

Universidad de La Habana  
Facultad de Matemática y Computación



# Generación Automática de Ontologías

Autor: **José Ariel Romero Costa**

Tutor: **MSc. Juan Pablo Consuegra Ayala**

Trabajo de Diploma  
presentado en opción al título de  
Licenciado en Ciencia de la Computación



Septiembre de 2020

– *Solo se vive una vez, y ya hice mi elección.*  
– *¡Error!, solo se muere una vez; vivimos todos los días. Además, no importa cuánto tiempo hayas viajado en la dirección equivocada, siempre puedes cambiar de dirección.*

*Dedicado a mis padres, Ramón y Amparo, quienes han batallado a mi lado incansablemente y de forma incondicional durante estos 23 años de vida y 18 años de estudio.*

*A Julio, mi hermano, quien siempre ha estado certero con sus consejos y apoyándome en todo lo que he necesitado fuera y dentro del ámbito educacional.*

*A Odalmis, mi novia, prometida y próximamente esposa, hemos cursado los buenos y malos momentos en estos últimos 4 años y ha estado siempre presente para dar un consejo o apoyo cuando lo necesito.*

*A todo aquel que, sin importar si fue un simple “buenos días”, una charla pasajera o esporádica o una larga amistad de años, intervino en mi vida para hacer de mí lo que soy hoy.*

# Agradecimientos

Considero fuertemente que, si un día no hubiera conocido a una persona en específico o algo “tan sencillo” como no haber saludado a alguien o no haber visto una noticia o información, hoy, quizás, no sería quien soy. Por este motivo aprovecho la investigación que aquí se presenta para agradecer a todas las personas que me han llevado a ser quien soy hoy. Hayamos compartido momentos buenos, malos o un simple gesto de saludo. Gracias a ellos siempre quedarán las marcas que me hicieron llegar a este punto, forjando mi carácter, físico y cualidades buenas y malas.

Algunos de ellos me acompañan desde que nací, por ejemplo, mis padres Ramón y Amparo, que me han dedicado gran parte de su vida y por lo que estoy eternamente agradecido. No tengo manera de pagar todo lo que han hecho por mí, más allá de que con cada paso que doy intento que se sientan más orgullosos de mí. Mi hermano Julio y su esposa Judith, mi abuela Rosa, a la cual cariñosamente desde pequeño le apodé la y así la conocen hoy día, a mis tíos Teté y Alberto, mis primos Mayté y Albertico, mis padrinos María Eugenia y Manero. En fin, sin ánimos de extenderme tanto agradezco a modo general a mi familia, sin importar dónde vivan, por haberme apoyado y ayudado en las decisiones que tomé y al mismo tiempo, ofrecer consejos y su amor incondicional.

Otros aparecieron en mi vida, pero lo importante no es cuándo lo hayan hecho, sino los aportes que en ella hicieron y que hayan llegado para quedarse. Agradezco a mi novia Odalmis por todo el apoyo y amor que me ha dado, a Ismael, gran amigo que me ayudó durante los cinco años de universidad en los viajes de Matanzas a La Habana y viceversa para que pudiera estudiar y alimentarme, e incluso, me dio varios consejos como si fuera mi segundo padre. A Mayte y Jorge, junto a Fany y Jorge Carlos mis vecinos, quienes en estos últimos dos años han sido como mis padres y hermanos adoptivos en La Habana. A Carlos mi padrino de boda, del cual estoy muy agradecido de haber tenido la posibilidad de conocer y

entablar una buena amistad. Nardo, quien nos ha ayudado a mí y a mi familia a lo largo de todos estos años.

Otras personas han quedado en el pasado, pero no porque nos hayamos distanciado emocionalmente, sino físicamente, las distintas situaciones en las que nos puso la vida nos llevó a dejar de vernos, pero son personas que cada vez que hablamos o nos vemos recordamos los momentos compartidos con gran alegría y vivimos nuevos para recordarlos en la próxima ocasión. Ellos son Andy, mi amigo de la infancia, crecimos juntos y fue como un hermano para mí. Arla, su abuela, la cual también me cuidó y enseñó como si fuera su nieto. Yusmeidys, amiga de la vocacional, vivimos muy buenos momentos juntos y entablamos una bonita amistad, si de algo es culpable, es de haber alimentado mi “bichito interior” con los deseos de estudiar medicina, pero lo logró tarde, una vez comenzado a cursar esta carrera, aunque por ese motivo trabajaré duro en el futuro para dedicarme a la medicina computacional. A Yudisleydis, mi profesora de la primaria, que me inculcó los primeros pasos en el mundo del estudio, me enseñó a leer, escribir y calcular, y fue de las primeras personas que incentivaron las matemáticas en mí.

Por último, pero no menos importante, dos seres que a pesar de que no pueden hablar, no hacen más que expresar sus sentimientos y amor hacia mí: mis hijos caninos Gema y Ody, los cuales me acompañan desde hace 4 y 3 años respectivamente. En más de una ocasión han sabido alegrarme el día en una situación donde me sentía triste y estoy más que agradecido y orgulloso por como son.

Este último reto no lo llevé a cabo solo, por eso quiero agradecer también al profesor Juan Pablo por tutorear y ayudarme en esta investigación. Quisiera darle mención a todos pero no puedo, son muchos los que han aparecido y estado presente en mi vida. Espero que sigamos relacionándonos en el futuro, junto con las nuevas personas que conoceré.

*Indudablemente hoy cierro un capítulo en mi vida: uno que llevo escribiendo durante los últimos 19 años. Gracias a todo aquel que aportó un granito de arena en mi formación y que creyó en mí; incluso en los momentos en que ni yo mismo creía.*

José Ariel Romero Costa  
Facultad de Matemática y Computación  
Universidad de La Habana

# Opinión del tutor

La representación de conocimiento se ha convertido en un área de investigación muy activa en los últimos años, motivada tanto por la disponibilidad masiva de nuevos recursos, como por la necesidad de hacer computacionalmente tratable el volumen de datos producidos diariamente. Su relevancia en tareas más amplias, como el descubrimiento automático de conocimiento, la vuelven un área crucial para el desarrollo de varios sectores de la sociedad. En el dominio médico, la aplicación de estas técnicas se vuelve especialmente interesante, ya que procedimientos de inferencia sobre una base de conocimiento puede potencialmente ayudar a diseñar nuevos tratamientos para combatir enfermedades aún no resueltas. En este marco se desarrolla la tesis de licenciatura de José Ariel Romero Costa, con quien pude trabajar este último año en el diseño y validación de un algoritmo para la construcción de ontologías a partir de textos anotados. Esta tesis da continuidad a una línea de investigación que se ha venido desarrollando en la facultad en los últimos años ligada al descubrimiento de conocimiento.

La propuesta de José consiste en un algoritmo para la creación automática de ontologías a partir de una colección anotada de documentos. El sistema utiliza el esquema de anotación del *eHealth-KD Challenge* que ha sido empleado en dos competencias internacionales de extracción de conocimiento, en el marco de los eventos *IberLEF 2019* e *IberLEF 2020*. El trabajo conllevó reconstruir un corpus de texto de Medline sobre el que identificar y reordenar las oraciones del corpus anotado. A partir de las entidades y relaciones señaladas en el texto, se realiza un proceso de normalización con el objetivo de unificar aquellas entidades que difieren sintácticamente pero comparten la misma semántica. La tesis presenta un procedimiento para organizar la información recogida en múltiples oraciones, formando una base de conocimiento que integra las distintas instancias de anotaciones mencionadas entre colecciones. La representación final obtenida constituye un paso de avance en la formalización del esquema de anotación, y sienta las bases

para futuros procesos de inferencia.

Durante el desarrollo de la tesis José demostró independencia y creatividad para lidiar con los problemas encontrados. Tuvo que dominar conceptos y tecnologías del estado del arte, con muchas de las cuales no tuvo contacto durante la carrera. Los problemas que hubo de resolver le servirán de aprendizaje para su desarrollo futuro. El proceso de investigación e implementación desarrollado por José queda recogido en un documento de tesis que avala la capacidad adquirida para presentar resultados de investigación de forma concisa y coherente. Todo esto lo han realizado a la par de las actividades docentes, como estudiante de pregrado y como alumno ayudante de la asignatura *Programación*, donde ha sabido asumir con éxito todas las responsabilidades y retos.

José ha sido alumno ayudante desde su tercer año en la carrera, tiempo que pude compartir con él directamente en clases y en las reuniones del colectivo. En esos años he podido comprobar su interés y dedicación por la asignatura y otros temas relacionados. Este último ejercicio demuestra que ya ha adquirido la madurez necesaria para desarrollar proyectos de alta complejidad con calidad y esmero. Como tutor, estoy complacido por los resultados obtenidos, y por el trabajo realizado con José, que aunque no estuvo exento de obstáculos, logró superar los desafíos. Por estos motivos estoy convencido de que José será un excelente profesional de la Ciencia de la Computación.

*MSc. Juan Pablo Consuegra Ayala*  
Facultad de Matemática y Computación  
Universidad de La Habana

# Resumen

En los últimos años se ha evidenciado un aumento en el desarrollo de técnicas para descubrir conocimiento de forma automática en documentos escritos en lenguaje natural. El procesamiento automático va aparejado a la posibilidad de analizar colecciones de información con disímiles textos. El área de la medicina posee gran importancia para la sociedad y en ella el auge de estas tecnologías es especialmente significativo. Esto se debe a que permite aprovechar la gran cantidad de información disponible en función del avance de este campo. Por otra parte, estas técnicas suelen apoyarse en corpus anotados, los cuales son recursos escasos. Esto se vuelve crítico en el idioma español, donde la cantidad existente es menos generalizada.

En este estudio se define un modelo de anotación de propósito general con el objetivo de capturar los rasgos semánticos más relevantes en los documentos de texto. Además, se presenta un esquema de ontología que se usará para la extracción de conocimiento de forma automática. También se describen los pasos a seguir para la implementación de un algoritmo computacional que busca representar un corpus anotado como un grafo de conocimiento, siguiendo las reglas definidas por la propia ontología. Por último, se muestran las tareas realizadas para la validación de las propuestas dadas, así como resultados en términos matemáticos.

Los resultados alcanzados muestran que el descubrimiento de conocimiento constituye un campo de investigación activo, donde pueden aplicarse técnicas de aprendizaje automático logrando resultados positivos. Se propone la verificación y comparación de un grafo de conocimiento específico creado a partir de las propuestas brindadas en este estudio respecto a la capacidad de aprendizaje e interpretación de un grupo de expertos en el mismo tema. Además se ofrece la continuación de esta línea de investigación con el objetivo de mejorar la efectividad de las propuestas dadas y su aplicación en otros dominios.



# Abstract

In recent years there has been an increase in the development of techniques for automatic knowledge discovery from documents written on natural language. Automatic processing provides the possibility to analyze collections of information containing a large number of texts. The medical field is really important to society and the rise of these technologies is significantly special. An ontology allows taking advantage of the huge amount of data available for it and improve research on this area. These techniques tend to rely on annotated corpus, and they are a scarce resource. This becomes a critical fact in Spanish language, where the existing amount of them is even smaller.

In this study, a general-purpose annotation model is defined to capture the most relevant semantic features contained in text documents. Also, an ontology scheme is presented and used for automatic knowledge extraction. A theoretical step by step implementation of a computational algorithm aiming to build a knowledge graph from an annotated corpus and following the rules of the previously defined ontology is also proposed. Finally, the evaluation and validation process is exposed, as well as statistics results.

The results achieved shows that knowledge discovery constitutes an active research field, where machine learning techniques can be applied achieving positive results. The verification and comparison of a specific knowledge graph built from the proposals provided in this investigation against the learning and interpretation skills of a group of experts on the same field is proposed. Also, the continuation of this research line is suggested, aiming to improve the effectiveness of the proposals given and their application in other domains.

# Índice general

<b>Introducción</b>	<b>1</b>
<b>1. Generación Automática de Ontologías</b>	<b>6</b>
1.1. Dominio médico . . . . .	11
1.2. Métodos de evaluación . . . . .	12
1.2.1. Comparación con un estándar dorado . . . . .	13
1.2.2. Evaluación a través de una aplicación . . . . .	13
1.2.3. Evaluación basada en datos . . . . .	14
1.2.4. Evaluación por humanos . . . . .	14
<b>2. Modelo de Anotación</b>	<b>16</b>
2.1. Esquema de anotación . . . . .	16
2.1.1. Conceptos . . . . .	18
2.1.2. Acciones . . . . .	19
2.1.3. Referencias . . . . .	20
2.1.4. Predicados . . . . .	21
2.1.5. Componiendo conceptos . . . . .	22
2.1.6. Relaciones taxonómicas . . . . .	23
2.1.7. Causalidad e implicación . . . . .	25
2.1.8. Contextualización . . . . .	27
2.1.9. Atributos . . . . .	30
2.2. Formato de anotación . . . . .	30

2.2.1.	Archivo de texto . . . . .	31
2.2.2.	Archivo de anotación . . . . .	31
2.2.3.	Estructura general de la anotación . . . . .	31
2.2.4.	Convenio de anotación de identificadores . . . . .	32
2.2.5.	Anotación de texto . . . . .	32
2.2.6.	Anotación de relaciones . . . . .	33
2.2.7.	Anotación de atributos . . . . .	34
2.2.8.	Anotación de comentarios . . . . .	35
2.2.9.	Consideraciones finales . . . . .	36
2.3.	Anotación automática de documentos . . . . .	37
2.4.	Análisis del corpus . . . . .	38
<b>3.</b>	<b>Propuesta de Solución</b>	<b>39</b>
3.1.	Analizador sintáctico . . . . .	39
3.2.	Modelo ontológico . . . . .	39
3.2.1.	Clases en la ontología . . . . .	40
3.2.2.	Relaciones en la ontología . . . . .	41
3.3.	Grafo de conocimiento . . . . .	43
3.3.1.	Orden topológico . . . . .	44
3.3.2.	Ejemplos de generación automática de ontologías . . . . .	44
3.3.3.	Resumen del algoritmo . . . . .	50
3.4.	Alineación de términos . . . . .	50
<b>4.</b>	<b>Análisis de Resultados</b>	<b>51</b>
4.1.	Marco experimental . . . . .	51
4.2.	Resultados computacionales . . . . .	53
4.3.	Discusión . . . . .	59
	<b>Conclusiones</b>	<b>61</b>
	<b>Recomendaciones</b>	<b>63</b>



# Índice de figuras

2.1. Esquema conceptual del modelo de anotación . . . . .	17
2.2. Anotación de conceptos . . . . .	19
2.3. Anotación de acción . . . . .	20
2.4. Anotación de referencia y predicado . . . . .	20
2.5. Anotación de conceptos compuestos . . . . .	22
2.6. Anotación de las relaciones taxonómicas . . . . .	24
2.7. Anotación de causalidad e implicación . . . . .	26
2.8. Anotación de contextualización . . . . .	28
2.9. Anotación de los atributos . . . . .	29
2.10. Ejemplo de escritura del identificador de anotaciones . . . . .	31
2.11. Estructura de anotación de texto . . . . .	32
2.12. Ejemplo de anotación de texto . . . . .	33
2.13. Estructura de anotación de texto . . . . .	33
2.14. Ejemplo de anotación de relaciones . . . . .	34
2.15. Ejemplo de anotación de la relación same-as . . . . .	34
2.16. Estructura de anotación de texto . . . . .	35
2.17. Ejemplo de anotación de atributo . . . . .	35
2.18. Estructura de anotación de comentarios . . . . .	35
2.19. Ejemplo de anotación de comentario . . . . .	36
2.20. Ejemplo de anotación de comentario . . . . .	37
2.21. Esquema del procesamiento inicial del corpus . . . . .	38
3.1. Ejemplo 1: documento “desmayo.txt” . . . . .	45

3.2. Ejemplo 1: documento “desmayo.ann” . . . . .	45
3.3. Ejemplo 1: grafo de conocimiento luego de realizado el punto 1 .	46
3.4. Ejemplo 1: grafo de conocimiento luego de realizado el punto 3 .	46
3.5. Ejemplo 1: grafo de conocimiento luego de realizado el punto 4 .	46
3.6. Ejemplo 2: documento “higiene.txt” . . . . .	47
3.7. Ejemplo 2: documento “higiene.ann” . . . . .	47
3.8. Ejemplo 2: grafo de conocimiento luego de realizado el punto 1 .	48
3.9. Ejemplo 2: grafo de conocimiento luego de realizado el punto 2 .	48
3.10. Ejemplo 2: grafo de conocimiento luego de realizado el punto 3 .	48
3.11. Ejemplo 2: grafo de conocimiento luego de realizado el punto 4 .	49
4.1. Grado de salida de los nodos del grafo . . . . .	54
4.2. Grado de entrada de los nodos del grafo . . . . .	54
4.3. Grado de salida de los nodos del grafo por rol . . . . .	55
4.4. Grado de entrada de los nodos del grafo por rol . . . . .	55

# Índice de tablas

1.1. Enfoques de las categorías de evaluación de generación de ontologías . . . . .	12
4.1. Estadísticas del corpus anotado . . . . .	52
4.2. Estadísticas del corpus tomado de <i>Medline</i> y del anotado . . . . .	53
4.3. Estadísticas del grafo de conocimiento . . . . .	56
4.4. Ejemplo de extracción de conocimiento implícito . . . . .	57
4.5. Ejemplo de extracción de conocimiento implícito . . . . .	58

# Introducción

El desarrollo tecnológico se ha exacerbado con el advenimiento cada vez mayor del uso del Internet y otros medios avanzados y efectivos que garantizan un mejor futuro para cuestiones importantes de la vida. Debido al continuo aumento del flujo informativo, se hace cada vez más necesaria la utilización de herramientas que permitan identificar, capturar y representar el conocimiento dentro de los sistemas de información, ya sea de dominio específico o de propósito general.

