informatics*, 2(2):202–224, 2011.

Palak Gupta and Barkha Narang. Role of text mining in business intelligence. *Gian Jyoti E-Journal*, 2(1):123 – 154, 2012.

Vishal Gupta and Gurpreet Singh Lehal. A survey of text summarization extractive techniques. *Journal of emerging technologies in web intelligence*, 2(3):258–268, 2010.

Udo Hahn and Inderjeet Mani. The challenges of automatic summarization. *Computer*, 33(11):29–36, 2000.

José Hernández Orallo, María José Ramírez Quintana, and Cèsar Ferri Ramírez. *Introducción a la minería de datos*. Pearson Educación, 2004.

John H. Holland and Judith S. Reitman. Cognitive systems based on adaptive algorithms. *ACM SIGART Bulletin*, (63):49–49, 1977.

Patricia Jimbo Santana, Augusto Villa Monte, Enzo Rucci, Laura Lanzarini, and Aurelio F. Bariviera. Analysis of methods forgenerating classification rules applicable to credit risk. *Journal of Computer Science & Technology*, 17, 2017.

Clay A. Johnson. *The Information Diet: A Case for Conscious Consumption*. Oreilly and Associate Series. O'Reilly Media, 2011.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing (Third Edition draft)*. Available at https://web.stanford.edu/ ~jurafsky/slp3/, 2018.

Immanuel Kant, Paul Guyer, and Allen W. Wood. *Critique of Pure Reason*. Oeuvre. Cambridge University Press, 1999.

Richard N. Keers, Steven D. Williams, Jonathan Cooke, and Darren M. Ashcroft. Causes of medication administration errors in hospitals:a systematic review of quantitative and qualitative evidence. *Drug safety*, 36(11):1045–1067, 2013.

James Kennedy and Rusell C. Eberhart. Particle swarm optimization.In *Proceedings of the IEEE International Conference on Neural Networks*,pages 1942–1948, 1995.

James Kennedy and Rusell C. Eberhart. A discrete binary version ofthe particle swarm algorithm. In *Proceedings of the IEEE InternationalConference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 5, page 4104–4109, 1997.

James Kennedy and Russell C. Eberhart. *Swarm Intelligence*. Morgan Kaufmann Publishers Inc., 2001.

Elpida T. Keravnou. Temporal reasoning in medicine. *ArtificialIntelligence in Medicine*, 8(3):187 – 191, 1996.

Jaegwon Kim. Causation. In *The Cambridge Dictionary of Philosophy*, pages 125–127. Cambridge University Press, 1995.

Raymond Kosala and Hendrik Blockeel. Web mining research: A survey. *ACM SIGKDD Explorations Newsletter*, 2(1):1–15, 2000.

Laura Lanzarini, Victoria Leza, and Armando De Giusti. Particle swarm optimization with variable population size. In *Artificial Intelligence and Soft Computing – ICAISC 2008*, volume 5097 of *Lecture Notes in Computer Science*, pages 438–449. Springer Berlin Heidelberg, 2008.

Laura Lanzarini, Javier López, Juan Andrés Maulini, and Armando De Giusti. A new binary PSO with velocity control. In *Advances in Swarm Intelligence*, volume 6728 of *Lecture Notes in Computer Science*, pages 111–119. Springer Berlin Heidelberg, 2011a.

Laura Lanzarini, Augusto Villa Monte, Germán Aquino, and Armando De Giusti. Obtaining classification rules using lvqPSO. In *Advancesin Swarm and Computational Intelligence*, volume 9140 of *Lecture Notesin Computer Science*, pages 183–193. Springer International Publishing, 2015a.

Laura Lanzarini, Augusto Villa Monte, Aurelio F. Bariviera, and Patricia Jimbo Santana. Obtaining classification rules using LVQ+PSO: An application to credit risk. In *Scientific Methodsfor the Treatment of Uncertainty in Social Sciences*, volume 377 of*Advances in Intelligent Systems and Computing*, pages 383–391. Springer International Publishing, 2015b.

Laura Lanzarini, Augusto Villa Monte, and Franco Ronchetti. SOM+PSO: A novel method to obtain classification rules. *Journal of Computer Science & Technology*, 15, 2015c.

Laura Lanzarini, Augusto Villa Monte, Aurelio F. Bariviera, and Patricia Jimbo Santana. Simplifying credit scoring rules using LVQ+PSO. *Kybernetes*, 46(1):8–16, 2017.

Laura Cristina Lanzarini, Augusto Villa Monte, and César Armando Estrebou. E-mail processing with fuzzy SOMs and association rules. *Journal of Computer Science and Technology*, 11(01):41–46, 2011b.

Joel Larocca Neto, Alexandre D. Santos, Celso A. A. Kaestner, and Alex A. Freitas. Generating text summaries through the relative importance of topics. In *Advances in Artificial Intelligence*, volume 1952 of *Lecture Notes in Computer Science*, pages 300–309. Springer Berlin Heidelberg, 2000.

Claudia Leacock and Martin Chodorow. Combining Local Context and WordNet Similarity for Word Sense Identification. In *WordNet: An electronic lexical database*, chapter 13, pages 265–283. MIT Press, 1998.

Ludovic Lebart, André Salem, and Mónica Bécue Bertaut. *Análisis estadístico de textos*. Milenio, 2000.

Quanzhi Li and Yi-Fang Brook Wu. Identifying important concepts from medical documents. *Journal of Biomedical Informatics*, 39(6):668 –679, 2006.

Yanhong Liang and Runhua Tan. A text-mining-based patent analysisin product innovative process. In *Trends in Computer Aided Innovation*,pages 89–96. Springer US, 2007.

Marina Litvak, Mark Last, and Menahem Friedman. A new approach to improving multilingual summarization using a genetic algorithm. In

Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10, pages 927–936, 2010a.

Marina Litvak, Hagay Lipman, Assaf Ben Gur, Mark Last, Slava Kisilevich, and Daniel Keim. Towards multi-lingual summarization:a comparative analysis of sentence extraction methods on english and hebrew corpora. In *Proceedings of the 4th Workshop on Cross Lingual Information Access*, pages 61–69, 2010b.

Elena Lloret and Manuel Palomar. Text summarisation in progress: a literature review. *Artificial Intelligence Review*, 37(1):1–41, 2012.

Félix López and Víctor Romero. *Mastering Python Regular Expressions*. Community experience distilled. Packt Publishing, 2014.

Cristina G. Lucio. Cronoterapia: por qué la hora a la que te medicaspuede ser más importante que la dosis. https://www.elmundo.es/papel/historias/2018/05/11/5af43ad2468aeb486e8b466b.html, 2018.

Hans Peter Luhn. The automatic creation of literature abstracts. *IBMJournal of Research and Development*, 2(2):159–165, 1958.

Ernst Mach. *Knowledge and error: sketches on the psychology of enquiry*. Vienna Circle collection. D. Reidel Publishing Company, 1976.

Nicolás Maestropiedra. Carga de datos contextuales para el análisis y gestión del proceso judicial. In *XVIII Simposio Argentino de Informáticay Derecho (SID) - JAIIO 47*, pages 47–56, 2018.

Inderjeet Mani. Summarization evaluation: An overview. In *Proceedingsof the Second NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization*, 2001a.

Inderjeet Mani. *Automatic Summarization*. J. Benjamins Publishing Company, 2001b.

Klaus Martiny, Else Refsgaard, Vibeke Lund, Marianne Lunde, Britta Thougaard, Lone Lindberg, and Per Bech. Maintained superiority of chronotherapeutics vs. exercise in a 20-week randomized follow-up trial in major depression. *Acta Psychiatrica Scandinavica*, 131(6): 446–457, 2015.

Matthew Mayo. A general approach to preprocessing text data. https://www.kdnuggets.com/2017/12/general-approach-preprocessing-text-data.html, 2017.

Matthew Medland, Fernando E. B. Otero, and Alex A. Freitas. Improving the cAnt-MinerPB classification algorithm. In *Proceedingsof the 8th International Conference on Swarm Intelligence*, ANTS'12, pages 73–84. Springer-Verlag, 2012.

Yogesh Kumar Meena and Dinesh Gopalani. Evolutionary algorithms for extractive automatic text summarization. *Procedia Computer Science*, 48:244 – 249, 2015.

Rada Mihalcea. Graph-based ranking algorithms for sentence extraction, applied to text summarization. In *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*, ACLdemo '04, 2004.

John Stuart Mill. *A System of Logic, Ratiocinative and Inductive: Being a Connected View of the Principles of Evidence and the Methods of Scientific Investigation*. John W. Parker, 1843.

Gary Miner, John Elder, Thomas Hill, Robert Nisbet, Dursun Delen, and Andrew Fast. *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*. Academic Press, 2012.

Rashmi Mishra, Jiantao Bian, Marcelo Fiszman, Charlene R. Weir, Siddhartha Jonnalagadda, Javed Mostafa, and Guilherme Del Fiol. Text summarization in

the biomedical domain: A systematic reviewof recent research. *Journal of Biomedical Informatics*, 52:457 – 467, 2014.

Ryan Mitchell. *Web Scraping with Python: Collecting Data from the Modern Web*. O'Reilly Media, 2015.

National Information Standards Organization (NISO). Guidelines for Abstracts. https://www.niso.org/publications/ ansiniso-z3914-1997-r2015-guidelines-abstracts, 2015.

National Institute of Standards and Technology (NIST). Document Understanding Conferences (DUC). http://www-nlpir.nist.gov/ projects/duc/index.html, 2002.

Ani Nenkova and Kathleen McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. Springer, 2012.

Joel Larocca Neto, Alex Alves Freitas, and Celso A. A. Kaestner. Automatic text summarization using a machine learning approach.In *Advances in Artificial Intelligence*, volume 2507 of *Lecture Notes in Computer Science*, pages 205–215. Springer-Verlag, 2002.

Chikashi Nobata, Satoshi Sekine, Masaki Murata, Kiyotaka Uchimoto, Masao Utiyama, and Hitoshi Isahara. Sentence extraction system assembling multiple evidence. In *Proceedings of the Second NTCIR Workshop Meeting*, 2001.

José Á. Olivas. *Búsqueda eficaz de información en la Web*. EDULP, 2011.

Hilário Oliveira, Rafael Ferreira, Rinaldo Lima, Rafael Dueire Lins, Fred Freitas, Marcelo Riss, and Steven J. Simske. Assessing shallow sentence scoring techniques and combinations for single and multi-document summarization. *Expert Systems with Applications*, 65:68 – 86, 2016.

Rosalía Peña, Ricardo Baeza-Yates, and José V. Rodríguez. *Gestión digitalde la información: de bits a bibliotecas digitales y la web*. Ra-Ma, 2002.

María Pinto, María Mitre, Anne-Vinciane Doucet, and María José Sánchez. *Aprendiendo a resumir: prontuario y resolución de casos*. Trea,2005.

Martin F. Porter. An algorithm for suffix stripping. *Program*, 14(3): 130–137, 1980.

Cristina Puente. *Extraction and analysis of conditional and causal sentencesfor information retrieval*. PhD thesis, Universidad Pontificia Comillas, 2010.

Cristina Puente, Alejandro Sobrino, José A. Olivas, and Roberto Merlo. Extraction, analysis and representation of imperfect conditional and causal sentences by means of a semi-automatic process. In *2010 IEEE International Conference on Fuzzy Systems*, pages 1–8, 2010.

Cristina Puente, Eduardo Garrido, and José A. Olivas. Answering questions by means of causal sentences. In *Flexible Query Answering Systems*, volume 8132 of *Lecture Notes in Artificial Intelligence*, pages 91–99. Springer-Verlag, 2013a.

Cristina Puente, José A. Olivas, Eduardo Garrido, and Roberto Seisdedos. Creating a natural language summary from a compressed causal graph. In *2013 Joint IFSA World Congress and NAFIPS AnnualMeeting*, pages 513–518, 2013b.

Cristina Puente, Augusto Villa Monte, Laura Lanzarini, Alejandro Sobrino, and José A. Olivas. Evaluation of causal sentences in automated summaries. In *2017 IEEE International Conference on FuzzySystems*, pages 1–6, 2017.

Cristina Puente, Alejandro Sobrino, Augusto Villa Monte, and José A. Olivas. Alert system for timely medication administration. In *Proceedings of the 2018*

International Conference on Artificial Intelligence(ICAI'18), located at 2018 World Congress in Computer Science, Computer Engineering, & Applied Computing (CSCE'18), pages 387–392. CSREA Press, 2018.

Cristina Puente, Alejandro Sobrino, José A. Olivas, and Augusto Villa Monte. Designing a system to extract and interpret timed causal sentences in medical reports. *Journal of Experimental & Theoretical Artificial Intelligence*, 31(1):1–13, 2019.

John Ross Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., 1993.

Paul Reber. What is the memory capacity of the human brain? https://www.scientificamerican.com/article/ what-is-the-memory-capacity/, 2010.

Keith A. Rodvold and Donna M. Kraus. Drug interactions involving anti-infective agents. In *Antibiotic and Chemotherapy*, pages 68 – 103. W.B. Saunders, 2010.

Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, 2009.

Horacio Saggion and Thierry Poibeau. Automatic text summarization: Past, present and future. In *Multi-source, Multilingual Information Extraction and Summarization*, pages 3–21. Springer Berlin Heidelberg,2013.

Gerard Salton. *The SMART Retrieval System—Experiments in Automatic Document Processing*. Prentice-Hall, Inc., 1971.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513 – 523, 1988.

Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11): 613–620, 1975.

Gerard Salton, Edward A. Fox, and Harry Wu. Extended boolean information retrieval. *Communications of the ACM*, 26(11):1022–1036, 1983.

Roberto Saracco. A never ending decrease of technology cost. http://sites.ieee.org/futuredirections/2017/10/18/ a-never-ending-decrease-of-technology-cost/, 2017.

Susan Schreibman, Ray Siemens, and John Unsworth. *A New Companion to Digital Humanities*. Blackwell Companions to Literature and Culture. Wiley, 2016.

Alejandro Sobrino, Cristina Puente, and José A. Olivas. Extracting answers from causal mechanisms in a medical document. *Neurocomputing*, 135:53 – 60, 2014.

John F. Sowa. Processes and causality. Available at http://www.jfsowa. com/ontology/causal.htm, 2000.

Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, 1972.

Karen Spärck Jones. Automatic summarising: The state of the art. *Information Processing & Management*, 43(6):1449–1481, 2007.

Paul Thagard. Explaining disease: Correlations, causes, and mechanisms. *Minds and Machines*, 8(1):61–78, 1998.

Alvin Toffler. *Future Shock*. Random House, 1970.

Juan Manuel Torres Moreno. *Automatic Text Summarization*. Cognitive science and knowledge management series. Wiley, 2014.

Alan M. Turing. Computing machinery and intelligence. *Mind*, 59(236): 433–460, 1950.

Heleen Van der Sijs, Laureen Lammers, Annemieke van den Tweel, Jos Aarts, Marc Berg, Arnold Vulto, and Teun van Gelder. Time-dependent drug–drug interaction alerts in care provider order entry: Software may inhibit medication error reductions. *Journal of the American Medical Informatics Association*, 16(6):864–868, 2009.

Lucy Vanderwende, Hisami Suzuki, Chris Brockett, and Ani Nenkova. Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618, 2007.

Gilles Venturini. SIA: A supervised inductive algorithm with genetic search for learning attributes based concepts. In *Machine Learning: ECML-93*, volume 667 of *Lecture Notes in Computer Science*, pages 280–296. Springer Berlin Heidelberg, 1993.