Para ello, ciencias avanzadas como la Ciencia de la Computación y de la Comunicación comprenden la ontología como la definición de conceptos y relaciones en algún dominio, de forma compartida y consensuada. Esta conceptualización debe ser representada de una manera formal, legible y utilizable por los ordenadores [52]. Son creadas para limitar la complejidad de cualquier tema y para organizar la información. Es una medida eficaz en las soluciones de problemas comunes en la vida diaria, que debido a la sobreinformación no se podrían llevar a cabo de forma manual.

Otro de sus beneficios es el hecho de que permiten crear entendimiento compartido al unificar los diferentes puntos de vista. Esto sirve para facilitar la comunicación entre los actores implicados en la construcción de sistemas de información referidos al dominio. Además, permiten el reuso del conocimiento del tema, pues sirve de base ya creada para posteriores investigaciones. También facilitan la recuperación, integración e interoperatividad entre fuentes de conocimiento heterogéneas. Con ellas se provee una base para la representación del conocimiento del dominio y ayudan a identificar las categorías semánticas del mismo [76].

Surge entonces el creciente interés de estudiar técnicas para el descubrimiento automático de conocimiento. El procesamiento automático trae consigo la posibilidad de analizar colecciones masivas de información. Sin embargo, la mayor parte de estas colecciones almacenan la información disponible en documentos textuales escritos en lenguaje natural. La naturaleza en que se expresa la infor-



mación y su estructura semántica poco unificada se vuelven la principal fuente de retos de dicho procesamiento.

En la actualidad, las ontologías se están aplicando en áreas heterogéneas. Entre ellas se encuentran la búsqueda de información, el comercio electrónico, configuraciones de aplicaciones o productos. Además, todo sitio web grande debería usarlas, al menos para organización y navegación [57]. También se están utilizando para el desarrollo de mecanismos que faciliten la comunicación entre las personas y las máquinas por medio del lenguaje natural [33].

En el contexto de la salud y la medicina las ontologías adquieren particular interés, debido a que se están utilizando cada vez más para la solución de disímiles tareas, como la recuperación de información y la búsqueda de respuestas en fragmentos de texto que resuelven preguntas. Diariamente se publican muchos artículos médicos y se hace imposible acceder a todos y obtener conocimiento sobre las novedades médicas y las herramientas que se van desarrollando para solucionar las enfermedades o los problemas de la salud de manera general.

La extracción automática de conocimiento proveería de una herramienta para asistir el desarrollo en esta área a partir de la normalización e integración de los resultados encontrados. Una vez extraído y representado el conocimiento computacionalmente, procesos de inferencia permitirían el descubrimiento de nuevo conocimiento. Ejemplo de esto es el constante descubrimiento de nuevas interacciones entre medicamentos, proteínas y genes. Un sistema de descubrimiento de conocimiento posibilitaría la detección automática de nuevas relaciones entre ellos, y por ende, el descubrimiento de nuevas causas de enfermedades, síntomas y tratamientos.

Algunas de las razones principales para el desarrollo de una ontología son:

- Compartir conocimiento de la estructura de la información entre investigadores y/o usuarios.
- Permitir la reutilización del conocimiento del dominio.
- Hacer explícitas las suposiciones o conocimientos a priori del dominio.
- Separar el conocimiento explícito del dominio del conocimiento implícito operacional.
- Analizar el conocimiento del dominio.

*Compartir conocimiento de la estructura de la información entre investigadores y/o usuarios* es uno de los objetivos comunes en el desarrollo de ontologías [32, 63]. Por ejemplo, si varios sitios web diferentes entre sí contienen información médica o proporcionan servicios médicos de comercio electrónico, y estos comparten y publican la misma ontología subyacente de los términos que utilizan, los agentes informáticos pueden extraer y agregar información de los mismos. Además, estos últimos pudieran utilizar dicha información para responder consultas de los usuarios o como datos de entrada para otras aplicaciones.

*Permitir la reutilización del conocimiento del dominio* fue una de las fuerzas impulsoras detrás del reciente aumento de la investigación ontológica. Por ejemplo, los modelos para muchas áreas diferentes deben representar la idea de tiempo. Esta representación incluye, entre otros, las nociones de intervalos, puntos y medidas relativas a este. Si un grupo de investigadores desarrolla tal ontología en detalle, otros pueden simplemente reutilizarla para sus dominios. Además, si se necesita construir una grande, se pueden integrar varias ya existentes que describan partes específicas de la rama deseada. También se puede reutilizar una de propósito general, como UNSPSC, y extenderla para describir el área de interés.

*Hacer explícitas las suposiciones o conocimientos a priori del dominio* hace posible cambiarlas fácilmente si cambian las ideas tenidas de antemano en este tema. Los supuestos de *codificación rígida* (del inglés *hard-coding*) sobre el mundo hechos en lenguajes de programación hacen que estas no solo sean difíciles de encontrar y comprender, sino también de cambiar, en particular para alguien sin experiencia en el ámbito computacional. Además, las especificaciones explícitas del conocimiento del dominio son útiles para los nuevos usuarios que deben aprender qué significan los términos de este.

*Separar el conocimiento explícito del dominio del conocimiento implícito operacional* es otro uso común de las ontologías. Se puede describir la tarea de configurar un producto a partir de sus componentes, de acuerdo con una especificación requerida e implementar un programa que realice esta configuración independientemente del producto [59]. Seguido de esto, se puede desarrollar una ontología de componentes y características de los ordenadores personales y aplicar el algoritmo para configurar uno de ellos a medida [81].

*Analizar el conocimiento del dominio* es posible una vez se disponga de una especificación declarativa de los términos. El análisis formal de estos es extremadamente valioso cuando se intenta reutilizar ontologías existentes y ampliarlas [56].

## Problemática

En disímiles ocasiones, es necesaria la lectura de un amplio grupo de documentos con gran cantidad de páginas para poder tener conocimiento acerca de algún tema. Incluso algunas veces la información leída no es relevante para lo que se desea, y por tanto, fue una inversión de tiempo en vano. Las ontologías, por otra parte, aceleran en gran medida este proceso, pues el análisis y representación de uno o más corpus en un grafo de conocimiento es cuestión de segundos. Esto posibilita posteriormente, buscar la información deseada a través de consultas realizadas a un sistema computacional.

Para diseñar una ontología no existe una única forma o metodología correcta a emplear y tampoco es objetivo de este estudio definir una. Con esta investigación se busca dar solución al problema de representar un corpus de documentos anotados como base de conocimiento, por medio de una ontología. Para ello es necesaria la definición de la propia ontología a usar y un algoritmo computacional que posibilite la realización de este proceso.

La generación de ontologías es un proceso que de ser realizado de forma manual, toma demasiado tiempo y esfuerzo. Además, en aras de completar la construcción de una base de conocimientos medianamente buena o buena, es necesario involucrar expertos en el dominio. Por otra parte, estas restricciones imposibilitan la realización de una ontología para todos los posibles corpus de estudio. En cambio, con esta investigación se busca realizar este proceso de forma totalmente automática, sin la necesidad tener conocimiento previo del dominio y tras la espera de unos pocos segundos puede verse el resultado de la ontología construida.

Este problema lleva implícito el procesamiento de lenguaje natural, pues en este están escritos los corpus de documentos que serán usados. Además, el campo de estudio de la generación automática de ontologías es relativamente moderno. Este conjunto de aspectos lo hace ser un problema interesante. Es por esto que es el objetivo de esta investigación.

En esta investigación, además de definir un esquema de ontología de propósito general, se lleva a cabo la realización de una base de conocimiento mediante la utilización de un corpus de dominio médico extraído de *Medline* [60].

## Objetivos

La investigación se plantea como objetivo general definir un diseño de ontología de propósito general que sea capaz de representar el conocimiento descrito en uno o más corpus anotados.

Para darle solución al objetivo general es necesaria la implementación de un algoritmo computacional capaz de representar uno o más corpus de documentos anotados como un grafo de conocimiento que responde a las reglas establecidas por la ontología propuesta.

Se proponen los siguientes objetivos específicos:

- Estudiar los esquemas de anotación y corpus usados en diversas tareas de extracción del conocimiento.
- Definir un esquema conceptual de anotación para la representación de los rasgos semánticos más relevantes en textos escritos en lenguaje natural.
- Definir un formato de anotación de archivos para el esquema conceptual previamente definido.
- Diseñar una propuesta de ontología donde se pueda representar un corpus de documentos escritos en lenguaje natural.
- Implementar un algoritmo computacional para representar un corpus anotado como grafo de conocimiento a través de dicha ontología.
- Implementar un marco experimental para la evaluación de la propuesta de solución.

## **Organización de la tesis**

El contenido de la tesis se organiza de la siguiente forma. El capítulo 1 introduce los principales conceptos relacionados con las ontologías y la extracción y representación de conocimiento. En este capítulo, además, se analizan los principales corpus y representaciones semánticas existentes en la literatura. El capítulo 2 describe un modelo de anotación de propósito general que busca capturar los rasgos semánticos más importantes en documentos de texto. En el capítulo 3 se presenta una propuesta para la creación de un grafo de conocimiento a través de un corpus anotado. En el capítulo 4 se muestran los resultados alcanzados en esta investigación, y en función de estos, se discute la efectividad de cada uno de los elementos propuestos en la misma. La investigación finaliza presentando las conclusiones pertinentes y las recomendaciones para su continuidad y mejora.

# Capítulo 1

## Generación Automática de Ontologías

A comienzos del siglo XXI, con el avance de las tecnologías en diferentes dominios, la información no estructurada en internet en forma de noticias y literatura científica ha aumentado de manera exponencial. Sin embargo, la web no ha estado eficiente. Si un autor escribió sobre un tema en algún sitio web, otro pudiera publicar información totalmente contradictoria sobre el mismo tema en otro página web. En otras palabras, la web ha estado desconectada e inconsistente. La extracción de información útil a partir de ella ha sido un proceso erróneo. En aras de resolver este problema, se construyó el concepto de *web semántica* [54]. La motivación detrás de esta idea fue la creación de una plataforma web que estuviera grandemente enlazada, consistente e inteligente. Las ontologías juegan un papel fundamental para llevar a cabo esta idea.

Las ontologías pueden ser creadas a partir de la extracción de la información relevante en un texto, a través de un proceso llamado *ontology population* (traducido al español como *poblar la ontología*). Sin embargo, realizar este proceso de forma manual para ontologías grandes es un proceso trabajoso y se hace imposible construir ontologías para todos los dominios [21]. Por lo tanto, en vez de llevar a cabo la construcción de forma manual, esto ha ido cambiando en la actualidad hacia la generación automática de ontologías.

Las ontologías son ampliamente usadas en sistemas de información, y la construcción de estas ha sido objeto de estudio en varias investigaciones. El mayor problema que trae consigo la creación de ontologías y su uso es la adquisición de

conocimiento y el tiempo que conlleva la utilización de diferentes algoritmos en este proceso [89]. En los últimos años han habido principalmente dos acercamientos para darle solución a este problema:

1. El desarrollo de métodos, metodologías, herramientas y algoritmos para integrar ontologías existentes. Han sido creadas muchas de ellas para diferentes dominios y aplicaciones. Además, existen varios enfoques para usar y unificar las ontologías creadas. El proceso de integración encuentra recursos comunes entre las ontologías usadas y de ellos deriva una nueva que facilita la interoperabilidad entre sistemas informáticos que se basan en las ontologías fuente [90]. La integración puede ser llevada a cabo a través de uno de los puntos siguientes [67]:
  - a) Uniendo las ontologías para crear una única ontología coherente.
  - b) Establecer relaciones entre las ontologías permitiendo la reutilización de la información de unas a otras.
  - c) Relacionando las ontologías a través de encontrar elementos similares en ellas.

Como ejemplo de unión de ontologías se puede mencionar el proyecto de unión de las ontologías *SENSUS* [45] y *Cyc* [51] para crear una única ontología de conocimiento [14]. La propia ontología *SENSUS* fue el resultado de la unión de forma manual de varias ontologías.

Hay algunas investigaciones acerca de métodos generalizados para la unión y establecimiento de relaciones entre ontologías. Muestra de esto es, por ejemplo, *PROMPT* [68], un algoritmo semiautomático que puede realizar ambos procesos. También hay investigaciones en el último apartado de los anteriores (apartado c), por ejemplo el acercamiento realizado por Lacher y Groh a través de clasificación supervisada [49].

2. El desarrollo de métodos, metodologías, herramientas y algoritmos para adquirir y aprender nuevas ontologías de forma automática o semiautomática.

En conjunción al problema de *ontology population* mencionado anteriormente, hay otro problema llamado *ontology enrichment* (que puede traducirse al español como *enriquecer la ontología*). Entre estos dos problemas hay tareas en común y la mayoría de los enfoques que existen no pueden ser catalogados a plenitud con estos términos [27].

En los últimos años ha habido un aumento notable en el área de generación automática o semiautomática de ontologías. Algunas de las herramientas usadas para ello, y ontologías en sí, son mencionadas a continuación en orden cronológico según el año en que fueron dadas a conocer:

Web→KB	Combinando métodos de estadística (aprendizaje Bayesiano) y lógica (regla de aprendizaje FOL), intenta construir una base de conocimiento de propósito general a partir de la web [19, 20].
ASIUM	Aprende marcos verbales y conocimiento taxonómico. Basado en análisis estadístico y sintáctico de textos en francés, intenta construir de forma automática jerarquías entre conceptos usando técnicas de agrupamiento [23, 30, 97].
Clouds	Basado en análisis léxico-sintáctico y aprendizaje inductivo, es una herramienta semiautomática para la creación de un mapa de conceptos, el cual consiste en conceptos como nodos y relaciones como aristas [70].
TextStorm/Clouds	Muy parecido al enfoque anterior, pero esta vez, de forma completamente automática, basando en TextStorm el trabajo que anteriormente realizaba un humano [69].
Hasti	Aprende palabras, conceptos, relaciones y axiomas de ambas formas, incremental y no incremental. Comienza por una pequeña base (aprendizaje desde cero) y usa métodos híbridos para el aprendizaje, como por ejemplo, lógica, algoritmos lingüísticos y heurística [85, 86, 87, 88].
Svetlan	Construye una ontología a través del aprendizaje de categorías de sustantivos, sin importar el dominio del texto [13].
SYNDIKATE	Aprendizaje de palabras, conceptos y relaciones de forma incremental. Basado en la interpretación de ambos niveles: oraciones y textos. También usa lógica inductiva y acercamientos lingüísticos [35, 36].
Doddle II	Aprende relaciones taxonómicas y no taxonómicas usando métodos estadísticos (análisis de ocurrencias), también se apoya en el uso de un diccionario legible por computadoras ( <i>WordNet</i> ) y de textos de dominio específico [96].
ADAPTATIVA	Herramienta semiautomática que permite al usuario escoger un corpus y construir una ontología preliminar. Posterior al desarrollo de forma automática de una ontología

	más grande, puede retroalimentar sus resultados mediante información ofrecida por el usuario [8].
OntoLearn	Emplea un análisis constructivo. Usado entre otras cosas, para traducción e interpretación de texto [41, 61, 65, 66].
ARTEQUAKT	Es un sistema de resolución de preguntas de dominio específico en el arte. Usa una ontología para realizar ese proceso. Basa su aprendizaje en la búsqueda en documentos de internet de información que concuerde con la estructura de clasificación definida [1].
OntoLT	La creación de la base de conocimiento va unida a análisis lingüístico. Esto es realizado mediante el enlace entre estructuras lingüísticas con conceptos y relaciones en la ontología [11].
KnowItAll	Realiza automáticamente el proceso de extracción de conocimiento de la web, de forma autónoma e independiente del dominio. Además, le asocia una probabilidad a cada una de las instancias en aras de seleccionar las relevantes y mejorar la precisión y recobrado [28].
Text2Onto	Tiene enfoques probabilísticos, mediante análisis de ocurrencias, agrupamiento, agrupamiento jerárquico y minería de reglas de asociación [3, 4, 15, 16].
VIKEF	Usa catálogos de productos como fuente de datos y la estructura inherente encontrada en estos [27, 91].
SOBA	Ontología usada para la extracción de información de forma automática de páginas web referentes al fútbol. Además, puede ser usada para responder a preguntas en este dominio [10].
ISOLDE	Genera una ontología de dominio específico al extraer candidatos de clases del contexto lingüístico del texto. Además infiere conocimiento de estas clases a través de los recursos disponibles en la web [94].
LEILA	Extrae instancias de relaciones binarias arbitrarias a partir de lenguaje natural en documentos web. El proceso es automático y sin la intervención de un humano [92].
The BOEMIE	Herramienta que no solo se centra en procesamiento de texto, sino también en multimedia, como por ejemplo imágenes y videos. Se enfoca en poblar la ontología con nuevas instancias o nuevos conceptos y nuevas relaciones semánticas [12].
OPTIMA	Construye la ontología a partir de texto no estructurado y semiestructurado pertenecientes a páginas web [44].