Augusto Villa Monte, César Estrebou, and Laura Lanzarini. E-mail processing using data mining techniques. In *Computer Science & Technology Series: XVI Argentine Congress of Computer Science - Selectedpapers*, pages 109–120. Edulp, 2011.

Augusto Villa Monte, Franco Ronchetti, Laura Lanzarini, and Marcela Jeréz. Obtención de reglas de clasificación usando SOM+PSO. In*Proceedings of the XVIII Argentine Congress of Computer Science*, pages210–219, 2012.

Augusto Villa Monte, Laura Lanzarini, Luis Rojas Flores, and José A. Olivas. Document summarization using a scoring-based representation. In *2016 XLII Latin American Computing Conference*, pages 1–7, 2016.

Augusto Villa Monte, Julieta Corvi, Laura Lanzarini, Cristina Puente,Alfredo Simon Cuevas, and José A. Olivas. Text pre-processing toolto increase the

exactness of experimental results in summarization solutions. In *Proceedings of the XXIV Argentine Congress of Computer Science*, 2018a.

Augusto Villa Monte, Laura Lanzarini, Aurelio F. Bariviera, and José A. Olivas. Obtaining and evaluation of extractive summaries from stored text documents. In *Proceedings of the Third Conference on Business Analytics in Finance and Industry*, pages 65–66, 2018b.

Eder Vázquez, René Arnulfo García-Hernández, and Yulia Ledeneva. Sentence features relevance for extractive text summarization using genetic algorithms. *Journal of Intelligent & Fuzzy Systems*, 35(1): 353–365, 2018.

Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, ACL '94, pages 133–138, 1994.

Ronald R. Yager. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190, 1988.

Lotfi Zadeh. Fuzzy sets. *Information and Control*, 8(3):338 – 353, 1965.

Li Zhou and George Hripcsak. Temporal reasoning with medical data—a review with emphasis on medical natural language processing. *Journal of Biomedical Informatics*, 40(2):183 – 202,2007.

ANEXOS

APÉNDICE A

Extracción de reglas de clasificación

En la actualidad, son numerosas las áreas interesadas en extraerconocimiento a partir de la información almacenada. Esta información representa todo lo ocurrido en el pasado y su análisispermite justificar las decisiones que fueron tomadas en su momento.Comprender los criterios utilizados con anterioridad y asociarlos conlos resultados obtenidos permite elegir el camino a seguir ante una nueva situación.

En este anexo se describirá un método nuevo basado en la técnica de optimización desarrollada en el capítulo 3, capaz de construir, a partir de la información disponible, un conjunto de reglas de clasificaciónque resuman e identifiquen adecuadamente las características más relevantes de los datos. El objetivo del método es obtener un modelo sencillo que permita explicar las decisiones tomadas y a la vez posea una tasa de acierto aceptable.

A.1. Introducción

La *minería de datos* es una de las etapas más importantes del KDD. Cuenta con el conjunto de técnicas capaces de modelizar y resumir datos históricos, facilitando su comprensión y ayudando a la toma de decisiones.

Disponer de herramientas capaces de representar de una manera descriptiva las decisiones tomadas resulta sumamente útil para explicar lo hecho y poder decidir lo que es conveniente hacer en el futuro. Cuanto más simple de comprender sea dicha representación descriptiva, más fácil será decidir cuándo y cómo utilizarla para tomar decisiones.

Este anexo trata sobre la obtención automática de proposiciones condicionales con el objetivo de construir una representación con capacidad para explicar la manera en que se encuentra organizada la información histórica e identificar las razones por las cuales se tomaronciertas decisiones.

Para lograr este objetivo, se desarrolló un método basado en la técnica de optimización presentada en el capítulo 3. El método es capaz de construir, a partir de la información disponible, un conjunto de reglas de clasificación con tres características principales: precisión adecuada, baja cardinalidad y simplicidad del antecedente (Lanzarini *et al*., 2015a).De esta forma, se facilita la interpretación del modelo ayudando a la toma de decisiones.

Cuando se habla de reglas de clasificación, se hace referencia a proposiciones expresadas en forma condicional cuya estructura es la siguiente:

«**SI** (ocurre esto) **ENTONCES** (también ocurre aquello otro)»

donde la primera parte corresponde al antecedente y la segundaal consecuente. El antecedente o condición de la regla consiste en la combinación de expresiones "(variable = valor)", cuyas variables pueden ser cualitativas o cuantitativas. Por tratarse de reglas de clasificación, el consecuente siempre utiliza el mismo atributo para referirse a la respuesta esperada (Aggarwal, 2015).

Frente a otro tipo de soluciones, las reglas son las preferidas a la hora de caracterizar esa enorme cantidad de datos históricos que fueron guardados automáticamente. Tienen capacidad de explicarse por sí mismas y de su lectura se obtiene una justificación inmediata de las decisiones sugeridas. Lamentablemente, la mayoría de los métodos existentes cubre la información histórica utilizando un conjunto de reglas generalmente extenso y complejo que, pese a tener la forma «SI-ENTONCES», se torna prácticamente ilegible.

A través de la simplicidad de las reglas obtenidas con el método aquí propuesto se hace frente al problema que generalmente tienen los conjuntos de reglas. Esta simplicidad se consigue a través del uso de un número reducido de atributos en la conformación del antecedente. Esta característica, sumada a la baja cardinalidad del conjunto de reglas, permite distinguir patrones sumamente útiles a la hora de comprenderlas relaciones entre los datos y posteriormente tomar decisiones.

El método propuesto ha sido aplicado sobre varios conjuntosde datos del *UCI Machine Learning Repository* (Dheeru and Karra Taniskidou, 2017) y los resultados han sido comparados con los obtenidos por otros métodos existentes en la literatura. También selo ha utilizado en dos casos reales de

empresas financieras que otorganpréstamos para consumo en el Ecuador (Lanzarini *et al*., 2015b).

A.2. Método propuesto

La obtención de reglas de clasificación es una tarea supervisada que en función de los ejemplos disponibles, busca establecer las mejores condiciones para cubrir de manera adecuada la mayor cantidad de casos. El método desarrollado utiliza la optimización mediante cúmulo de partículas propuesta en el capítulo 3 para construir el conjunto de reglas.

El énfasis del método está puesto en alcanzar una buena cobertura utilizando un número reducido de reglas, donde cada una de ellas posea un número mínimo de conjunciones en su antecedente. De esta forma, se facilitará la interpretación del modelo ayudando en la toma dedecisiones.

A continuación serán detallados los aspectos necesarios para entender cómo se utiliza la técnica de optimización mencionada para extraer reglas.

A.2.1. Información a manejar en cada individuo

Tal como se describió en el capítulo 3, Particle Swarm Optimization (PSO) es una estrategia poblacional donde los individuos buscan mejorar su capacidad de resolver el problema a medida que el proceso evolutivo avanza. Esta capacidad está relacionada con la información que cada partícula contiene obligatoriamente en su estructura.

Con respecto a la extracción de reglas de clasificación, para formar el antecedente se utiliza una representación de longitud fija formada por dos partes:

PA RT E B I NA R I A Permite identificar cuáles son las condiciones que formarán parte del antecedente. En el caso de los atributos cualitativos, utiliza tantos dígitos binarios como valores diferentes presente dicho atributo. En el caso de los atributos numéricos, utiliza solo dos dígitos binarios para indicar si interviene el límite inferior y/o superior de dicho atributo. Su valor será 1 cuando se utiliza y 0 si no.

PA R T E R E A L Contiene los valores que permiten acotar a los atributos numéricos a la hora de conformar la condición. Dichos valores son calculados a través de la técnica de optimización evitando, de esta forma, tener que discretizar los atributos numéricos antes de iniciar el proceso. Su longitud coincide con la parte binaria aunque solo se utiliza para los atributos numéricos. Este aspecto, si bien incrementa levemente la longitud del individuo, facilita el funcionamiento del algoritmo.

Como puede notarse, al involucrar la representación tanto atributos numéricos como nominales, la estructura de la partícula es la misma que la vista en la sección 3.4. La única diferencia tiene que ver con el usoque se le da a cada parte (véase sección 3.5). En este caso, la binaria sirvepara para elegir los atributos que se utilizan en la regla y la continua para delimitar los atributos cuantitativos.

Por otro lado, el método utiliza el cúmulo completo para operarcon una misma clase a la vez. Es decir que todos los individuos de la población corresponden a reglas de una misma clase predeterminada. De esta manera,

cada partícula evoluciona el antecedente de la única regla que representa y su consecuente no está codificado en la estructura (Freitas, 2003).

A.2.2. Inicialización de la población

La ubicación de los individuos en el espacio de búsqueda es un aspecto determinante para lograr buenos resultados. En especial si se debetrabajar con representaciones binarias se corre el riesgo de obtener valores de aptitud muy disímiles, aunque se hayan producido pequeños ajustes en la representación.

Iniciar cerca del óptimo no solo incrementa la calidad de los resultados sino que reduce el tiempo de la búsqueda. Para conseguir una adecuada inicialización, se utiliza un agrupamiento de los datos de entrada a través de una red neuronal competitiva supervisada. Una vezentrenada, los centroides permitirán conocer cuáles son las zonas más prometedoras del espacio de búsqueda facilitando la inicialización del cúmulo.

A.2.3. Aptitud de una partícula

El valor de aptitud de cada partícula se calcula de la siguiente forma:

$$Fitness = \alpha * balance * support * conf\ idence - \beta * antecedent\_length \quad (16)$$

donde $support$ y $conf\ idence$ son medidas conocidas para establecer la calidad de la regla, α y β son constantes que representan la importanciaque se le da a cada término, $balance$ permite compensar el efectoque tiene el desbalance entre clases a la hora de calcular el soporte y $antecedent\_length$ es la

proporción de condiciones utilizadas en el antecedente respecto de la cantidad total de atributos que pueden utilizarse.

La evaluación del desempeño del individuo no se limita solo a una única regla sino que utiliza al individuo binario como la selección de los ítems que pueden formar el antecedente pero no se considera obligatorio utilizarlos a todos. Cada vez que se evalúa la partícula, también se analizan las distintas reglas que se forman al suprimir unaa una las condiciones que participan en el antecedente. Esto simplifica las reglas a medida que las partículas realizan la búsqueda. De todas las reglas evaluadas para una partícula, se toma la de mayor fitness. La regla resultante se almacena en la partícula conservando solo las condicionesrelevantes con valor 1 y el resto con 0.

A.2.4. Desplazamiento de las partículas

Como ya se ha mencionado, para poder operar con atributos cualitativos y cuantitativos, el movimiento de la partícula se encuentra dividido en dos partes: una binaria y otra continua. La primera indica cuáles serán los atributos a utilizar mientras que la segunda determina los intervalos que controlan los atributos numéricos.

Operar con una representación doble implica definir el criterio con el que se realizará el movimiento. Aquí deben hacerse ajustes para poder hallar la solución más adecuada dentro de un entorno reducido. Para mejorar la calidad de la búsqueda, el método realiza una breve expansión al inicio para concentrarse rápidamente en la soluciónesperada (Lanzarini *et al.*, 2008).

De no contar con la posibilidad de trabajar con una población variable, es preciso indicar *a priori* la cantidad de partículas que formarán la población. Si

este valor se encuentra por debajo de lonecesario, no se logrará llegar a una solución adecuada y, por el contrario, si es excesivo, el costo computacional se verá incrementado.

Por otro lado, pequeñas modificaciones realizadas sobre los atributos nominales generan grandes cambios en el valor de aptitud de la partícula provocando que una regla que se estaba formando adecuadamente pase abruptamente a tener un valor de aptitudnulo. Para resolver este aspecto es preciso controlar los movimientos realizados. El método utiliza una variante de PSO definida en (Lanzarini *et al*., 2011a) donde se propone controlar las modificacionesdel vector velocidad.

A.2.5. Proceso de obtención de reglas

El método sigue el enfoque llamado Iterative Rule Learning (IRL), enel cual por cada ejecución de un proceso iterativo se extrae el mejor individuo de la siguiente manera.

El método iterativamente busca cubrir los ejemplos de la clase mayoritaria. En cada ejecución del PSO, las partículas competirán entreellas para formar la mejor regla de la clase seleccionada. Luego de la obtención de cada regla, se retiran del conjunto de entrada los ejemplos correctamente cubiertos y se vuelve a repetir el proceso con los restantes hasta hallar el conjunto de reglas completo. Esto evita generar reglas Similares, pero, a diferencia del enfoque *Michigan* (Holland and Reitman, 1977), las reglas deben aplicarse a los datos en el mismo orden en que fueron generadas (Venturini, 1993). Las reglas forman lo que se conocecomo lista de clasificación.

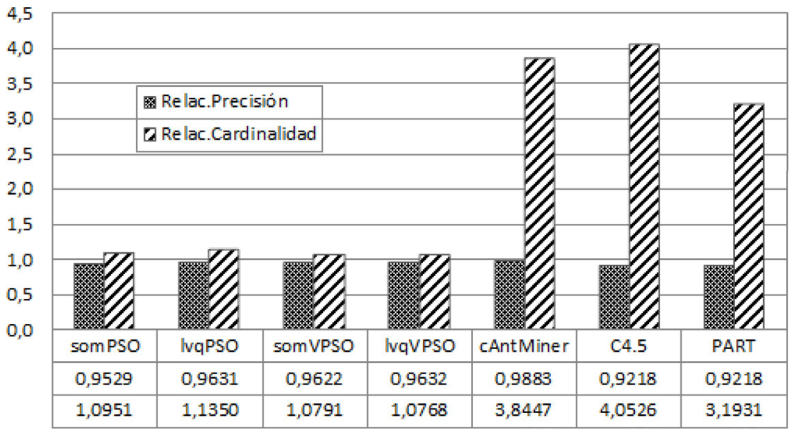
A.3. Resultados obtenidos

La figura 33 ilustra un análisis comparativo de los resultados obtenidos de aplicar cuatro variantes del método descrito en este anexoa 13 bases de datos del repositorio UCI. La inicialización del cúmulo de partículas se realizó de dos formas distintas: con una red SOM y con una red LVQ.

Se trabajó con poblaciones de tamaño fijo utilizando redes neuronales de 30 neuronas competitivas. Para la red SOM se utilizó una grilla de 6x5 con 4 vecinas como máximo por neurona. Estas combinaciones aparecen mencionadas en la figura 33 como "somPSO"y "lvqPSO".

Las versiones con población variable, denominadas *somVPSO* y *lvqVPSO*, utilizan la estrategia de edad para modificar la cantidad de individuos y pueden comenzar con una población menor ya que agregarán o quitarán partículas durante el proceso de búsqueda según consideren adecuado.

Los resultados obtenidos fueron comparados con los arrojados por los siguientes métodos existentes en la literatura: PART (Frank and Witten, 1998), *cAntMinerPB* (Medland *et al.*, 2012) y *C4.5* (Quinlan, 1993).



somPSO	lvqPSO	somVPSO	lvqVPSO	cAntMiner	C4.5	PART
0,9529	0,9631	0,9622	0,9632	0,9883	0,9218	0,9218
1,0951	1,1350	1,0791	1,0768	3,8447	4,0526	3,1931

Figura 33: Resultados obtenidos sobre 13 bases de datos de repositorio.