- OntoGain Sistema de creación de ontologías a partir de aprendizaje no supervisado y texto no estructurado. También realiza métodos de análisis formal de conceptos y agrupamiento jerárquico [25].
- CRCTOL Crea ontologías de forma automática y de dominio específico. Realiza un análisis sintáctico del texto completo, combinando métodos estadísticos y de análisis léxico-sintácticos [42].
- NELL Aprendizaje continuo mediante procesamiento de datos de la web. Además, infiere conocimiento nuevo a partir del que ya aprendió. Ambas cosas posibilitan que con el paso del tiempo, la cantidad y calidad del conocimiento aprendido sea mayor [62].

Con la introducción del concepto de *web semántica*, las ontologías se han vuelto común en el espacio de la *red mundial* (conocida en inglés como *world wide web*). En esta red ellas abarcan un gran espectro de campos, desde grandes taxonomías que categorizan sitios web, como sucede en Yahoo [48], hasta categorizaciones de productos a la venta y sus características, como sucede en Amazon [58]. Algunas ontologías representan conocimiento de propósito general, como por ejemplo DBpedia [50]. Otras como la de Ivanović y Budimac [40], son de dominio específico, pero contienen conocimiento más detallado [26].

El *World Wide Web Consortium* (W3C) desarrolló el *Resource Description Framework* [9] (RDF, traducido al español como *Marco de Descripción de Recursos*). El mismo, es un lenguaje para codificar el conocimiento en las páginas web y que sea comprensible para los usuarios que buscan esa información.

La *Agencia de Proyectos de Investigación Avanzada de Defensa* (del inglés *Defense Advanced Research Projects Agency*, DARPA), en conjunto con W3C, desarrollaron el *DARPA Agent Markup Language* (DAML, traducido al español como *Lenguaje de Mercado de DARPA para Agentes*), el cual es una extensión de RDF con construcciones más expresivas destinadas a facilitar la interacción de los agentes en la web [38].

También están surgiendo amplias ontologías de propósito general. Por ejemplo, *United Nations Development Program* (traducido al español como *Programa de las Naciones Unidas para el Desarrollo*) y *Dun & Bradstreet* juntaron sus esfuerzos para desarrollar la ontología *United Nations Standard Products and Services Code* [93] (UNSPSC, traducido al español como *Código Estándar de Productos y Servicios de las Naciones Unidas*) que proporciona terminología para productos y servicios.

Uno de los problemas más importantes en el campo de la generación auto-

mática de ontologías, es cómo reconocer el conocimiento relevante en un documento escrito en lenguaje natural. Algunas métricas de relevancia han sido propuestas [5, 55] en los últimos años. En sentido general, el conocimiento relevante puede ser asociado a los conceptos y acciones que aparecen con más frecuencia en un dominio [26].

## 1.1. Dominio médico

La industria de la e-salud y de las ciencias de vida están en proceso de brindar datos electrónicos a los pacientes para un mejor procesamiento y una rápida manipulación de estos. En aras de lograr que estos datos sean útiles para las aplicaciones de inteligencia artificial, se necesita trabajar con su semántica. De esta forma, se le da paso a la toma de decisiones de forma automática.

Se han desarrollado varias herramientas y ontologías en pos de que los expertos en el dominio puedan usarlas para compartir y anotar información en sus campos. Ejemplo de esto es:

- Resumen de pacientes europeos [47] es uno de los proyectos cuya columna vertebral está inmersa en las tecnologías de la web semántica.
- Guardar e integrar datos acerca del síndrome de prolapso de la válvula mitral [77].
- Ontología de electrocardiografías para enfermedades del corazón y su posterior base de conocimiento [43].
- Ontología basada en e-salud. Enfocada al problema de la discordancia entre conceptos jerárquicos en ontologías [31].
- Ontología del fenotipo humano. Compuesta por 10,088 clases (o términos) y 13,326 relaciones describiendo anomalías del fenotipo humano [46].
- *SNOMED CT* almacena terminología médica en varios idiomas [22, 78].
- *Unified Medical Language System* (traducido al español como *Sistema de Lenguaje Médico Unificado*) integra y clasifica terminología médica y estándares de código [39].
- *Translational Medicine Ontology* (traducido al español como *Ontología de Medicina Transaccional*) tiene carácter evolutivo. Mejora dinámicamente con el pasar del tiempo y las nuevas relaciones que explora de datos en la web y de historias clínicas de pacientes [84].

## 1.2. Métodos de evaluación

Evaluar la calidad de un algoritmo o sistema de generación de ontologías es un aspecto muy importante, puesto que permite a los investigadores y usuarios comprobar la correctitud de las mismas. Además, posibilita refinar o volver a modelar por completo el proceso de generación de la ontología en caso de resultados inesperados y que no se ajustan a los requerimientos iniciales.

Dado que la generación automática de ontologías se realiza a través de varios niveles, su evaluación es un proceso difícil. Considerando esto, un sinnúmero de técnicas han sido propuestas en los últimos años y en la actualidad esta área continúa en desarrollo. Los métodos propuestos pueden ser clasificados en una de las categorías siguientes [2, 27, 71]:

- comparación con un estándar dorado
- evaluación a través de una aplicación
- evaluación basada en datos
- evaluación por humanos

La siguiente tabla pretende mostrar los diferentes enfoques en los que trabajan y evalúan las categorías de evaluación de generación de ontologías previamente mencionadas.

Nivel de evaluación	Estándar dorado	Aplicación	Datos	Humanos
Léxico, vocabulario, conceptos y datos	X	X	X	X
Jerarquía y taxonomía	X	X	X	X
Otras relaciones semánticas	X	X	X	X
Aplicación y contexto		X		X
Sintaxis	X			X
Estructura, arquitectura y diseño				X

Tabla 1.1: Enfoques de las categorías de evaluación de generación de ontologías.

### 1.2.1. Comparación con un estándar dorado

La *comparación con un estándar dorado* (del inglés *golden standard-based evaluation*) se basa en evaluar la ontología resultante con una previamente definida en el mismo dominio, esta última es considerada el “*estándar dorado*” [18]. Mediante esta evaluación se puede validar eficientemente cuán abarcadora del dominio y consistente es la ontología creada respecto a la de referencia. El estándar dorado puede ser una ontología en particular, estadísticas extraídas de un corpus o formalizadas por expertos del dominio.

Estas técnicas de evaluación también son conocidas como *similitud de ontologías* o *alineación de ontologías* (del inglés *ontology mapping* y *ontology alignment* respectivamente). Encontrar un estándar dorado apropiado para la evaluación puede conllevar un reto enorme, puesto que debe ser uno que haya sido creado bajo objetivos y condiciones similares que la ontología creada. Esto implica que usualmente se seleccionen como estándar dorado taxonomías creadas manualmente o taxonomías confiables del mismo dominio.

Las técnicas de evaluación en esta categoría usualmente miden la completitud, consistencia y precisión de los factores de la ontología generada [2]. Además, aunque dos ontologías hayan sido extraídas de un mismo dominio de forma manual y por medio de expertos, pudieran tener amplias diferencias respecto a su estructura. Por tanto, usualmente se necesita algún tipo de normalización o isomorfismo entre ambas ontologías en aras de llevar a cabo la comparación [27].

### 1.2.2. Evaluación a través de una aplicación

La *evaluación a través de una aplicación* es también conocida como *evaluación basada en tareas*, debido a que la ontología creada es evaluada a través de una aplicación realizando una o más tareas. El resultado de una tarea en particular determina cuán buena es en ella la ontología creada, sin tener en cuenta su estructura. Este tipo de evaluación posibilita la detección de inconsistencia entre conceptos y permite la evaluación de la adaptabilidad de la ontología creada mediante el análisis del rendimiento en varios contextos y tareas [83].

Un enfoque práctico de esta evaluación consiste en encontrar una aplicación y evaluar si la ontología creada mejora a la aplicación en el aspecto que se quiere evaluar [34]. Esto permite una evaluación automática y con resultados instantáneos en cuanto a la ontología creada y su uso en la aplicación seleccionada. En cambio, validar la ontología generada en un único caso de estudio no necesaria-

mente implica que el resultado alcanzado se extenderá al resto de casos, ni siquiera se valida la calidad y correctitud de la información extraída [27].

En adición, estas técnicas pueden ser usadas en el proceso de evaluación de la compatibilidad entre varias herramientas con la ontología creada. Además, sirve como comparación de correctitud, contenido y compatibilidad entre varias aplicaciones que usen la ontología creada.

### **1.2.3. Evaluación basada en datos**

La *evaluación basada en datos* es también conocida como *evaluación basada en corpus* [24] utiliza conocimiento específico del dominio (usualmente corpus de textos) para comprobar el contenido aprendido por la ontología creada [5]. Puede ser llevada a cabo comparando las entidades y relaciones en la ontología creada mediante un corpus de datos representativos del mismo dominio, pero que no haya sido usado durante la generación de dicha ontología.

La mayor ventaja de este tipo de evaluación es la posibilidad de comparar una o más ontologías a través de un corpus. Los criterios de comparación son parecidos a los tomados para la evaluación con un estándar dorado, como por ejemplo: completitud, consistencia y precisión. El mayor reto de esta evaluación es encontrar un corpus del dominio deseado que posea las cualidades, condiciones y conocimiento que se desea medir de la ontología generada.

Este enfoque ha sido usado para comparar diferentes ontologías creadas por expertos basándose en el mismo corpus y decidir qué ontología provee el mejor “ajuste” a este [7]. Sin embargo, obtener una métrica absoluta del “ajuste” entre una ontología y un corpus es una tarea difícil, principalmente porque se desconoce a priori cuál sería el mejor “ajuste” [27].

### **1.2.4. Evaluación por humanos**

La *evaluación por humanos* es llevada a cabo por uno o más expertos en el dominio de la ontología creada mediante la comprobación manual de las entidades y relaciones en dicha ontología y evaluarlas basados en alguna métrica [80]. Además, también puede ser usada para definir y formular varios criterios de decisión para la selección de forma manual de la mejor ontología entre un conjunto específico de candidatos.

Esta evaluación también puede ser llevada a cabo mediante la asignación de

una puntuación numérica a cada criterio definido. Luego, una suma (puede ser con pesos en los criterios o no) es llevada a cabo para ofrecer el resultado final. Este tipo de evaluación es conocida como *evaluación basada en criterios*[24]. Esta modalidad es la que usualmente se usa en la selección de la mejor ontología en un conjunto específico [2].

La mayor desventaja de esta evaluación es el alto costo en tiempo y esfuerzo que requiere la comprobación manual de los términos y relaciones pertenecientes a una ontología. No obstante, este enfoque está siendo dejado en desuso en la actualidad [2].

## Capítulo 2

### Modelo de Anotación

El primer paso para el algoritmo presentado en este trabajo, es tener un corpus<sup>1</sup> anotado basado en el esquema presentado a continuación. Además, en este capítulo son analizadas algunas estadísticas de un corpus de oraciones del dominio médico en idioma español. Esto es utilizado en la presente investigación. También son mostradas las herramientas empleadas para construirlo y trabajar con el mismo.

#### 2.1. Esquema de anotación

El modelo de anotación de propósito general empleado busca capturar los rasgos y relaciones semánticas más relevantes presentes en oraciones del lenguaje natural. Este debe evitar ambigüedades tanto como sea posible, de forma que anotadores humanos distintos tengan una alta probabilidad de coincidir. A la misma vez, necesita ser lo suficientemente expresivo para representar los conceptos relevantes del dominio y sus interacciones. Además, debe ser capaz de construir conceptos complejos a partir de combinar otros más simples mediante el uso de un conjunto reducido de reglas. También está diseñado para asistir en desarrollo de sistemas de descubrimiento de conocimiento. Por este motivo es necesario independizar la representación del modelo de la estructura gramatical de las oraciones, y en su lugar, tratar de representar el significado semántico.

---

<sup>1</sup>Un corpus es un conjunto de oraciones y/o documentos de ejemplos reales usados en el lenguaje natural.

Este modelo de anotación se basa en las tripletas *Subject-Action-Target* (en español *Sujeto-Acción-Objetivo*) y en la estructura gramatical *sujeto-verbo-objeto*, normalmente expresada con su abreviatura **SVO**. Tiende a ser el orden predeterminado porque el verbo se usa para dividir el sujeto del predicado, sin necesidad de usar partículas para indicar dónde empiezan o terminan los mismos. Además, es una de las secuencias más frecuente en el lenguaje natural y de hecho es usada en la mayoría de lenguas occidentales y un buen número de orientales [26].

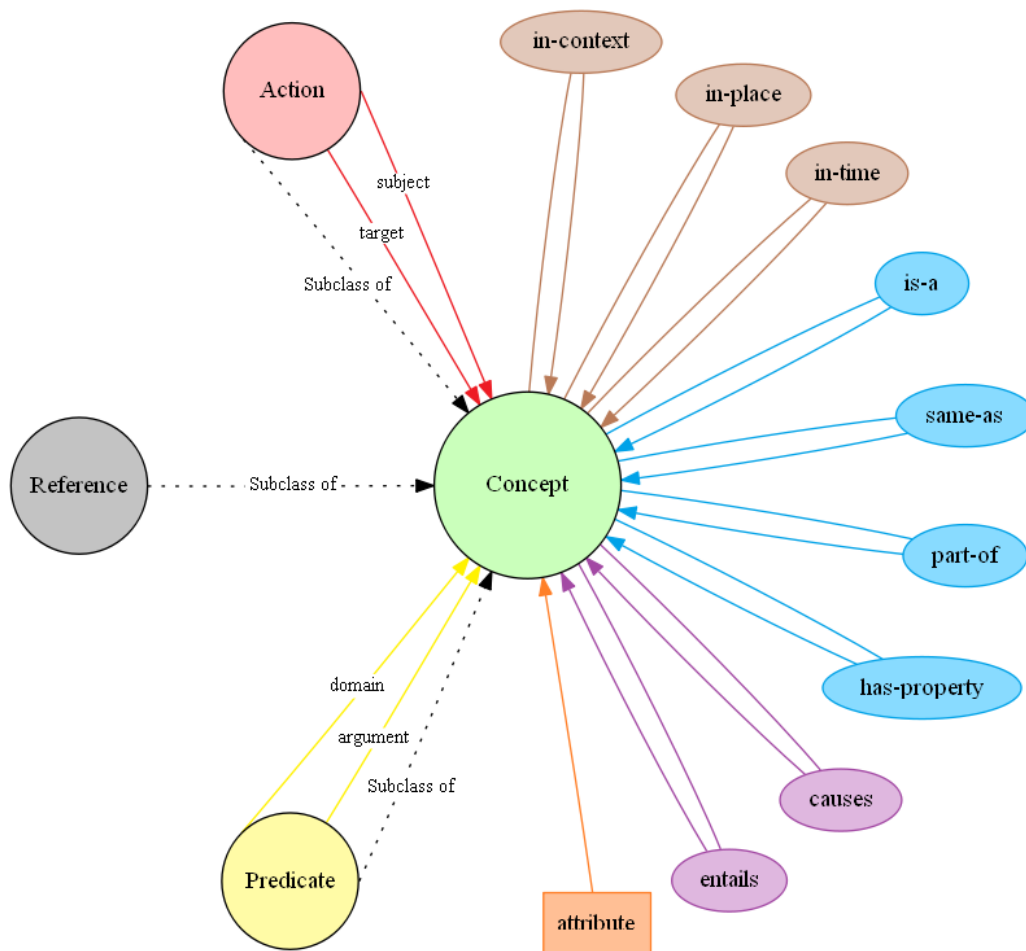


Figura 2.1: Esquema conceptual del modelo de anotación.

Es válido destacar que al estar interesados en fragmentos de conocimiento, el rol semántico de las entidades anotadas puede no coincidir con su rol grama-



tical. Los roles semánticos fundamentales de este modelo son *Concept* y *Action* (en español *Concepto* y *Acción* respectivamente), siendo usados para representar información objetiva acerca de lo que se está haciendo, por quién, y a quién. Estas estructuras pueden ser contextualizadas en tiempo, lugar y otras circunstancias generales.

Existen otros 2 roles semánticos, llamados *Predicate* y *Reference* (en español *Predicado* y *Referencia* respectivamente). *Predicate* es utilizado para construir conceptos más complejos a partir de otros más simples. *Reference* define un término del que se menciona un hecho, pero en el contexto de la oración no está escrito explícitamente, por lo que la información semántica de este rol no está contenida en las anotaciones.