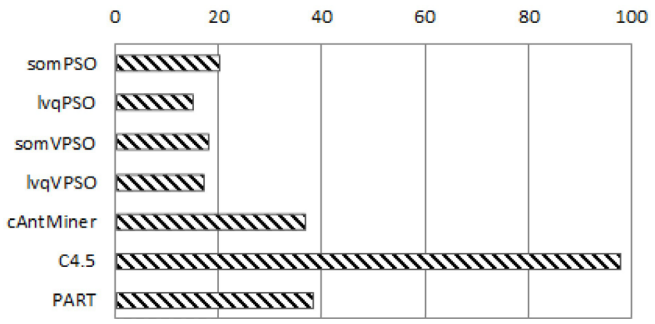


Figura 34: Simplicidad del modelo obtenido medida a partir de la cantidad de comparaciones totales realizadas en cada caso.

Dado que se trata de conjuntos de datos con distintas características,se han normalizado los resultados obtenidos en lo referido a precisión, cardinalidad del conjunto de reglas y longitud promedio por antecedente de una regla. En la figura 33, se denomina "Relac.Precisión"al promedio de los cocientes entre la precisión obtenida por el método y la precisión de la mejor solución para cada base de datos. Aplicando el mismo razonamiento, "Relac.Cardinalidad" es el promedio de los cocientes entre las cantidades promedio de reglas que posee la soluciónhallada por el método y la solución con cardinalidad mínima para cada base. Como puede observarse en la figura 33, las cuatro variantes delmétodo propuesto tienen una cardinalidad parecida y son las de menor tamaño. En particular, *lvqVPSO* es, en promedio, un 7 % superior a la solución de menor tamaño mientras que los restantes incrementan este valor entre un 284 % y un 300 %. Es decir que existe una gran diferencia entre la cardinalidad de la solución del método propuesto y los restantes métodos analizados. Con respecto a la precisión, *cAntMiner* es el método que ofrece los mejores resultados, siendo un 2 % superior al método propuesto pero, como se dijo anteriormente, para lograr este

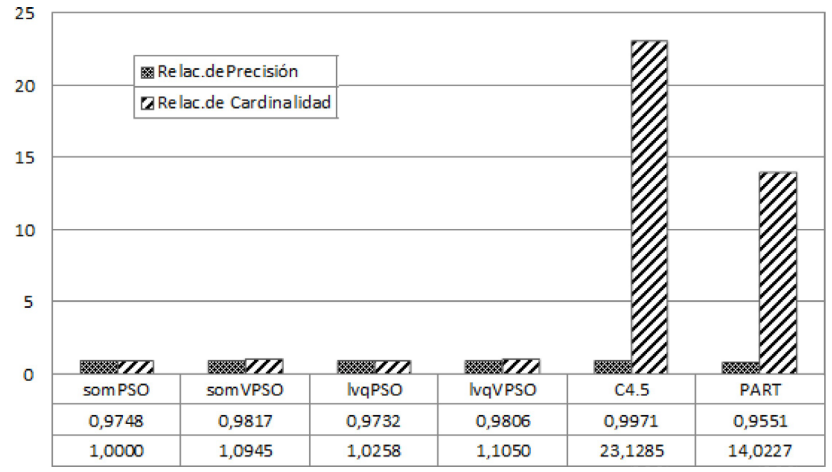
2 % de mejora en la precisión utiliza en promedio un conjunto de reglascuya cardinalidad casi cuadruplica la del método propuesto.

La simplicidad del modelo obtenido por cada método puede verseen la figura 34 y fue estimada calculando la cantidad promediode comparaciones que utiliza cada uno, es decir, el producto entrela cardinalidad del conjunto de reglas y la longitud promedio del antecedente en cada caso. Allí se observa que las variantes del método propuesto requieren menos del 50 % de las utilizadas por *cAntMiner*, aproximadamente el 15 % de las necesarias para *C4.5* y el 35 % de las empleadas por PART.

También se lo ha utilizado en dos casos reales de empresas financieras que otorgan préstamos para consumo y dos bases de datos financieros de crédito al consumidor del repositorio UCI. Una de las bases de datos reales proviene de una importante entidad de ahorro y crédito de Ecuador con más de 20 años de trayectoria en el mercado interno. La otra base de datos real pertenece a la Cooperativa de Ahorro y Crédito, una institución de ahorro mutuo de Ecuador. Los préstamos para consumo son operaciones con montos muy inferiores a los préstamos hipotecarios y requieren tomar decisiones rápidas, ya que, generalmente, son acordados con los clientes a través de un servicio en línea.

La figura 35 resume los resultados obtenidos para estos cuatro conjuntos de datos. Los resultados de la aplicación de las cuatro variantes del método propuesto han sido comparados con los métodos *C4.5* y PART. En esta

oportunidad no se ha incluido *cAntMiner* por el excesivo tiempo requerido por este método para brindar el conjunto dereglas.



somPSO	somVPSO	lvqPSO	lvqVPSO	C4.5	PART
0,9748	0,9817	0,9732	0,9806	0,9971	0,9551
1,0000	1,0945	1,0258	1,1050	23,1285	14,0227

Figura 35: Resultados obtenidos sobre dos casos reales de empresas financieras que otorgan préstamos para consumo y dos bases de datos financieros de crédito al consumidor de repositorio.

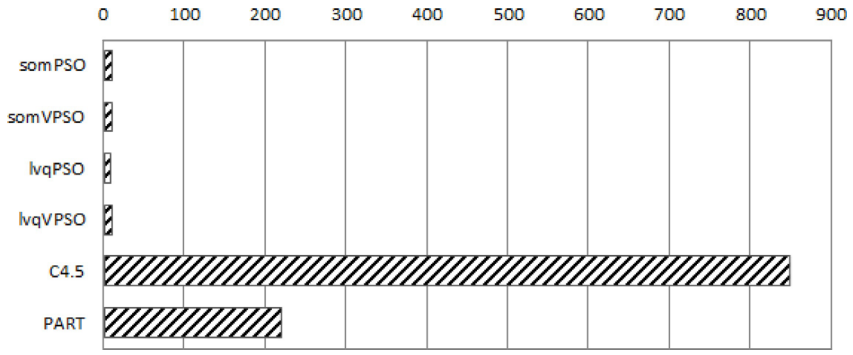


Figura 36: Simplicidad del modelo.

Nuevamente, se observa que todas las variantes del método propuesto arrojan los conjuntos de reglas con la menor cardinalidad.Se recuerda que el valor 1 en "Relac.Cardinalidad" representa que se ha obtenido el conjunto con la menor cantidad de reglas. Los otros métodos son 23 o 14 veces más grandes. La versión *lvqPSO* es la de mejor desempeño entre las variantes propuestas con una precisión promedio un 1 % inferior a *C4.5* pero si se observa la simplicidad graficada en la figura 36, equivalente a la cantidad de condiciones promedio que involucra el modelo completo, puede verse que es

80 veces menor requiriendo un promedio de 10 preguntas para decidir si otorga o no el préstamo contra las más de 800 que plantea *C4.5*. El 1 % de precisión es aceptable en este tipo de situaciones ya que el volumen de operaciones compensará las decisiones incorrectas. Sin embargo, tomaruna decisión en base a un cuestionario corto que no tendrá más de 10 preguntas incrementa notablemente las posibilidades de concretarlo.Ningún cliente soportará en línea un cuestionario tan extenso como el que proponen los otros métodos.

A.4. Conclusiones

En este anexo se han descripto brevemente los aspectos fundamentales y necesarios para aplicar la variante de PSO desarrollada en el capítulo 3 en la extracción de reglas de clasificación. El énfasis estuvo en alcanzaruna buena cobertura utilizando un número reducido de reglas donde cada una de ellas posea un número mínimo de conjunciones en suantecedente. Esto ha dado como resultado varias publicaciones en las cuales puede obtenerse mayor detalle al respecto (Lanzarini *et al*., 2017;Jimbo Santana *et al*., 2017; Lanzarini *et al*., 2015c; Lanzarini *et al*., 2015b;Lanzarini *et al*., 2015a; Villa Monte *et al*., 2012).