Por último, son usadas seis relaciones con semántica específica para representar conocimiento de propósito general. Las relaciones *is-a*, *part-of*, *same-as* y *has-property* (en español *es-un*, *parte-de*, *igual-que* y *tiene-propiedad* respectivamente) son tomadas de representaciones ontológicas y taxonómicas, mientras que *causes* y *entails* (en español *causa* e *implica* respectivamente) se toman del dominio de la comprensión del texto. Además, las relaciones *in-time*, *in-place* e *in-context* (en español *en-tiempo*, *en-lugar* y *en-contexto* respectivamente) son usadas para dar contexto y cuatro atributos booleanos son asociados a los conceptos. Las próximas secciones explican cada rol semántico y las relaciones detalladamente, incluyendo ejemplos de su uso en oraciones del lenguaje natural.

La figura 2.1 muestra una representación gráfica del modelo de anotación. En el esquema conceptual se puede apreciar que cada uno de los roles semánticos definidos en el modelo de anotación está representado por un círculo. Además, las posibles relaciones definidas entre cada pareja de roles se representan con óvalos y con un rectángulo los atributos que pueden tener los mismos. En color café están representadas las relaciones de contexto, en azul las taxonómicas y en violeta las de causalidad e implicación.

### 2.1.1. Conceptos

El rol *Concept* es usado para anotar fragmentos de texto que representan una unidad atómica de información en el dominio. Puede ser una entidad nombrada, un sustantivo, adjetivo o verbo, que representa un concepto relevante en el dominio del texto. Por ende, la gran mayoría de palabras o frases que expresan un significado propio es anotado de esta manera (o uno de sus derivados, como se

explica más adelante). Palabras tales como artículos, preposiciones y conjunciones, las cuales solo realizan una función gramatical y sin significado semántico, no son anotados.

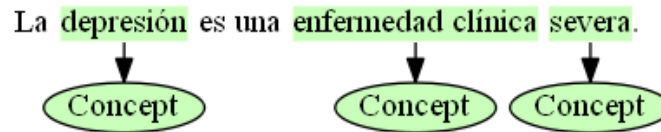


Figura 2.2: Ejemplo de anotación de conceptos.

En la figura 2.2 se distinguen claramente como conceptos en el dominio médico las palabras «depresión» y «severa», cuyo significado de cada uno de ellos es independiente del rol gramatical que tengan en la oración. Algunos conceptos, como «enfermedad clínica» en este caso, se componen de múltiples palabras, ya sea porque de manera independiente no tienen relevancia, o porque al unirlos cobra un significado diferente al de sus componentes individuales. En esta ocasión, a pesar de que «enfermedad» y «clínica» poseen una connotación bien definida por sí mismas, el concepto «enfermedad clínica» tiene gran importancia en el dominio médico, lo cual lo hace una unidad única de información, es decir, un especialista en este campo puede identificarla claramente. Las palabras que conforman un concepto no tienen que estar consecutivas en el texto, pero sí son seleccionadas de izquierda a derecha.

## 2.1.2. Acciones

El rol *Action* es un tipo particular de *Concept* que indica una acción o evento que otro concepto puede realizar o ser objetivo de ella. Un *Action* puede ser enlazado con otros conceptos relevantes a partir de 2 roles semánticos: *subject* y *target* (en español *sujeto* y *objetivo* respectivamente). El *subject* es el que produce la acción, mientras que el *target* es el que recibe los efectos o el objetivo de la acción.

En la figura 2.3 la acción es indicada por una palabra con el rol gramatical de verbo. Intuitivamente este es el caso más común, sin embargo, una acción puede ser indicada además por una palabra con otro rol gramatical, como los sustantivos. Por ejemplo, en la frase "... el empeoramiento de los síntomas ...", la palabra «empeoramiento» se considera también un *Action* a pesar de que no es un verbo, dado que describe un proceso o evento que ocurre sobre otros conceptos.

Por tanto, el rol semántico *Action* describe el significado de un concepto en el dominio semántico, en lugar de su función gramatical en una oración específica. Si un concepto del dominio expresa un proceso o evento que realiza otro concepto o produce un efecto sobre otro(s), entonces es un *Action*, incluso si puede ser usado con una función gramatical distinta.



Figura 2.3: Ejemplo de anotación de acción.

### 2.1.3. Referencias

El rol *Reference* es un tipo de *Concept* que no tiene un significado semántico específico, pero que es necesario por razones gramaticales. Es usado para anotar pronombres (por ejemplo, *este*, *aquel*) y demás elementos que hacen referencia a otro *Concept* presente en la oración, documento y/o corpus. En la figura 2.4 puede verse un ejemplo.

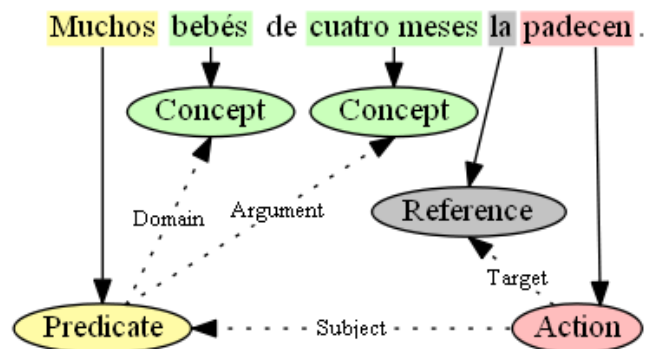


Figura 2.4: Ejemplo de anotación de referencia y predicado.

### 2.1.4. Predicados

El rol *Predicate* es usado para formar conceptos más complejos a partir de aplicar un determinado criterio sobre otros en una oración. Un caso de uso común es para definir el subconjunto perteneciente a un concepto y que cumple determinadas propiedades.

Por ejemplo, en la figura 2.4, la palabra *muchos* cumple la función de filtrar algunos de los bebés, por eso es anotada como *Predicate*.

De conjunto con esta relación, cualquier concepto puede jugar dos roles adicionales: *domain* y *argument* (en español *dominio* y *argumento* respectivamente), completando así su significado. El *Predicate* define el conjunto de objetos pertenecientes al dominio (el concepto enlazado con el rol *domain*) que cumplen el predicado anotado según los argumentos señalados (el o los conceptos anotados con el rol *argument*).

De forma matemática, la relación *Predicate* define al conjunto:

$$\{x \in \text{Domain} \mid \text{Predicate}(x, \text{arg}_1, \text{arg}_2, \dots, \text{arg}_n)\}$$

En el ejemplo de la figura 2.4, el dominio de este *Predicate* es representado por el *Concept* «bebés», y el único argumento es «cuatro meses». Esta construcción da lugar a un nuevo concepto, el de «muchos bebés de 4 meses», el cual puede ser entendido como la aplicación del filtro «muchos» sobre el conjunto de elementos definido por el *Concept* «bebés», de los cuales son seleccionados aquellos con el argumento «cuatro meses».

$$\{x \in \text{Bebés} \mid \text{muchos}(x, \text{cuatro meses})\}$$

El nuevo concepto complejo construido de esta forma es representado en la oración por la anotación *Predicate* en sí misma. Por tanto, para continuar con el ejemplo anterior, en caso de querer que estos «muchos bebés» jugaran el rol *subject* o *target*, la anotación correspondiente debe ir desde un *Action* hacia el *Predicate*, como se muestra en la figura 2.4. Es un error anotar que el *subject* de «padecen» es «bebés» porque este concepto representa «todos los bebés». Por ende, el *Predicate* es usado para representar el concepto filtrado en sí, no el operador de filtrado.

Como caso de uso particular de esta anotación, se encuentra el caso en que un término no representa un concepto relevante por sí mismo (por tanto no debe ser

anotado como *Concept*), sino que denota una propiedad o rasgo medible de otro concepto. Por ejemplo, «tipo», «parte», «nivel» y «cada» en “*tipo de cáncer*”, “*parte del cuerpo*”, “*nivel de glucosa*” y “*cada trimestre*” respectivamente. En tales casos, el *Predicate* debe carecer de alguno de los roles *domain* o *argument*. Si el tipo o clase resultante de formar el predicado coincide con el del concepto a enlazar, entonces el rol utilizado es *domain*. En otro caso, se enlaza al concepto con el rol *argument*.

### 2.1.5. Componiendo conceptos

Así como un *Predicate* puede utilizarse para componer conceptos, se puede lograr un resultado similar al considerar un *Action* como el *subject* o *target* de otro.

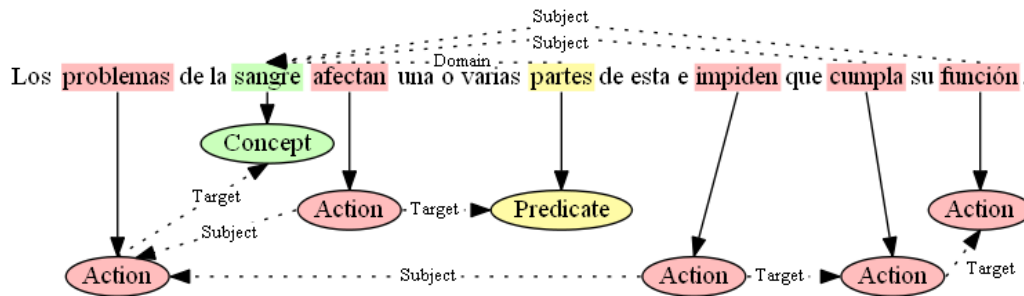


Figura 2.5: Ejemplo de anotación de conceptos compuestos.

Por ejemplo, en la figura 2.5, hay un concepto complejo formado con las palabras «problemas» y «sangre». Este a su vez, actúa como *subject* de «afectan», dado que no todos los «problemas» se «afectan», sino solo aquellos que son «problemas de la sangre». Por otro lado, la propia palabra «sangre» actúa como *domain* del predicado «partes», el cual es el *target* de «afectan». De manera similar sucede con los otros tres conceptos complejos «impiden», «cumpla» y «función».

De esta forma puede apreciarse que la construcción y/o anotación de conceptos complejos es una tarea compleja en sí. Además, esta estrategia puede ser usada para representar la nominalización de un verbo, pues al anotar el *Action* y los correspondientes *subject* y *target* se construye el concepto complejo.

### 2.1.6. Relaciones taxonómicas

Los roles *Action* y *Concept* permiten capturar gran parte del significado semántico de una oración a partir de anotar como acción todos los conceptos que indican alguna interacción entre ellos. Sin embargo, algunos tipos específicos de interacciones son tan comunes que son considerados en diferentes dominios del conocimiento como los bloques constructores para las representaciones ontológicas y taxonómicas. Tal es el caso de las parejas de hiperonimia/hiponimia, anotadas como relaciones *is-a* (en español *es-un*) y meronimia/holonimia, anotadas como relaciones *part-of* (en español *parte-de*), que forman el centro de muchas bases de conocimiento.

Estos dos tipos de relaciones son muy comunes en la mayoría de los dominios del conocimiento, y hay muchas formas distintas para expresar estas ideas en texto. Debido a ello, resulta mejor representarlas explícitamente como relaciones entre conceptos, en lugar de recurrir a anotar como *Action* las formas del verbo ser o estar. Además, una anotación explícita de estas relaciones permite que sistemas de descubrimiento de conocimiento entrenados en estas anotaciones extraigan estructuras más compactas y concisas, dado que no es necesario realizar interpretaciones adicionales.

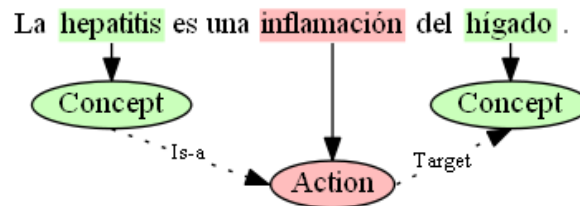
Las relaciones *is-a* y *part-of* pueden ser indicadas explícitamente en el texto por la aparición de patrones textuales comunes, como es el caso de los patrones de Hearst [37]. Sin embargo, aun cuando no ocurrieran en el texto indicaciones explícitas de estas relaciones, se considera su anotación.

En la figura 2.6a puede verse un ejemplo de anotación de la relación *is-a*. En esta oración, las palabras «hepatitis» e «hígado» son claramente conceptos, mientras que «inflamación» es una acción. Como se ha visto anteriormente, una relación con un rol complejo, es decir, un rol que esté relacionado hacia otros roles, implica la relación con él como un todo y no solo con su significado semántico. Por ende, «hepatitis *is-a* inflamación del hígado» es el resultado de la anotación de esta oración.

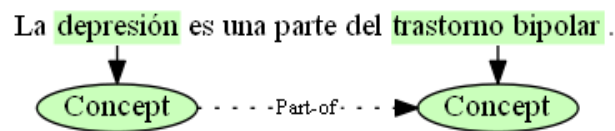
Por otra parte, en la figura 2.6b se puede apreciar un ejemplo de la relación *part-of*. En esta oración son anotadas como conceptos la palabra «depresión» y la frase «trastorno bipolar». Esta oración, a modo de anotación, resulta en «depresión *part-of* trastorno bipolar».

Las parejas de sinonimia, anotadas como relaciones *same-as* (en español *igual-que*) es usada para indicar sinónimos o conceptos que son considerados iguales en

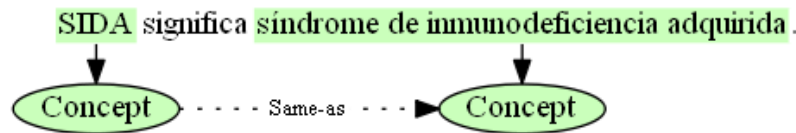
el dominio del documento. Puede ser usada cuando un concepto simple es definido a partir de describirlo como otro concepto más complejo.



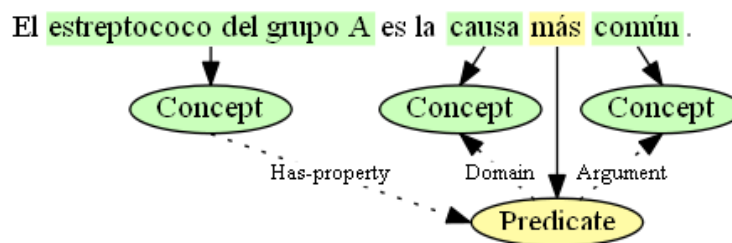
(a) Ejemplo de anotación de hiperonimia e hiponimia.



(b) Ejemplo de anotación de meronimia y holonimia.



(c) Ejemplo de anotación de sinonimia.



(d) Ejemplo de anotación de propiedad.

Figura 2.6: Anotación de las relaciones taxonómicas

La figura 2.6c muestra un ejemplo de anotación de la relación *same-as*. La palabra «SIDA» y la frase «síndrome de inmunodeficiencia adquirida» son anotadas como conceptos. Por tanto, esta oración a modo de anotación queda «SIDA *same-as* síndrome de inmunodeficiencia adquirida».

Las propiedades, anotadas como relaciones *has-property* (traducido al español como *tiene-propiedad*) es usada para especificar que un concepto tiene una propiedad, característica, o puede ser descrita por otro concepto. Sin embargo, este tipo de relación puede conllevar a ciertas dificultades, como por ejemplo la paradoja de Bertrand Russell [82] y la de Grelling-Nelson [95]. Además, una propiedad puede implicar gran cantidad de propiedades e incluso una cantidad infinita de ellas. Por ejemplo si «la persona pesa más de 60 kilogramos» entonces también se cumple que «la persona pesa más de 59 kilogramos» y por consiguiente, que «la persona pesa más de 58 kilogramos». De manera similar, dado que en este caso, el peso es un valor numérico decimal, se pueden construir infinitas propiedades de este tipo. Esto cobra especial importancia a la hora de crear e interpretar la base de conocimientos explicada en el capítulo 3. Puesto que no pueden crearse infinitas relaciones, y por tanto, la interpretación de estas es dependiente del contexto en que se esté analizando.

En la figura 2.6d se puede observar un ejemplo de la anotación *has-property*. En esta oración son anotadas la frase «estreptococo del grupo A» y las palabras «causa» y «común» como conceptos. También, la palabra «más» es anotada como un predicado, en conjunto con «causa» y «común» como dominio y argumento respectivamente. Al ser anotada la relación *has-property*, la anotación resulta como «estreptococo del grupo A *has-property* causa más común».

Para todas las relaciones taxonómicas, solo se considera su anotación cuando la oración implica la existencia de ella, aun cuando fuese implícita. En ningún caso se anota basada solamente en conocimiento externo o del dominio.

### 2.1.7. Causalidad e implicación

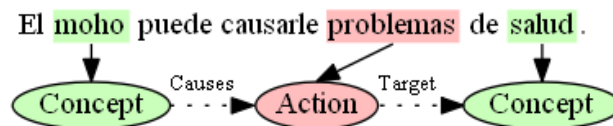
Las cuatro relaciones semánticas presentadas hasta ahora son útiles para capturar la estructura taxonómica del conocimiento expresado en textos del lenguaje natural. Dos relaciones adicionales son definidas para construir conexiones lógicas entre conceptos: *causes* y *entails* (en español *causa* e *implica* respectivamente). La relación *causes* es usada para expresar que un evento, identificado en general como un concepto, es una posible causa para otro evento. En la figura 2.7a se muestra un ejemplo anotado.



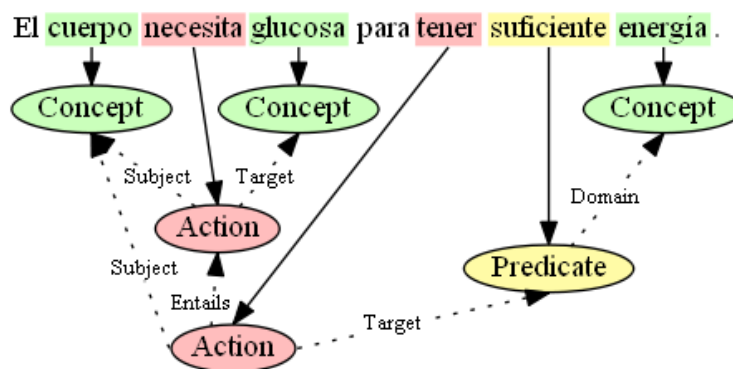
Esta relación indica causalidad, no correlación ni implicación lógica. Por tanto, debe estar declarado con claridad en la oración que hay una conexión de causa directa entre ambos eventos. Además, hay un grado de incertidumbre implicada en la causalidad, lo cual significa que si «A *causes* B», eso no necesariamente implica que cada vez que pase «A» sería seguido por «B», ni que en cualquier caso que ocurra «B» será a causa de «A».

En contraste, la relación *entails* es usada para denotar implicación lógica. En este caso, no es necesario que los eventos estén relacionados por causalidad; lo único que debe cumplirse es que cuando la proposición «A» es verdadera entonces siempre sucede el caso de que la proposición «B» es verdadera. En la figura 2.7b puede verse un ejemplo, donde un concepto complejo, en este caso «tener» implica «necesita», el cual es también un concepto complejo. Desde otro punto de vista, la anotación de esa oración resulta en:

«(tener [suficiente energía] en el cuerpo)  
*entails*  
 (necesitar glucosa en el cuerpo)»



(a) Ejemplo de anotación de causalidad.



(b) Ejemplo de anotación de implicación.

Figura 2.7: Anotación de causalidad e implicación

La anotación de causalidad e implicación evita anotar varias palabras y frases que comparten el mismo significado semántico. Por ejemplo, en la figura 2.7a no resulta necesario anotar «puede causarle» debido a que el significado correcto está siendo representado por la relación *causes*.

### 2.1.8. Contextualización

En ocasiones, los conceptos solo participan en determinada relación con pre-condiciones, como por ejemplo, si dura un período específico de tiempo, solo en una ubicación específica o con algunas propiedades adicionales.

En la figura 2.8b, la anotación «injerto óseo-transplanta-tejidos» falla en capturar la semántica completa del mensaje, dado que el «injerto óseo» no es necesariamente siempre «transplantar tejidos», sino solo en la situación específica en la que este tejido es de los «huesos». Para resolver estas situaciones, se incluyen tres relaciones de contexto: *in-time*, *in-place* y el más general *in-context* (en español *en-tiempo*, *en-lugar* y *en-contexto* respectivamente).

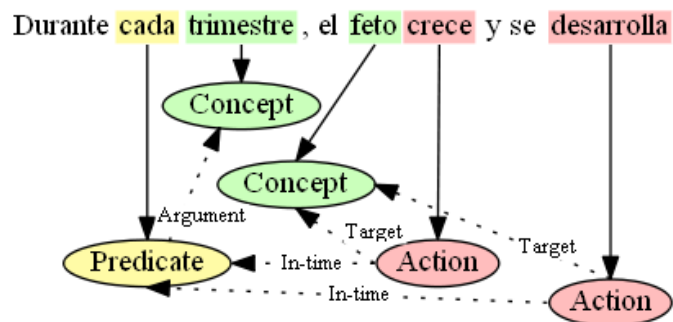
La relación *in-time* restringe un concepto a un instante de tiempo determinado. Además, permite atrapar restricciones más generales, siempre que hablen del concepto mientras cumpla determinada condición o durante el tiempo que lo hace. Puede verse un ejemplo en la figura 2.8a.

La relación *in-place* restringe un concepto a un lugar determinado. Además, puede ser visto como la contextualización de la relación *part-of*, de esta forma permite plantear un hecho sobre un concepto que es parte de otro. Puede verse un ejemplo en la figura 2.8b.

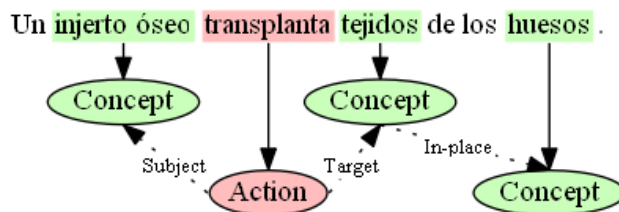
La relación *in-context* restringe un concepto a condiciones más abarcadoras que las descritas anteriormente. Es el contextualizador más general y al igual que el resto, solo debe ser aplicado cuando el contexto habla de un rasgo o valor que puede tener el concepto a contextualizar. Eso implica que el objeto a contextualizar debe tener semántica propia independiente del contexto. A grandes rasgos, puede verse como el contextualizador de la relación *has-property*. Un caso particular de su uso es en oraciones imperativas, donde fragmentos de oración escritos como «... si X entonces haga Y ...» se anotarían como «Y *in-context* X». Puede verse un ejemplo en la figura 2.8c.

La diferencia entre las relaciones de contexto y el resto es que ellas no definen una aserción, sino que son útiles solo para construir conceptos más complejos. Por ejemplo, la anotación «problemas *in-context* únicos» no solo significa que las mujeres tienen problemas de salud, sino que además, son únicos. Es exclusi-

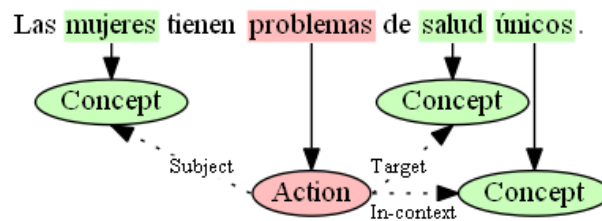
vamente cuando se enlaza con otro concepto, a través de *has-property* u otra relación, que la construcción toma sentido. Por esta razón, no es correcto intercambiar arbitrariamente *in-context* con *has-property*, ya que una relación *has-property* declara una aserción concreta por sí misma. De igual forma enlazar un concepto sobre el que se ha establecido una relación que no es de contextualización, con otro, a través de alguna relación o rol, no indica que dicha relación o rol sea válida solamente para aquellas instancias del concepto que cumplan la propiedad indicada por la relación que no es de contextualización, puesto que estas relaciones, no construyen conceptos complejos que se puedan enlazar.



(a) Ejemplo de anotación de tiempo.



(b) Ejemplo de anotación de lugar.



(c) Ejemplo de anotación de contexto.

Figura 2.8: Anotación de contextualización

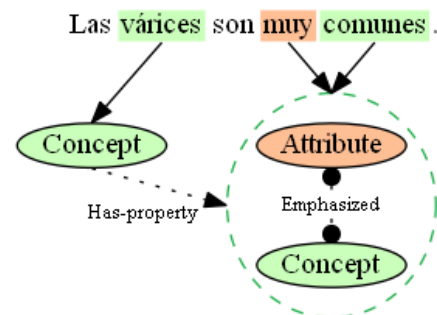
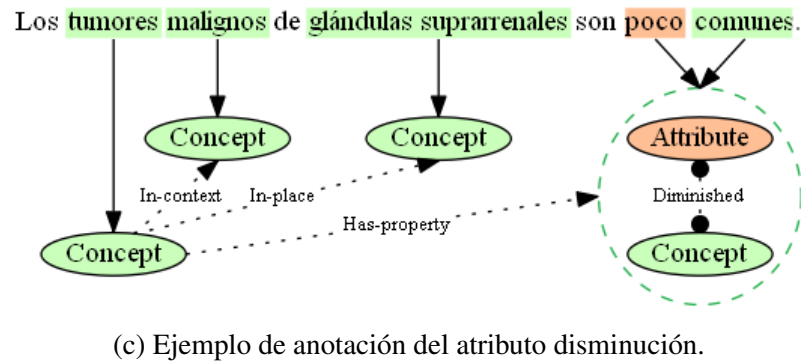
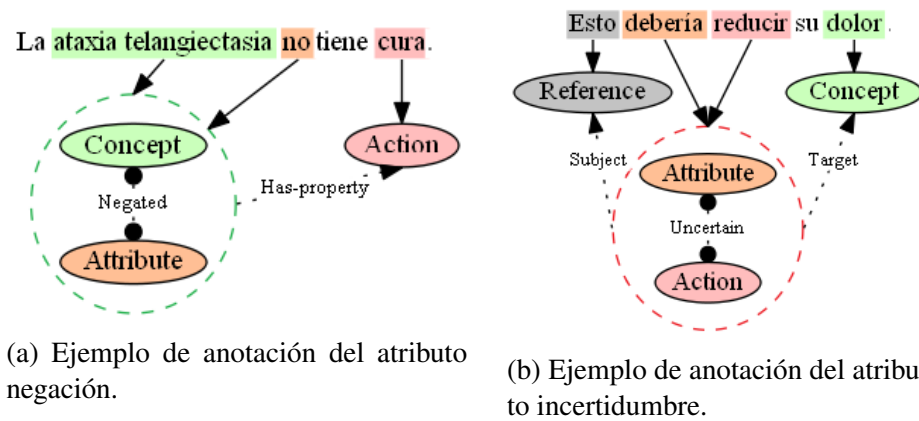


Figura 2.9: Anotación de los atributos

### 2.1.9. Atributos

Cuatro atributos booleanos<sup>2</sup> adicionales pueden ser asociados a cualquier concepto para calificarlo o describirlo un poco más. Ellos son: *negated*, *uncertain*, *diminished* y *emphasized* (en español *negación*, *incertidumbre*, *disminución* y *énfasis* respectivamente). Estos atributos son usados para evitar anotar palabras del idioma que son usadas con bastante frecuencia como *no*, *puede*, *poco*, *mucho*, y en su lugar asociar directamente el calificador correspondiente al concepto en sí. Además, los atributos capturan la negación, incertidumbre, disminución o énfasis que se pretendía en la oración, aun cuando sea implícito y no indicado explícitamente por otra palabra de la misma. Estos atributos acompañan al concepto que modifican en todas las relaciones en que este participe. En la figura 2.9 puede verse un ejemplo anotado de cada uno de estos cuatro atributos.

## 2.2. Formato de anotación

Las anotaciones creadas con este formato son guardadas en archivos separados del documento de texto. Este último nunca es modificado. Para cada documento que vaya a ser anotado, se crea su respectivo archivo de anotación. Ambos archivos son asociados a través de su nombre, los cuales coinciden completamente exceptuando en su extensión. Las extensiones correspondientes al documento de texto y al anotado son *.txt* y *.ann* respectivamente.

En el documento anotado, las anotaciones individuales se conectan a fragmentos de texto a través de rangos de posiciones de los caracteres. Por ejemplo, si el documento comenzara así:

*“El consumo de alcohol puede causar problemas en el hogar ...”*

La palabra «consumo» se identifica por el rango de posiciones 3...10. Las posiciones comienzan en 0 con el inicio del documento, además, todo carácter cuenta como posición válida, incluyendo los espacios en blanco y los cambios de línea.

---

<sup>2</sup>El tipo de dato lógico o booleano es en computación aquel que puede representar valores de lógica binaria, estos son dos valores, los cuales normalmente se representan como falso y verdadero. Se utiliza generalmente en la programación, estadísticas, electrónica y matemáticas mediante la utilización del álgebra booleana.

De manera formal, lo anterior queda expresado como: sea  $\Sigma$  un alfabeto de símbolos, el documento  $D = c_1c_2 \dots c_n$  y la palabra o frase  $w$  perteneciente a este, port tanto,  $\exists v$  tal que  $vw$  es un prefijo de  $D$ , entonces el rango de posiciones de  $w$  en el archivo anotado comienza en  $|v|$  y acaba en  $|v| + |w|$ .

### 2.2.1. Archivo de texto

El archivo de texto debe tener como extensión «.txt» y contener la información del documento original. Además, debe estar guardado en texto plano y codificado usando UTF-8<sup>3</sup>. Puede contener cambios de línea, los cuales cuentan como un símbolo.

### 2.2.2. Archivo de anotación

El archivo de anotación debe tener como extensión «.ann». Además, estar guardado en texto plano y codificado usando UTF-8<sup>3</sup>. Los tipos de anotación específicas que pueden estar presentes en este archivo se explican en las proximas secciones.

### 2.2.3. Estructura general de la anotación

Todas las anotaciones tienen la misma estructura básica: cada línea tiene una sola anotación específica y esta tiene un identificador único que se encuentra al comienzo de la misma. Luego, separado por un carácter TAB, se encuentra el resto de la información en la anotación específica.

El resto de la estructura de cada anotación específica varía en dependencia de los distintos tipos que hay. Esto se explica detalladamente en las siguientes secciones.

```
T9  Concept 658 664 cuerpo
R13 in-place Arg1:T3 Arg2:T13
A1  Negated T10
*   same-as T1 T2
```

Figura 2.10: Ejemplo de escritura del identificador de anotaciones.

<sup>3</sup>Siglas de *Unicode Transformation Format - 8*, un formato de codificación de caracteres.

### 2.2.4. Convenio de anotación de identificadores

Todos los identificadores de anotaciones consisten en un único carácter en mayúsculas identificando el tipo de anotación y a continuación su número. Este carácter inicial es:

- T: texto
- R: relación
- A: atributo
- #: nota o comentario

Adicionalmente, un asterisco (“\*”) puede ser usado como un identificador, pero solo en casos especiales.

### 2.2.5. Anotación de texto

La anotación de texto es una categoría importante, incluso pudiera decirse que es la base de la anotación, pues es quien delimita los fragmentos de texto específicos que serán usados y además, sobre estos es que las relaciones y atributos surten efecto. Estas se basan en la siguiente estructura:

`T<id> [TAB] <type> [SPACE] <span> [TAB] <text>`

Figura 2.11: Estructura de anotación de texto.

- [TAB] y [SPACE] son el carácter TAB y el carácter espacio respectivamente.
- <id> es el número correspondiente a esa anotación específica.
- <type> es el tipo de rol, el cual puede ser solo uno de los cuatro siguientes: *Concept*, *Action*, *Reference* o *Predicate*.
- <span> son los rangos de posiciones de las palabras pertenecientes al texto que será anotado. Si este fragmento contiene más de una palabra, entonces se unirán sus rangos de posiciones usando el carácter punto y coma (“;”) como separador y estas serán escritas separadas por un carácter espacio.
- <text> es el texto específico que será anotado.

Para este ejemplo, y los restantes en esta sección pertenecientes a explicar el formato de anotación, se usará el siguiente documento de ejemplo, el cual contiene una única oración:

*«Las mujeres embarazadas también pueden desarrollar diabetes, llamada diabetes gestacional.»*

En la figura 2.12 se puede ver el resultado de anotar el texto relevante en el documento de prueba.

T1	Concept	4	11	mujeres
T2	Concept	12	23	embarazadas
T3	Action	39	50	desarrollar
T4	Concept	51	59	diabetes
T5	Concept	69	77;78	89 diabetes gestacional

Figura 2.12: Ejemplo de anotación de texto.

## 2.2.6. Anotación de relaciones

Todas las relaciones anotadas son binarias<sup>4</sup>. Estas anotaciones siguen la siguiente estructura:

`R<id>[TAB]<type>[SPACE]Arg1:T<id1>[SPACE]Arg2:T<id2>`

Figura 2.13: Estructura de anotación de texto.

- [TAB] y [SPACE] son el carácter TAB y el carácter espacio respectivamente.
- <id> es el número correspondiente a esa anotación específica.
- <type> es el tipo de relación anotada, el cual puede ser solo uno de los trece vistos anteriormente: *subject*, *target*, *domain*, *argument*, *is-a*, *part-of*, *same-as*, *has-property*, *causes*, *entails*, *in-time*, *in-place* o *in-context*.
- <id1> e <id2> son los identificadores de las anotaciones de texto que participan en esta relación.

<sup>4</sup>Una relación binaria  $R$  es el subconjunto de los elementos del producto cartesiano  $A_1 \times A_2$  que cumplen una determinada condición:  $R = \{(a_1, a_2) : (a_1, a_2) \in A_1 \times A_2 \wedge R(a_1, a_2) = \text{Verdadero}\}$ .



Es necesario tener en cuenta que este tipo de anotaciones se interpretan de izquierda a derecha, es decir, se interpretan «T<id1> <type> T<id2>». Por tanto, el orden en que son anotados sus argumentos tiene vital importancia.

En la figura 2.14 se puede ver el resultado de anotar las relaciones en el documento de prueba.

R1	in-context	Arg1:T1	Arg2:T2
R2	subject	Arg1:T3	Arg2:T1
R3	target	Arg1:T3	Arg2:T4
R4	same-as	Arg1:T3	Arg2:T5

Figura 2.14: Ejemplo de anotación de relaciones.