El método propuesto ha sido aplicado sobre varios conjuntos dedatos, tanto de repositorio como reales, con resultados satisfactorios. Sehan verificado las características deseadas aunque en algunos casos la precisión haya sido ligeramente superada por otros métodos existentes.Esto tiene que ver con la presión realizada por mantener la simplicidad del modelo, no permitiendo la generación de reglas con poca cobertura.

Todos los modelos obtenidos han sido los de menor tamaño y la precisión no ha sido inferior en promedio a un 2 % de lo ofrecido por otros métodos.

Por lo tanto, en base a los resultados obtenidos, se considera que el objetivo planteado inicialmente ha sido cumplido. En los casos en los cuales la velocidad en la toma de decisiones es fundamental, el método propuesto es una solución sumamente adecuada.

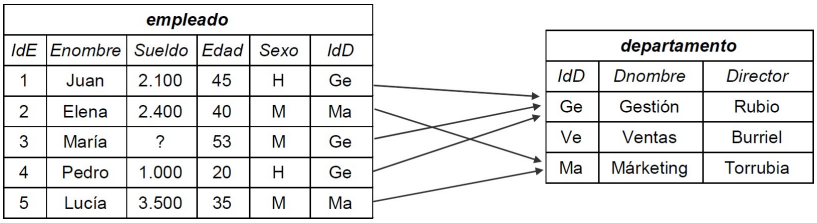
APÉNDICE B

Almacenamiento de documentos de manera estructurada

En esta tesis fue necesario recopilar de Internet varios documentos de texto. Los mismos se consiguieron, en el caso del capítulo 3, a través de la descarga de artículos científicos publicadosen una conocida revista médica y, en el caso del 4, por medio de la captura del contenido de varias páginas web especializadas en unatemática determinada.

Una vez recolectados los documentos de partida, fue necesario continuar con su preprocesamiento. Todos los documentos se expresanen lenguaje natural y, por ende, de forma no estructurada (textual, HTML, XML, JSON, etc.). En la mayoría de los casos, es imprescindible transformar los datos sin procesar (en bruto y disponibles en formato texto) en datos estructurados antes de poderlos utilizar y analizar.

En este anexo, se describe el diseño de una base de datos para persistir los documentos adecuadamente para su posterior procesamiento. Además, se detalla el procedimiento llevado a cabo para almacenarlos



empleado					
IdE	Enombre	Sueldo	Edad	Sexo	IdD
1	Juan	2.100	45	H	Ge
2	Elena	2.400	40	M	Ma
3	María	?	53	M	Ge
4	Pedro	1.000	20	H	Ge
5	Lucía	3.500	35	M	Ma

departamento		
IdD	Dnombre	Director
Ge	Gestión	Rubio
Ve	Ventas	Burriel
Ma	Márketing	Torrubia

Figura 37: Ejemplo de base de datos relacional compuesta por dos tablas.

Fuente: (Hernández Orallo et al., 2004)

```
SELECT D.IdD, D.DNombre, AVG(E.edad)
FROM empleado E JOIN departamento D ON E.IdD=D.IdD
WHERE E.sueldo >2.000
GROUP BY D.IdD, D.DNombre
```

⇓

Resultado de la consulta		
IdD	Dnombre	AVG_edad
Ge	Gestión	45,0
Ma	Marketing	37,5

Figura 38: Ejemplo de consulta SQL sobre los datos de la figura 37.

Fuente: (Hernández Orallo et al., 2004)

en dicha estructura indicando algunos recursos y herramientas utilizadas para manipular el lenguaje natural.

B.1. Introducción

En el capítulo 2, al mencionar el tipo de dato al que se le puede aplicar técnicas de DM, se diferenciaron los datos estructurados, provenientes comúnmente de bases de datos relacionales, de los no estructurados (en formato texto), provenientes, generalmente, de la web o de otros tipos de repositorios de documentos.

Una de las principales características de las bases de datos relacionales es la existencia de un esquema asociado. Los datos siguen una estructura y son, por tanto, estructurados. Una base de datos

relacional es una colección de entidades (tablas), cada una de las cualesconsta de un conjunto de atributos (columnas o campos) y puede contener muchas tuplas (registros o filas). Cada tupla representa unobjeto, el cual se describe a través de los valores de sus atributos. Lafigura 37 ilustra un pequeño esquema de base de datos compuesto de solo dos tablas: "empleado" y "departamento".

Generalmente, las bases de datos poseen un mayor número de tablas,atributos y registros. Si bien constituyen la fuente de datos de la mayoría de las aplicaciones de DM, muchas de las técnicas utilizadas son capacesde tratar una tabla a la vez. Tradicionalmente, con el uso de lenguajesde consulta especialmente diseñados para ello, como SQL, se consigue combinar la información que se requiere de varias tablas en una única tabla o vista minable. La figura 38 muestra una consulta típica sobre la tabla "empleado".

En cuanto a los tipos de datos de los atributos, existen muchos. Desdeel punto de vista de las técnicas de DM más habituales (como lo esla obtención de reglas de clasificación vista en el anexo A), interesa distinguir dos tipos: numéricos y categóricos. Sin embargo, existen otras representaciones de datos más complejas como, por ejemplo, las cadenas de caracteres o los campos tipo "memo" en los cuales el texto libre predomina y para los cuales hacen falta técnicas específicas para tratarlos.

Aunque hoy en día las bases de datos relacionales continúan siendo las más utilizadas, existen aplicaciones que utilizan otro tipo de organización de datos más complejos. Tal es el caso de las bases de datos documentales formadas por grandes colecciones de documentos no estructurados, constituidos en su mayoría de texto.

En estos casos, las técnicas de DM pueden utilizarse para obtener asociaciones entre los contenidos, agrupar o clasificar objetos textuales.Pero para ello, los métodos de minería deben integrarse con técnicas deIR y NLP y hacer uso de diccionarios y tesauros.

Generalmente, como se vió en el capítulo 2, se toma el cuerpo del documento (texto) y se le aplican ciertas transformaciones básicas, a fin de obtener objetos mucho más útiles a los que se les pueda posteriormente realizar una tarea de análisis más significativa. En este sentido, es útil imponer una cierta estructura en los datos que sea lo suficientemente rica como para permitir operaciones interesantes.

A continuación, se describirán las tareas de preprocesamiento quese le realizaron a los documentos descargados de *PLOS Medicine* y el diseño de base de datos propuesto para almacenarlos correctamente.

B.2. Descripción del diseño propuesto

El diseño de base de datos se realizó de forma tal que pudiera almacenarse todo el contenido de los documentos de texto de manera estructurada. Este diseño fue concebido para documentos científicos teniendo en cuenta las necesidades de los resúmenes extractivos.