Como se comentó anteriormente, en algunos casos especiales un asterisco (“\*”) puede ser usado en vez de un identificador. Este es el caso de la relación `same-as`, la cual es simétrica y transitiva. Ellas pueden tener un asterisco al comienzo o un identificador estándar como las restantes relaciones. Cuando se utiliza un asterisco, no es necesario anotar los argumentos explícitamente junto a los identificadores, ni tampoco que sean solo dos argumentos. Por este motivo puede ser compactada y los identificadores que actúan en esta relación deben ser anotados separados por un único carácter espacio, sin importar si estos son dos o más.

A continuación en la figura 2.15 se puede ver un ejemplo de anotación de este tipo en el documento de prueba.

* same-as T3 T5
-----------------

Figura 2.15: Ejemplo de anotación de la relación `same-as`.

### 2.2.7. Anotación de atributos

Los atributos booleanos anteriormente mencionados, son asociados a su respectivo concepto a través de una relación binaria. La misma contiene un único concepto específico unido a un tipo de atributo. Cabe aclarar que, como se comentó en la sección 2.1.9, estos sintetizan una familia de palabras y por ende, semánticamente no tienen una palabra explícitamente asociada.

Las anotaciones de atributos siguen la siguiente estructura:

```
A<id>[TAB]<type>[SPACE]T<id1>
```

Figura 2.16: Estructura de anotación de texto.

- [TAB] y [SPACE] son el carácter TAB y el carácter espacio respectivamente.
- <id> es el número correspondiente a esa anotación específica.
- <type> es el tipo de atributo anotado, el cual puede ser solo uno de los cuatro vistos anteriormente: *Negated*, *Uncertain*, *Diminished* o *Emphasized*.
- <id1> es el identificador de la anotación de texto que es modificada por este atributo.

En la figura 2.17 se puede ver el resultado de anotar el único atributo contenido en el documento de prueba.

```
A1 Uncertain T3
```

Figura 2.17: Ejemplo de anotación de atributo.

### 2.2.8. Anotación de comentarios

Un comentario es texto que no se procesa, por lo que sirve para escribir notas a modo de guía. Las anotaciones de comentarios siguen la siguiente estructura:

```
#<id>[SPACE]<comment>
```

Figura 2.18: Estructura de anotación de comentarios.

- [SPACE] es el carácter espacio.
- <id> es el número correspondiente a esa anotación específica. Este no es obligatorio para anotar comentarios.
- <comment> es el comentario en sí, y puede contener cualquier texto.

En la figura 2.19 se puede ver un ejemplo de un comentario anotado.

```
# Las mujeres embarazadas también pueden desarrollar  
# diabetes, llamada diabetes gestacional.
```

Figura 2.19: Ejemplo de anotación de comentario.

### 2.2.9. Consideraciones finales

Las oraciones anotadas en el documento con extensión «.ann» deben estar en el mismo orden con el que aparecen en el documento de texto con extensión «.txt». Además, no es necesario que todas las oraciones en el documento de texto sean anotadas, incluso puede que ninguna lo esté.

Como se vio en la sección 2.2.4, la letra inicial del identificador debe ser mayúscula, no obstante, no importa si es minúscula, aunque siempre guiarse por los convenios es una buena práctica.

Por otra parte, el convenio de anotación de los tipos es hacerlo con la primera letra en mayúscula y el resto en minúsculas para los textos y atributos, mientras que para las relaciones se debe anotar completamente en minúsculas.

Además, el orden de las anotaciones en el archivo no es relevante y tampoco lo son los números específicos de los identificadores. Aunque por cuestión de estética y para una mejor comprensión del anotador, se recomienda escribir siempre las anotaciones de texto primero, luego las relaciones y por último los atributos. En caso de los identificadores, estos deberían comenzar con el número 1 y continuar la secuencia de forma incremental de uno en uno.

Por último, es recomendable ordenar las anotaciones de texto respecto a la posición inicial de la primera palabra en estos. Las relaciones deben estar ordenadas por los identificadores de sus argumentos, es decir, ordenar primero por el identificador del primer argumento y en caso de que varios coincidan, ordenar por el del segundo argumento.

Como caso especial están las anotaciones de relaciones de igualdad que comienzan con asteriscos, las cuales deben ser ordenadas entre ellas con igual criterio que las demás relaciones, pero deben ir después de estas. Estas anotaciones deberían comenzar siempre con un asterisco, aunque, como se mencionó en la sección 2.2.6, no tiene ningún inconveniente hacerlo de manera similar a las demás.

Por último, los atributos deben ordenarse de manera similar, pero en este caso, guiándose por el identificador su único argumento.

Siguiendo las recomendaciones anteriores, el documento de prueba quedaría anotado de la siguiente manera:

```
# Las mujeres embarazadas también pueden desarrollar
# diabetes, llamada diabetes gestacional.
T1 Concept 4 11 mujeres
T2 Concept 12 23 embarazadas
T3 Action 39 50 desarrollar
T4 Concept 51 59 diabetes
T5 Concept 69 77;78 89 diabetes gestacional
R1 in-context Arg1:T1 Arg2:T2
R2 subject Arg1:T3 Arg2:T1
R3 target Arg1:T3 Arg2:T4
* same-as T3 T5
A1 Uncertain T3
```

Figura 2.20: Ejemplo de anotación de comentario.

## 2.3. Anotación automática de documentos

El proceso de anotación de un documento, llevado a cabo por un humano, puede ser engorroso. Dado que un corpus contiene varios de estos, anotarlo puede tomar gran cantidad de tiempo.

No es de extrañar que con el avance tecnológico y científico, principalmente de la inteligencia artificial en este último, este proceso pueda automatizarse. Los avances en este aspecto tienen ventajas y desventajas, a la vez de márgenes de errores y precisión.

En la competencia *eHealth-KD Challenge* presentada en *IberLEF 2019* [75] e *IberLEF 2020* [74] tomaron lugar sistemas automáticos para la anotación de corpus [17]. Las propuestas que compitieron están entrenadas para hacerlo en textos médicos del idioma español y usando el modelo de anotación explicado en esta investigación [17, 72]. De igual forma, otros pueden ser encontrados en la literatura para anotar automáticamente documentos pertenecientes a diferentes dominios e idiomas.

## 2.4. Análisis del corpus

El corpus usado [73] fue construido a partir de un fichero *XML*<sup>5</sup> tomado del sitio web de *Medline* el 9 de enero de 2018, específicamente a las 02 : 30 : 31. *Medline* fue producida y es mantenida por la Biblioteca Nacional de Medicina de los Estados Unidos. Recoge referencias bibliográficas de los artículos publicados en aproximadamente 5,500 revistas médicas desde 1966. Actualmente reúne más de 30,000,000 de citas. Cada registro de *Medline* es la referencia bibliográfica de un artículo científico publicado en una revista médica, con los datos bibliográficos básicos de un artículo: título, autores, nombre de la revista, año de publicación, entre otros. Esto permite la recuperación de estas referencias posteriormente en una biblioteca o a través de un *software* específico de recuperación.

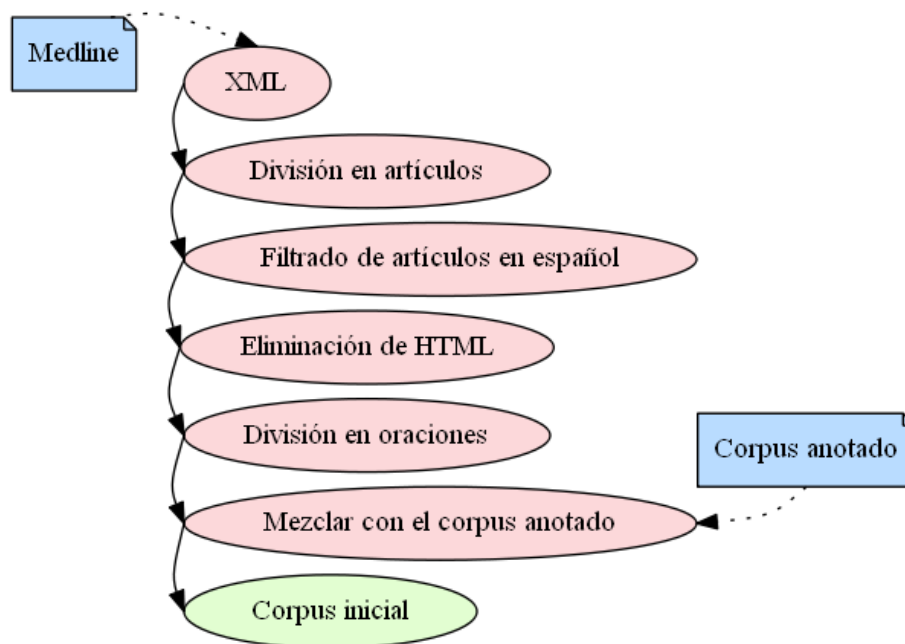


Figura 2.21: Esquema del procesamiento inicial del corpus.

En la figura 2.21 puede verse una representación esquemática del procesamiento inicial que se le hace al corpus de *Medline*, el cual de forma análoga puede aplicarse a otros.

<sup>5</sup>Siglas en inglés de *eXtensible Markup Language*, un lenguaje de marcado desarrollado por el *World Wide Web Consortium* (W3C).

## Capítulo 3

# Propuesta de Solución

Esta investigación busca poder expresar un corpus anotado a través de una ontología definida, generando un grafo de conocimiento como resultado. Otro de los objetivos claros, es poder hacer esto de forma automática mediante un algoritmo computacional.

### 3.1. Analizador sintáctico

Primeramente, es necesaria la creación de una herramienta capaz de fragmentar en objetos con significado computacional el contenido de los archivos de anotación escritos con el formato visto en la sección 2.2.

Dado que en el formato de archivo propuesto contiene una única relación anotada por línea y a su vez, las relaciones tienen su formato de escritura bien definido y sin ambigüedades, llevar a cabo la implementación de este analizador sintáctico es bastante sencillo. Esto puede hacerse a través de expresiones regulares, las cuales son ampliamente usadas y muchos de los lenguajes de programación modernos las incluyen como estructuras integradas.

### 3.2. Modelo ontológico

La ontología propuesta es de propósito general y basada en el modelo de anotación visto en la sección 2.1. Esto posibilita la continuidad del proceso, partiendo desde documentos escritos en lenguaje natural, hasta la creación de una base de conocimiento a partir de ellos.

Una ontología  $O = (C, R, A, Top)$  se define mediante un conjunto no vacío de conceptos  $C$ , un conjunto de relaciones  $R$ , un conjunto de axiomas  $A$  y  $Top$  es

el concepto con más alto nivel en la jerarquía. En esta investigación, la ontología propuesta es generada de forma automática, por tanto, las instancias pertenecientes a los conjuntos  $C$  y  $R$  son generadas de manera automática a medida que se procesa el corpus anotado. Dado que la ontología propuesta no tiene conocimiento previo del dominio, el conjunto  $A$  es vacío. A la vez que  $Top$  no se sabe a priori qué entidad o instancia de entidad es.

En contraste a definir las instancias de los conjuntos  $C$  y  $R$ , son definidas las clases que serán usadas como base de creación de instancias que pertenecen al conjunto  $C$  y las posibles relaciones entre estos, las cuales alimentarán al conjunto  $R$  a medida que se va generando la ontología.

Todo concepto en la ontología pertenece a una de las tres clases siguientes: *entidad simple*, *entidad con atributo* o *entidad compuesta*. El significado específico de estas clases, la creación de instancias a partir de ellas y las relaciones que pueden existir entre ellas son explicadas en las secciones siguientes.

### 3.2.1. Clases en la ontología

Los conceptos usados en la ontología propuesta en esta investigación pertenecen a una de las tres clases siguientes:

- Entidad simple: la más sencilla de las clases, no tiene ningún significado especial. En el modelo de la sección 2.1 representa un concepto sin haberle aplicado relaciones ni atributos. Cada concepto del modelo de anotación representa una entidad simple. Al ser creada una instancia, la propiedad «tipo de concepto» adopta el valor del tipo de concepto específico en el corpus anotado de la palabra o frase correspondiente a la entidad.
- Entidad con atributo: está compuesta por una *entidad simple* y uno o más atributos de los mencionados en la sección 2.1. Puede verse como el resultado de haber aplicado todos los atributos pertenecientes a un mismo concepto. Al ser creada una instancia, la propiedad «tipo de concepto» adopta el valor del «tipo de concepto» de la *entidad simple* que le corresponde.
- Entidad compuesta: es el resultado de aplicar las relaciones en el modelo de anotación explicado en la sección 2.1. Está compuesta por la entidad correspondiente al origen y una o más entidades que son objetivos de dicha relación. Al ser creada una instancia, la propiedad «tipo de concepto» adopta el valor del «tipo de concepto» de la entidad origen que le corresponde.

Estas clases contienen, además, dos propiedades; una de ellas es la palabra o fragmento de texto en sí que representa la entidad y la otra es el tipo de concepto al que representan. Esta última propiedad se basa en los cuatro tipos de concepto existentes en el modelo de anotación explicado en la sección 2.1, estos son: *Concept*, *Action*, *Reference* y *Predicate*.

No existen relaciones entre conceptos utilizados en el modelo de anotación y entidades debido a que cada concepto en el corpus anotado pasa a ser una instancia de entidad simple en la ontología. Por tanto, estas relaciones traerían consigo tener nodos inactivos o duplicados, dado que la información estaría duplicada en nodos de tipo concepto y nodos de tipo entidad simple, y además, las relaciones entre estos no aportarían conocimiento nuevo, en cambio, añadirían redundancia a la base de conocimiento.

### **3.2.2. Relaciones en la ontología**

Los tipos de relaciones definidos en la ontología tienen una estrecha relación con los trece tipos de relación vistos en la sección 2.2.6. Los actores pertenecientes a estas se describen en las secciones siguientes.

#### **Relación subject**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. La única restricción es que la entidad de origen debe tener *Action* en la propiedad «tipo de concepto».

#### **Relación target**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. La única restricción es que la entidad de origen debe tener *Action* en la propiedad «tipo de concepto».

#### **Relación domain**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. La única restricción es que la entidad de origen debe tener *Predicate* en la propiedad «tipo de concepto».

#### **Relación argument**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. La única restricción es que la entidad de origen debe tener *Predicate* en



la propiedad «tipo de concepto».

**Relación is-a**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. Relación taxonómica que potencialmente da la posibilidad de descubrir de conocimiento.

**Relación part-of**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. Relación taxonómica que potencialmente da la posibilidad de descubrir de conocimiento.

**Relación same-as**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. Relación taxonómica que potencialmente da la posibilidad de descubrir de conocimiento.

**Relación has-property**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. Relación taxonómica que potencialmente da la posibilidad de descubrir de conocimiento.

**Relación causes**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. Relación que potencialmente da la posibilidad de descubrir de conocimiento.

**Relación entails**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino. Relación que potencialmente da la posibilidad de descubrir de conocimiento.

**Relación in-time**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino.

**Relación in-place**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino.

**Relación in-context**

Puede tener cualquier instancia de entidad como origen y cualquiera como destino.

La relación directa entre dos instancias de entidades en la base de conocimiento implica la existencia de una relación directa entre ellos en el corpus anotado y viceversa. Por tanto, todo el conocimiento explícito descrito por el corpus será representado por relaciones entre instancias de entidades. Basado en esto, dado un camino  $P$  de tamaño dos o más entre dos instancias de entidades  $u$  y  $v$  en este grafo y además, todas las aristas en  $P$  son aristas que posibilitan el descubrimiento de conocimiento, entonces si no existe una arista directamente entre el nodo  $u$  y el nodo  $v$ , el conocimiento inferido por el camino  $P$  es válido e implícito en el corpus, por tanto, es conocimiento aprendido.

El proceso de creación de una base de conocimiento específica a partir de la definición de esta ontología es llevado a cabo de forma totalmente automática y no de la manera tradicional, con expertos en el dominio añadiendo relaciones entre clases una tras otra. A la interrogante de en qué orden se llevan a cabo estas relaciones y quiénes participan en ellas se le da respuesta en la próxima sección.

### 3.3. Grafo de conocimiento

Una vez que se haya analizado sintácticamente todo el corpus, se tendrá la información de las anotaciones en objetos computacionales y será más sencillo el trabajo con estos. En aras de evitar ambigüedades y concentrar el conocimiento para un mejor entendimiento de este y a la misma vez, poder facilitar la tarea de extraerlo de este grafo por un equipo de cómputo, el texto anotado es normalizado. Esto es llevado a cabo teniendo en cuenta las palabras que lo componen, y anotando en su lugar la palabra primitiva de esta. Por ejemplo, la palabra «sangramiento» será anotada como «sangrar» y la frase «glóbulos rojos» como «glóbulo rojo».

Como se vio en la sección 2.1, es necesario aclarar que hay que darle un orden a la creación de las instancias de las clases y las relaciones en este grafo, pues las propias anotaciones de texto y las relaciones tienen un orden implícito entre

ellas. Por ejemplo, en la propia figura 2.5, se debe procesar primero el rol ejercido por «problemas» y por «cumpla» antes de poder procesar «impiden»; de lo contrario, el conocimiento descrito por «impiden» quedaría incompleto o mal representado.