Los documentos científicos utilizan una estructura común que, generalmente, comienza con su título, el listado de autores e informaciónadicional (afiliación, correos electrónicos, etc.). Le sigue un resumen

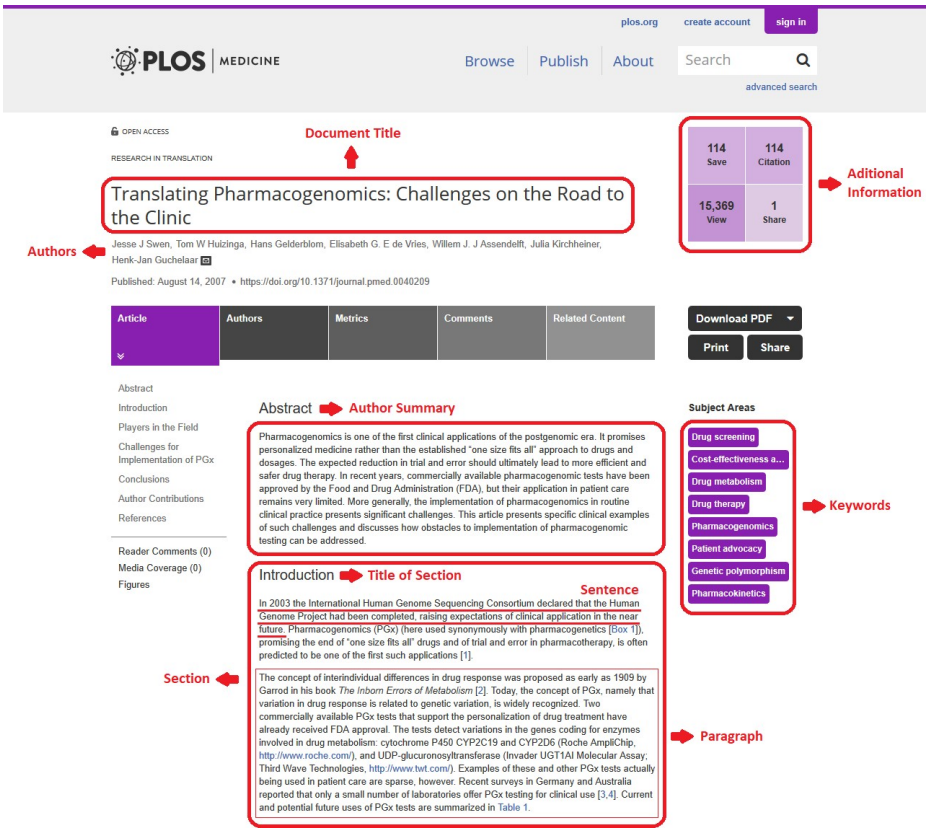


Figura 39: Estructura típica de un documento científico.

y las palabras clave. Luego, su contenido se organiza en secciones, subsecciones, etc., cada una de ellas con título propio. El contenidode cada sección consiste en una secuencia de párrafos formados por oraciones. Cada oración contiene una secuencia de palabras. Normalmente, en las primeras secciones se proporciona información general y luego detallan aspectos más específicos en relación con el tema tratado. Finalmente, se discuten los resultados y las conclusiones, antes de listar las referencias bibliográficas realizadas. Toda la información debe ser almacenada. La figura 39 muestra cómo estos documentos se organizan.

Independientemente del tipo de documento del que se trata, son fundamentalmente una secuencia de caracteres (en una codificación

específica) contenida en un archivo en algún formato. Mientras que los archivos de texto sin formato solo contienen los caracteres de un texto, otros (como los archivos XML) pueden expresar su contenido mediante una estrategia de marcado, como se muestra a continuación.

```
<article>
   <front>
      <article-title>...</article-title>
      <abstract>...</abstract>
   </front>
   <body>
      <sec id=`s1'>
         <title>...</title>
         <p>...</p>
      </sec>
      <sec id=`s1a'>...</sec>
      <sec id=`s1b'>...</sec>
      <sec id=`s2'>...</sec>
      ...
  </body>
</article>
```

Este formato, mediante etiquetas, identifica partes específicas dentrode un documento. El *Center for Digital Research in the Humanities* de la *Universidad de Nebraska–Lincoln* define el formato XML como un estándar de codificación que ayuda en la creación, recuperación y almacenamiento de documentos[22]. Este formato es fundamental para procesar el contenido del documento y almacenarlo con éxito en la basede datos propuesta.

En síntesis, los contenidos en la web constan de datos no estructurados (texto), datos muy poco estructurados (como en los documentos HTML), datos semiestructurados (como los documentos XML) y datos más

[22] https://cdrh.unl.edu/articles/basicguide/XML

estructurados (como el contenido de las bases de datos relacionales). Sin embargo, hoy la mayoría del contenido corresponde a texto no estructurado.

Cada una de las palabras de un documento debe considerarse junto con la ubicación dentro del mismo. Con el diseño de datos propuesto a continuación, el documento original podría reconstruirse utilizando la información almacenada, lo que garantiza su consistencia. La figura 40 muestra el diseño de base de datos propuesto para almacenar este tipode documentos.

La base de datos consta de quince tablas que almacenan toda la información necesaria para ubicar cada palabra dentro de un documento. El modelo tiene en cuenta la revista, el número y los artículos publicados en cada uno (tablas "journal", "edition" y "document"). El texto del documento se divide en las tablas denominadas "section", "paragraph" y "sentence". La tabla "section" es recursiva, lo que permite la inclusión de secciones dentro de otras, algo bastante común en este tipo de documentos. El resto de la información (todos los títulos, las palabras clave y el contenido de las secciones) está relacionado con la tabla "token". Cada palabra del texto se reduce a su raíz al aplicar un algoritmo de stemming y se almacena en la tabla "stem" relacionada con la palabra correspondiente en aquella tabla. Como puede recordarse del capítulo 2, el stemming es la tarea de reducción de términos más significativa en TM, ya que el número de palabras derivadas de la misma raíz es muy alto.

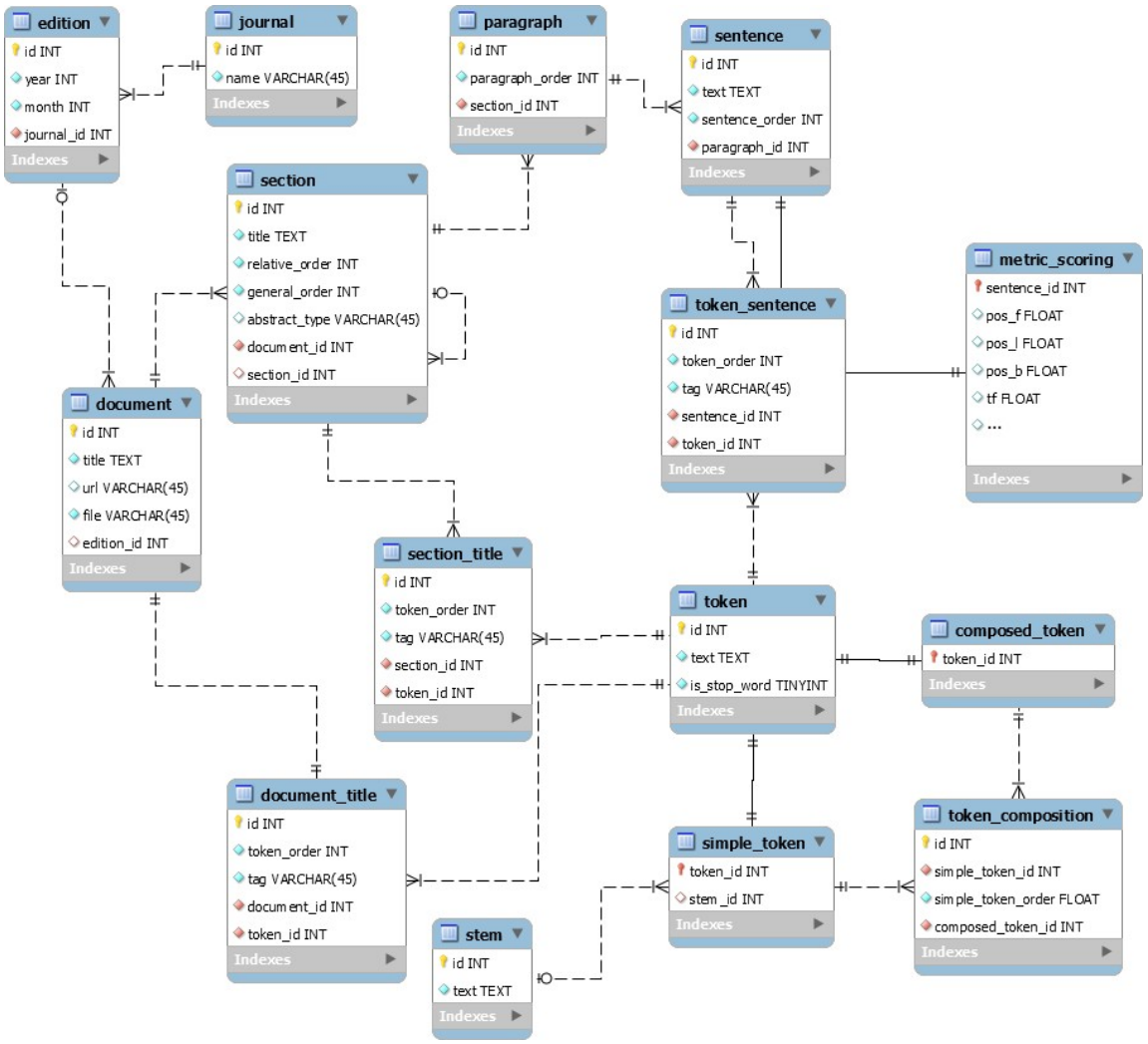


Figura 40: Modelo de base de datos propuesto para almacenar contenido textual de los artículos científicos.

En el capítulo 3, se utilizaron artículos sobre temas de salud publicados por una revista de PLOS. Estos documentos pudieron descargarse de forma gratuita en formato XML a través de Internet. Tener los documentos en este formato facilita la identificación de cada una de las partes del documento.

Los archivos en aquel formato pueden procesarse con bibliotecas especiales que crean un árbol con todos los elementos del documento. Cada nodo en aquel árbol es un objeto que representa una parte del documento. Esto puede usarse para extraer información específica del documento y almacenarla en la base de datos. Además, estos objetos pueden manipularse mediante

programación y cualquier cambio que se les haga se verá reflejado en la visualización posterior del documento.Esto puede resultar interesante, ya que al identificar las partes relevantes de un documento se las podría diferenciar automáticamente del resto visualizando el documento resaltado (tal como lo hace el ser humano con un marcador).

Una vez conseguidos los artículos en XML del sitio web de PLOS, se llevaron a cabo las siguientes acciones:

- Se extrajo el título junto con el resumen creado por el o los autores.
- Se procesó cada una de las secciones identificando su título. Ciertas secciones, como "Bibliography" y "Acknowledgments", no se tuvieron en cuenta. Tampoco las figuras, tablas y ecuaciones incluidas en el artículo. Todas ellas fueron descartadas.
- Para conseguir las oraciones, los párrafos de cada sección se segmentaron utilizando el punto final como delimitador, teniendo en cuenta cuando se lo usó como separador decimal o como parte de una abreviatura, entre otros casos.
- La lista de palabras clave, los títulos y las oraciones se tokenizaron utilizando espacios en blanco y signos de puntuación. Cada token se agregó a la tabla "token_sentence" después de verificar que el token existiera.
- En caso que el token no fuera encontrado, se lo agrega, por un lado,a la tabla "token", marcándolo como *stop-word* si lo es, y por otro lado, a "simple_token" o "composed_token", según corresponda. Cada token simple tiene su raíz correspondiente almacenada en la tabla "stem".

El proceso de cargado de información a la base de datos desde archivos XML debe llevarse a cabo cuidadosamente para evitar alterar el orden de las secciones y la estructura del documento. Durante este proceso, el documento se transforma, debiéndose tomar varias decisiones: ¿Qué algoritmo de stemming usar? ¿Qué se considera un token? ¿Cuáles delimitadores utilizar para segmentar el texto? ¿Cuál es la unidad de análisis textual? Y muchas preguntas más. El resultado final del proceso será una vista del documento a partir del cual, por ejemplo, calcular una serie de métricas (como las de la tabla 1). Por esta razón es esencial tener control sobre el impacto del procesamientoprevio realizado en los documentos y registrar en un diseño de datos todos los resultados parciales de cada una de las transformaciones realizadas.

A partir del conjunto de documentos almacenado, por cada oración se calcula cada métrica de puntuación utilizada en la generación extractiva de resúmenes. Esto requiere, entre otras cosas, procesar variables de tipo String varias veces usando métodos de partición de cadenas. Incluso si antes de calcular cada una de las métricas se probaran diferentes tareas de preprocesamiento y los resultados nose los almacenara, esto demoraría más tiempo. Entonces, cada una de las métricas podría calcularse sobre el resultado de una consultaSQL realizada para todas las oraciones en el documento. Por ejemplo, la consulta podría recuperar los identificadores correspondientes a los tokens asociados con cada oración del documento y luego, con algún lenguaje de programación como *Python*, calcularse fácilmente las frecuencias necesarias en varias de las métricas. Lo interesante es que, al igual que sus raíces, se pueden recuperar los tokens, e incluso obtenerlas palabras clave o también filtrar ciertos tipos de palabras (como

sustantivos, verbos, etc.). Además, es posible recalcular las métricas por párrafos sin requerir muchas modificaciones.

Una vez calculadas todas las métricas, sus valores en forma de matriz (tal como se la vio en la ecuación 4) serán almacenados en la tabla "metric_scoring" del modelo de datos (figura 40).

B.3. Conclusiones

En este anexo se presentó un modelo de datos para almacenar documentos científicos con una estructura predefinida. Al usarlo,un conjunto de métricas puede ser calculado para cada documentodel corpus usando consultas SQL y un lenguaje de programación apropiado, como *Python*. De esta forma, las oraciones de un documento pueden clasificarse para generar varios tipos de resúmenes extractivos. Para mayor información sobre esta representación consultar Villa Monte *et al*. (2018a).

El diseño de datos lo ha motivado la necesidad de contar con una herramienta que permita procesar documentos, dividir su contenido correctamente y asegurarse de que cada fragmento de texto no pierda su información contextual (Villa Monte *et al*., 2018b). Se lo mantuvo,por un lado, lo suficientemente general como para que pudiera ser útily aplicable a cualquier tarea de NLP y, por otro, lo más independiente posible del problema de TM a resolver.

Esta propuesta es una contribución práctica al área de resúmenes automáticos, permitiendo calcular métricas fácilmente, disminuyendo el tiempo de desarrollo y alguna posible ambigüedad en su interpretación. Ahorra tiempo de cálculo, permite expresar claramente la manera de calcular

cada métrica y facilita la experimentación en esta área de investigación. Además, el documento puede reconstruirse fácilmente de acuerdo con las necesidades del experimento a realizar, permitiendo su integración a otros sistemas que operan sobre los documentos con diferencias significativas.

Esta red se constituyó formalmente en noviembre de 1996 y actualmente 51 universidades argentinas son miembros activos. Sus objetivos son:
"Coordinar actividades académicas relacionadas con el perfeccionamiento docente, la actualización curricular y la utilización de recursos compartidos en el apoyo al desarrollo de las carreras de Ciencia de la Computación y/o Informática en Argentina".
"Establecer un marco de colaboración para el desarrollo de las actividades de posgrado en Ciencia de la Computación y/o Informática de modo de optimizar la asignación y el aprovechamiento de recursos".

Esta tesis se ha desarrollado siguiendo las líneas de investigación que el Instituto de Investigacion en Informática LIDI (III-LIDI, Argentina) y el grupo de investigación Soft Managemement of Internet and Learning (SMILe, España) llevan a cabo de manera colaborativa.
Contó con el apoyo externo de los Profesores Dres. Cristina Puente (Universidad Pontificia Comillas), Aurelio F. Bariviera (Universidad Rovira i Virgili) y Alejandro Sobrino (Universidad de Santiago de Compostela).
Fue presentada por Augusto Villa Monte, en el marco de su doctorado en cotutela, como requisito para obtener el grado de Doctor en Ciencias Informáticas por la Universidad Nacional de La Plata (UNLP, Argentina) y Doctor en Tecnologías Informáticas Avanzadas por la Universidad de Castilla-La Mancha (UCLM, España).

edulp
EDITORIAL DE LA UNLP

ISBN 978-987-8348-96-4