### 3.3.1. Orden topológico

El orden establecido por el modelo de anotación visto en el capítulo 2 es un orden topológico. Para ello se tiene en cuenta el siguiente orden:

1. Texto: son procesados primero los conceptos del corpus anotado, los cuales son convertidos a instancias de *entidad simple*.
2. Atributos: luego son procesados aquellos conceptos que son modificados por atributos en el corpus anotado, los cuales son convertidos a instancias de *entidad con atributo*, usando como base la *entidad simple* correspondiente a dicho concepto.
3. Relaciones de acción, predicado y contextualización: luego se procesan estas relaciones, las cuales son representadas por instancias de *entidad compuesta*. Además, para cada concepto que actúa en ellas, se busca su correspondiente instancia de entidad perteneciente al grafo de conocimiento.
4. Relaciones taxonómicas y de causa e implicación: por último son añadidas las relaciones que posibilitan el descubrimiento implícito en el corpus. Estas no representan nodos en el grafo, y de igual forma que en el paso anterior, para cada concepto que actúa en ellas, se busca su correspondiente instancia de entidad perteneciente al grafo de conocimiento.

Una vez divididas las relaciones en estos tres grupos, estas son asociadas nuevamente, esta vez teniendo en cuenta la parte izquierda de cada una de ellas (Arg1 en el archivo de anotación). Una instancia de clase es creada por cada una de estas agrupaciones resultantes, además, una instancia creada en un nivel más avanzado, teniendo en cuenta el orden visto anteriormente, representa una mayor cantidad de información y al mismo tiempo, información más específica respecto a su instancia asociada en niveles anteriores.

### 3.3.2. Ejemplos de generación automática de ontologías

A continuación se presentan dos ejemplos de la creación de una base de conocimiento a partir de un corpus anotado. En ambos casos, el corpus solo contiene

un documento, y este está compuesto por una sola oración. La flecha que puede haber en algunas líneas de los documentos de ejemplo, significa que esta en el mismo es muy larga para ser mostrada en una única línea en este escrito y se continuará escribiendo debajo. En estos ejemplos, las palabras o frases no son normalizadas para un mejor entendimiento en lenguaje natural y además, porque hay varias formas, métodos y decisiones de cómo normalizar palabras o frases.

### Primer ejemplo

En la figura 3.1 puede verse el documento de texto de prueba empleado en el primer ejemplo. A su vez, en la figura 3.2 se ve el documento anotado asociado a este. Como puede apreciarse, se siguieron los convenios establecidos en la sección 2.2.9. Por último, la base de conocimiento generada para este ejemplo puede verse en la figura 3.5.

```
El desmayo (o síncope) es una pérdida temporal de la
↪
conciencia.
```

Figura 3.1: Ejemplo 1: documento “desmayo.txt”.

```
# Sentence 1: El desmayo (o síncope) es una pérdida ↪
temporal de la conciencia.
# Keyphrases
T1 Concept 3 10    desmayo
T2 Concept 14 21   síncope
T3 Action 30 37    pérdida
T4 Concept 38 46   temporal
T5 Concept 53 63   conciencia
# Relations
R1 is-a Arg1:T1 Arg2:T3
R2 in-context Arg1:T3 Arg2:T4
R3 target Arg1:T3 Arg2:T5
*   same-as T1 T2
```

Figura 3.2: Ejemplo 1: documento “desmayo.ann”.

Siguiendo el orden topológico establecido anteriormente, se puede ver en la figura 3.3 el grafo de conocimiento resultante luego de realizado el punto 1, donde cada concepto del corpus anotado es transformado en una instancia de entidad simple en la base de conocimiento.

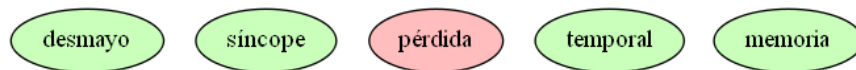


Figura 3.3: Ejemplo 1: grafo de conocimiento luego de realizado el punto 1.

En el punto 2 del orden topológico no puede hacerse nada en este corpus, pues no hay atributos, por tanto, el grafo de conocimiento quedará idéntico. Para el punto 3 son usadas las relaciones R2 y R3, resultando:

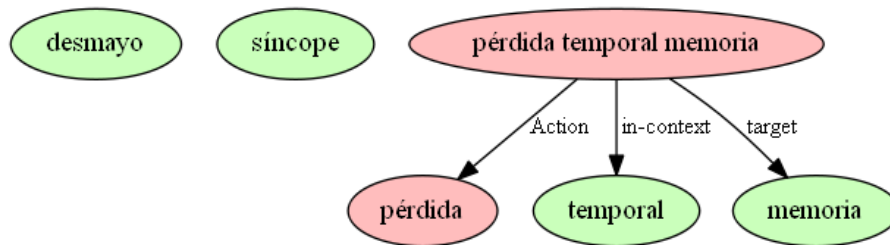


Figura 3.4: Ejemplo 1: grafo de conocimiento luego de realizado el punto 3.

Para el punto 4 son usadas las relaciones que potencialmente posibilitan el conocimiento implícito en el corpus. En este caso son: R1 y \*. Resultando:

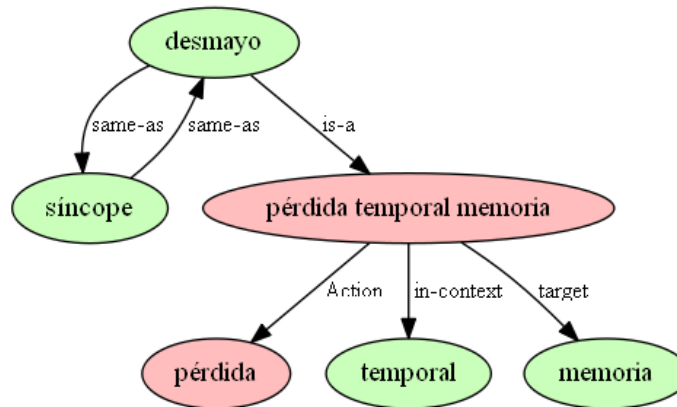


Figura 3.5: Ejemplo 1: grafo de conocimiento luego de realizado el punto 4.

El ejemplo anterior es sencillo, pero aun así, puede descubrirse conocimiento implícito. Se infiere que «síncope *is-a* pérdida temporal memoria».

### Segundo ejemplo

En la figura 3.6 puede verse el documento de texto de prueba empleado en el segundo ejemplo. A su vez, en la figura 3.7 se ve el documento anotado asociado a este. Como puede apreciarse, en este documento también se siguieron los convenios establecidos en la sección 2.2.9. Por último, la base de conocimiento generada para este ejemplo puede verse en la figura 3.11.

Las buenas prácticas de higiene, incluyendo lavarse las ↵  
manos correctamente, pueden evitar infecciones.

Figura 3.6: Ejemplo 2: documento “higiene.txt”.

```
# Sentence 1: Las buenas prácticas de higiene, incluyendo ↵
lavarse las manos correctamente, pueden evitar infecciones.
# Keyphrases
T1  Concept 4 10    buenas
T2  Predicate 11 20 prácticas
T3  Concept 24 31   higiene
T4  Action 44 51    lavarse
T5  Concept 56 61   manos
T6  Concept 62 75   correctamente
T7  Action 84 90    evitar
T8  Concept 91 102  infecciones
# Relations
R1  in-context Arg1:T2 Arg2:T1
R2  domain Arg1:T2 Arg2:T3
R3  causes Arg1:T2 Arg2:T7
R4  is-a Arg1:T4 Arg2:T2
R5  target Arg1:T4 Arg2:T5
R6  in-context Arg1:T4 Arg2:T6
R7  target Arg1:T7 Arg2:T8
# Attributes
A1  Uncertain T7
```

Figura 3.7: Ejemplo 2: documento “higiene.ann”.

Una vez más, siguiendo el orden establecido anteriormente, se puede ver en la figura 3.8 el grafo de conocimiento resultante luego de realizado el punto 1, representando cada concepto del corpus anotado como una instancia de entidad simple.

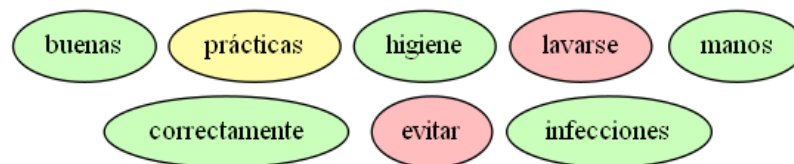


Figura 3.8: Ejemplo 2: grafo de conocimiento luego de realizado el punto 1.

Dándole solución al punto 2, los atributos del corpus anotado son representados como una instancia de entidad con atributo. Resultando:

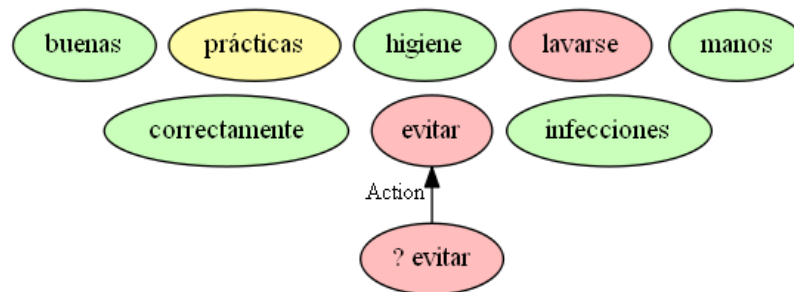


Figura 3.9: Ejemplo 2: grafo de conocimiento luego de realizado el punto 2.

En aras de completar el punto 3, las relaciones de acción, predicado y contextualización toman lugar. Resultando:

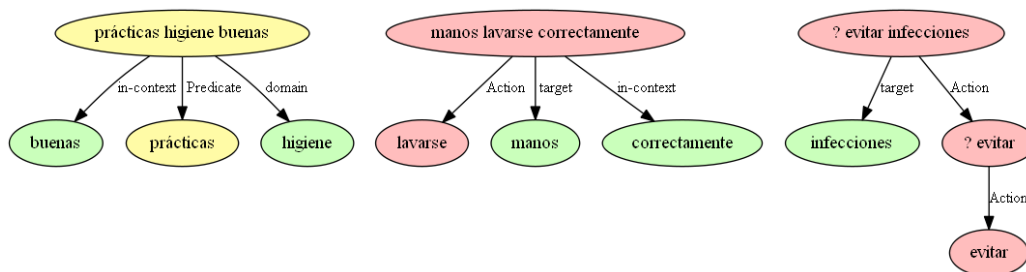


Figura 3.10: Ejemplo 2: grafo de conocimiento luego de realizado el punto 3.

Finalmente, al llevar a cabo el punto 4, son agregadas las relaciones que posibilitan el descubrimiento de conocimiento implícito en el corpus. El grafo de conocimiento resultante de este ejemplo es:

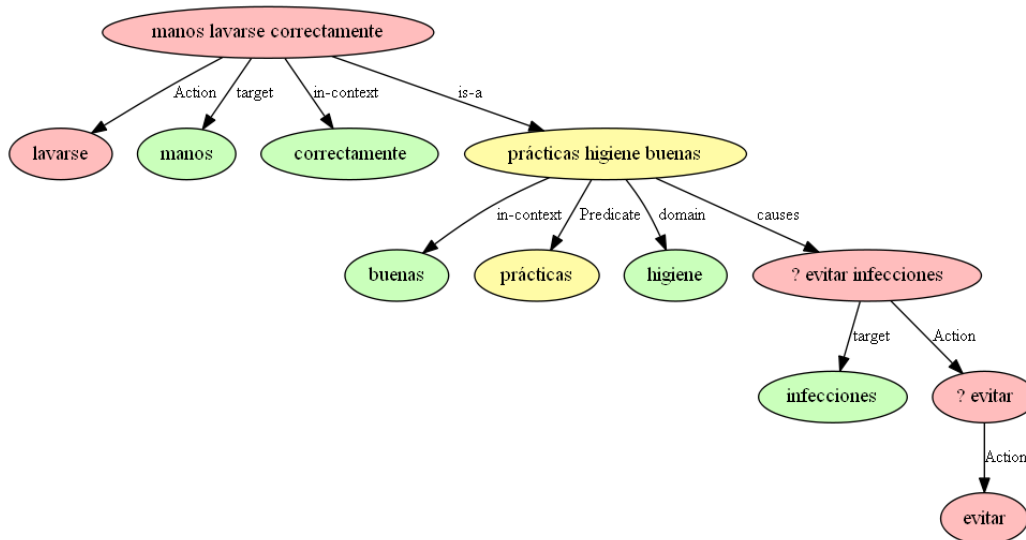


Figura 3.11: Ejemplo 2: grafo de conocimiento luego de realizado el punto 4.

Aunque el ejemplo anterior es sencillo, de igual forma que en el primer ejemplo, se puede descubrir conocimiento implícito. En esta ocasión se infiere que «manos lavarse correctamente *causes* ? evitar infecciones». Este conocimiento es traducido al lenguaje natural como “lavarse las manos correctamente causa que puedan evitarse infecciones”.

Como pudo apreciarse en el ejemplo anterior, se optó por mostrar los atributos en el grafo a través de caracteres en vez de la palabra en sí. Son usados los siguientes caracteres:

- ⊃ negación
- ? incertidumbre
- ↓ disminución
- ↑ énfasis



### 3.3.3. Resumen del algoritmo

Para llevar a cabo el punto 1 en el orden previamente expuesto, se crea una *entidad simple* por cada concepto existente en el documento de anotación. Esto sienta las bases para la posterior realización y correctitud del algoritmo expuesto en la sección anterior.

Para satisfacer lo propuesto en el punto 2, cumplen un papel protagónico las entidades de los conceptos que tienen atributos asociados. En este punto, todas estas son del tipo *entidad simple* y cada una de ellas se une con todos sus respectivos atributos, formando una *entidad con atributo*.

En el paso 3 tienen lugar algunas de las relaciones. Cada una de ellas conforma una *entidad compuesta*. Esta nueva instancia se relaciona con los conceptos de las partes derecha de dichas relaciones, ahora representados en alguno de los tres tipos de clases de esta ontología, a través del tipo de relación. A la vez que se relaciona con la parte izquierda de estas por medio del tipo de entidad que sean.

El paso 4 no crea instancias nuevas, solo establece la relación entre dos instancias creadas previamente en el grafo, aportando así conocimiento al mismo.

## 3.4. Alineación de términos

En el marco del ámbito social y mundial en que fue hecha esta investigación y por motivos principalmente de recursos, no pudo llevarse a cabo una investigación completa de este tema. En su lugar, se realizó un enfoque básico usando lematización y alineación de términos. Esto se logra por medio de la normalización de palabras o frases usadas como conceptos en el corpus anotado, una vez vayan a ser representados en la base de conocimiento. Además, se ofrecen estadísticas y resultados en este acercamiento al problema.

Con el paso del tiempo, la resolución de correferencias pasó de estar plenamente involucrado con este estudio a ser una recomendación para el futuro. Este problema es un primer acercamiento que mejorará la calidad y cantidad de conocimiento implícito descubierto, a la vez de mejorar los resultados alcanzados en esta investigación. De esta manera se dejan abiertas las puertas para la continuación y mejora de lo que aquí se presenta.

# Capítulo 4

## Análisis de Resultados

El esquema de anotación y el modelo ontológico presentado en este trabajo, y por tanto el grafo de conocimiento generado a partir de ellos, tienen como objetivo fundamental asistir en el desarrollo de sistemas de descubrimiento de conocimiento en documentos escritos en lenguaje natural.

En este capítulo, se presenta el marco experimental diseñado para comprobar la efectividad del esquema de anotación descrito en el capítulo 2 y del modelo ontológico y la propuesta de solución presentados en el capítulo 3.

### 4.1. Marco experimental

En esta investigación solo se trabajará con los artículos del corpus de *Medline* en español, estos son procesados para eliminar las marcas específicas de *HTML*<sup>6</sup> y ser divididos en oraciones. Luego de ser anotadas son mezcladas con sus respectivos artículos. Potencialmente, un artículo podrá no tener ninguna de sus oraciones anotadas o estarlo completamente.

Las tablas 4.1 y 4.2 muestran algunas estadísticas acerca de este corpus y de las oraciones anotadas pertenecientes al mismo. Estas cifras son las estadísticas del corpus sin haber aplicado el algoritmo de generar la base de conocimiento. Es decir, solo tomando los conceptos, atributos y relaciones que se encuentran en los documentos anotados. Cabe destacar que estas estadísticas pueden variar una vez

---

<sup>6</sup>Siglas en inglés de *HyperText Markup Language*, un lenguaje de marcado usado en la elaboración de páginas web.

ejecutado el algoritmo, ya sea por la creación de nuevas entidades, como se vio en la sección 3.3.3 o por el hecho de normalizar las palabras o frases, y de esta forma, muchos conceptos pueden llegar a ser la misma entidad.

Estos resultados son extraídos usando *python* [79] como lenguaje de programación y los paquetes *nlk* [64] y *spacy* [29] para el procesamiento del lenguaje natural.

Métrica	Total	
Oraciones	999	
Conceptos	6,324	% conceptos
Concept	3,914	$\approx 61.89$
Action	1,661	$\approx 26.27$
Reference	213	$\approx 3.37$
Predicate	536	$\approx 8.47$
Relaciones	5,925	% relaciones
Subject	859	$\approx 14.5$
Target	1,688	$\approx 28.49$
Domain	346	$\approx 5.84$
Argument	333	$\approx 5.62$
Is-a	570	$\approx 9.62$
Part-of	95	$\approx 1.6$
Same-as	124	$\approx 2.09$
Has-property	168	$\approx 2.84$
Causes	381	$\approx 6.43$
Entails	170	$\approx 2.87$
In-time	154	$\approx 2.6$
In-place	384	$\approx 6.48$
In-context	653	$\approx 11.02$
Atributos	559	% atributos
Negated	160	$\approx 28.62$
Uncertain	262	$\approx 46.87$
Diminished	17	$\approx 3.04$
Emphasized	120	$\approx 21.47$

Tabla 4.1: Estadísticas del corpus anotado.

<b>Métrica</b>	<b>Medline</b>	<b>Anotado</b>	<b>% anotado</b>
Artículos	1,013	25*	≈ 2.47
Oraciones	12,830	999	≈ 7.79
Promedio de oraciones por artículo	≈ 13	≈ 40	≈ 307.69
Menor cantidad de oraciones en un artículo	2	39	1,950
Artículos con la menor cantidad de oraciones	9	1	≈ 11.11
Mayor cantidad de oraciones en un artículo	65	40	≈ 61.54
Artículos con la mayor cantidad de oraciones	1	24	2,400
Palabras	191,256	14,529	≈ 7.6
Promedio de palabras por artículo	≈ 189	≈ 581	≈ 307.41
Promedio de palabras por oración	≈ 15	≈ 15	100
Menor cantidad de palabras en un artículo	33	489	≈ 1,481.82
Artículos con la menor cantidad de palabras	1	1	100
Menor cantidad de palabras en una oración	1	4	400
Oraciones con la menor cantidad de palabras	87	1	≈ 1.15
Mayor cantidad de palabras en un artículo	1,199	671	≈ 55.96
Artículos con la mayor cantidad de palabras	1	1	100
Mayor cantidad de palabras en una oración	258	46	≈ 17.83
Oraciones con la mayor cantidad de palabras	1	1	100

\*Esta cifra no son artículos en sí, sino archivos, los cuales pueden contener oraciones de varios artículos.

Tabla 4.2: Estadísticas del corpus tomado de *Medline* y del anotado.

## 4.2. Resultados computacionales

En la figura 4.1 se evidencia una relación entre el grado de salida de los nodos y la cantidad de estos que tienen un grado específico. Esto representa la cantidad de relaciones como las explicadas en la sección 3.3.3 en las que un nodo es parte izquierda (referenciado a través de Arg1).

En la figura 4.2 se evidencia una relación entre el grado de salida de los nodos y la cantidad de estos que tienen un grado específico. Esto representa la cantidad de relaciones como las explicadas en la sección 3.3.3 en las que un nodo es parte derecha (referenciado a través de Arg2).

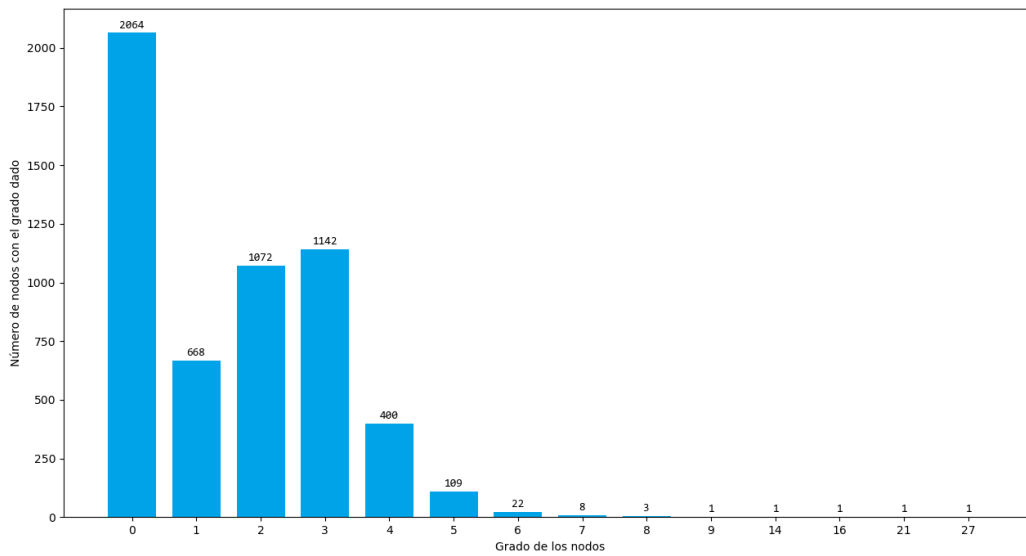


Figura 4.1: Grado de salida de los nodos del grafo.

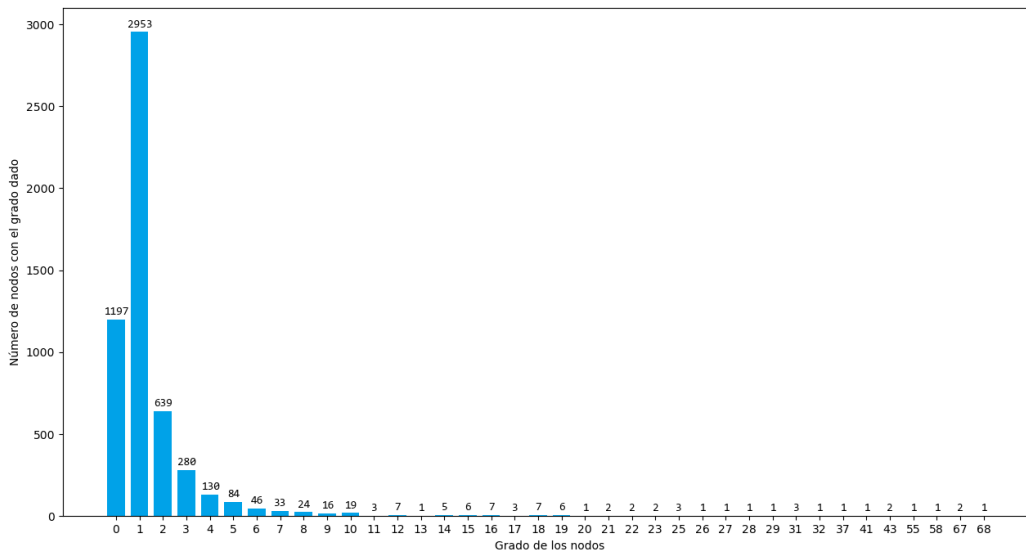


Figura 4.2: Grado de entrada de los nodos del grafo.

En la figura 4.3 se evidencia una relación entre el grado de salida de los nodos y la cantidad de estos que tienen un grado específico, pero esta vez agrupados por su rol semántico. Esto representa la cantidad de relaciones como las explicadas en la sección 3.3.3 en las que un nodo es parte izquierda (referenciado a través de Arg1).

En la figura 4.4 se evidencia una relación entre el grado de salida de los nodos y la cantidad de estos que tienen un grado específico, pero esta vez agrupados por su rol semántico. Esto representa la cantidad de relaciones como las explicadas en la sección 3.3.3 en las que el nodo es parte derecha (referenciado a través de Arg2).

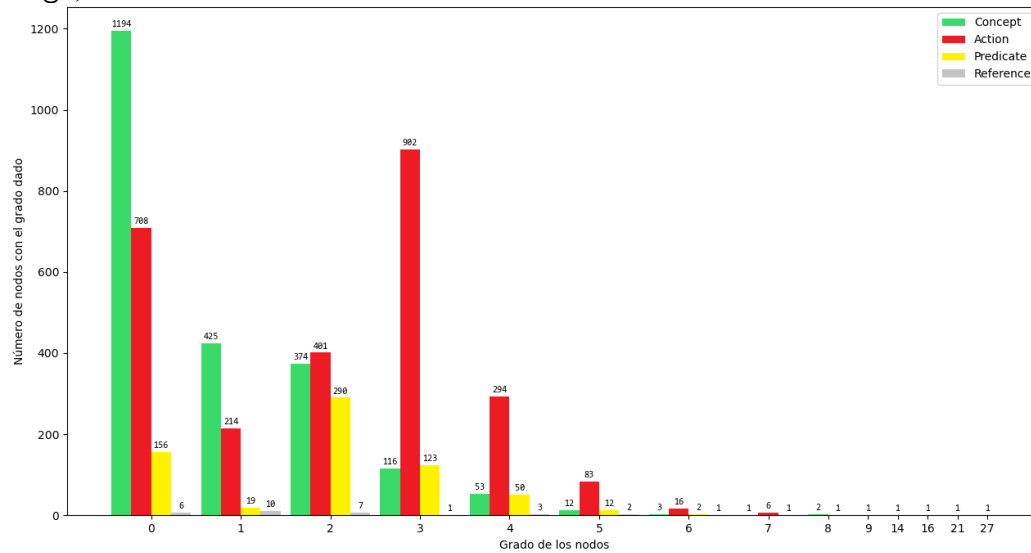


Figura 4.3: Grado de salida de los nodos del grafo por rol.

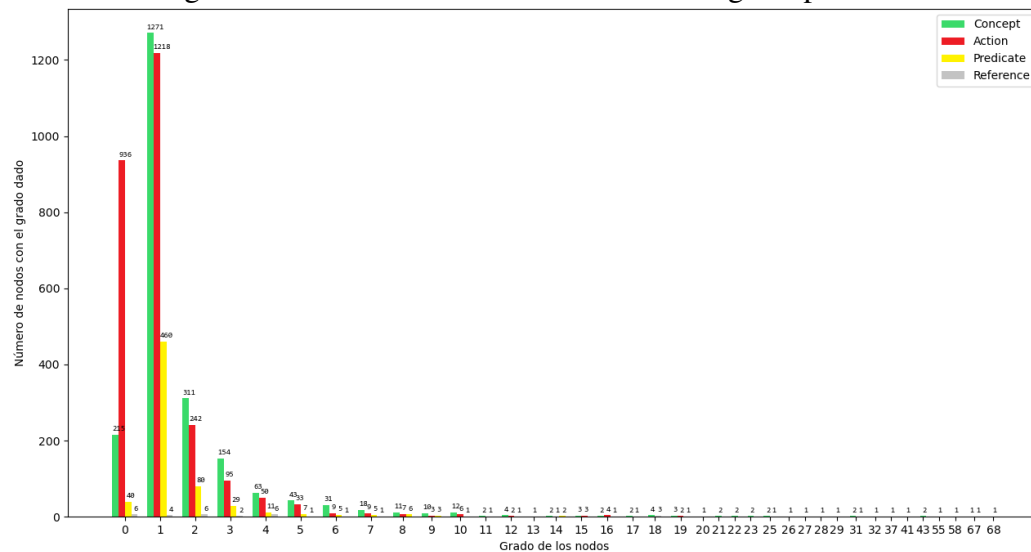


Figura 4.4: Grado de entrada de los nodos del grafo por rol.

En la tabla 4.3 se aprecia la cantidad de nodos y aristas según su tipo y a la misma vez se comparan los resultados obtenidos sin normalizar y normalizando las palabras respectivamente. La última columna representa el porcentaje de disminución en cada fila luego de normalizar.

<b>Métrica</b>	<b>Sin normalizar</b>	<b>Normalizando</b>	<b>% disminuido</b>
Conceptos	5,493	4,935	$\approx 10.16$
Concept	2,181	1,969	$\approx 9.72$
Action	2,625	2,335	$\approx 11.05$
Reference	35	21	40
Predicate	652	610	$\approx 6.44$
Relaciones	8,682	8,623	$\approx 0.68$
Concept	530	525	$\approx 0.94$
Reference	9	9	0
Action	1,902	1,875	$\approx 1.42$
Subject	923	922	$\approx 0.11$
Target	1,572	1,568	$\approx 0.25$
Predicate	501	496	$\approx 1$
Domain	298	296	$\approx 0.67$
Argument	308	307	$\approx 0.32$
Is-a	492	481	$\approx 2.24$
Part-of	89	89	0
Same-as	231	231	0
Has-property	143	143	0
Causes	360	360	0
Entails	200	199	0.5
In-time	154	154	0
In-place	361	360	$\approx 0.28$
In-context	609	608	$\approx 0.16$
Atributos	359	329	$\approx 8.36$
Negated	111	94	$\approx 15.32$
Uncertain	150	141	6
Diminished	83	80	$\approx 3.61$
Emphasized	15	14	$\approx 6.67$

Tabla 4.3: Estadísticas del grafo de conocimiento.

Como se ha podido observar anteriormente, una de las principales funciones de las bases de conocimiento es la extracción de conocimiento implícito. En la tabla 4.4 puede verse un ejemplo de esto en el corpus usado en esta investigación.

Documento	Relación explícita
cirugía.ann	«cirugía de corazón <i>is-a</i> operación»
hígado graso.ann	«operación <i>is-a</i> procedimiento médico»
Relación implícita	
«cirugía de corazón <i>is-a</i> procedimiento médico»	

Tabla 4.4: Ejemplo de extracción de conocimiento implícito.

Para el problema en general de aprendizaje de ontologías se pueden llevar a cabo varias métricas de evaluación y metodologías, por ejemplo OntoRand [6] y OntoMetric [53]. Como se vio en la sección 1.2, uno de los enfoques comúnmente utilizados para la evaluación de ontologías es la *evaluación basada en datos*: esta fue la evaluación que se realizó.

La tabla 4.5 muestra la división realizada entre las oraciones del corpus anotado y los resultados alcanzados con esto, en promedio, luego de 250 corridas de cada división.



<b>% de división</b>	<b>70</b>	<b>75</b>	<b>80</b>	<b>85</b>	<b>90</b>	<b>95</b>
oraciones	959	959	959	959	959	959
oraciones de entrenamiento	671	719	767	815	863	911
oraciones de verificación	288	240	192	144	96	48
entidades en el corpus	3,725	3,938	4,135	4,345	4,544	4,739
entidades de verificación	1,888	1,621	1,353	1,058	746	405
coincidencia de entidades	678	623	554	468	355	209
% de coincidencia de entidades	35.91	38.43	40.95	44.23	47.59	51.6
nuevas entidades	1,210	998	799	590	391	196
% de nuevas entidades	64.09	61.57	59.05	55.77	52.41	48.4
relaciones en el corpus	6,139	6,560	6,965	7,390	7,802	8,213
relaciones de verificación	2,728	2,283	1,845	1,390	932	469
coincidencia de relaciones	244	220	187	156	111	59
% de coincidencia de relaciones	8.94	9.64	10.14	11.22	11.91	12.58
nuevas relaciones por nuevas entidades	2,416	2,003	1,609	1,195	795	397
% de nuevas relaciones por nuevas entidades	88.56	87.74	87.21	85.97	85.3	84.65
nuevas relaciones en entidades existentes	68	60	49	39	26	13
% de nuevas relaciones en entidades existentes	2.5	2.62	2.65	2.81	2.79	2.77
% de coincidencia de relaciones válidas <sup>7</sup>	78.21	78.57	79.24	80	81.02	81.94

Tabla 4.5: Ejemplo de extracción de conocimiento implícito.

<sup>7</sup>Las relaciones válidas son aquellas entre entidades ya existentes en el corpus, pues si la entidad no existe es obvio que habrá que crearla y por tanto, la relación será un fallo seguro a la hora de comprobar si existe o no en el grafo de conocimiento de entrenamiento.

### 4.3. Discusión

Todos los nodos del grafo de conocimiento participan en al menos una relación, pero como se aprecia en las figuras 4.1, 4.2, 4.3 y 4.4, hay muchos nodos que tienen un grado bajo.

En el caso en que la gráfica muestra la cantidad de nodos con grado cero, esto viene dado o bien porque ese nodo no tiene relaciones de salida y en este caso tendría grado de salida cero o bien no tiene relaciones de entrada y por tanto grado de entrada cero. El hecho de que pocos nodos tengan un alto grado viene dado porque usualmente los nodos más grandes y con más palabras, al ser conocimiento más específico, pocos de ellos participan en un mayor número de relaciones.

Los roles semánticos *Concept* y *Action* que participan en una mayor cantidad de relaciones en este grafo de conocimiento son «persona» y «tratamiento» respectivamente. Dado que se trabajó con un corpus de documentos médicos, este resultado cobra sentido pues esas palabras son ampliamente empleadas en este medio.

En la tabla 4.3 se muestra el resultado de un grafo de conocimiento utilizando palabras o frases sin normalizar y normalizadas. Normalizar las palabras no solo reduce la cantidad de nodos y aristas en el grafo sino que también potencialmente aumenta la cantidad de conocimiento implícito que puede ser extraído. Esto sucede debido a que al fusionar nodos, la cantidad de caminos en el grafo de conocimiento aumentan. Por otra parte, todas las relaciones explícitamente escritas en el corpus representan dos nodos y una arista entre estos. Por tanto, si dos nodos no tienen aristas entre ellos, pero existe un camino que los conecta, esto es conocimiento implícito descubierto a través del grafo.

El hecho de normalizar implica, por ejemplo, que todas las conjugaciones de un mismo verbo deben resultar en el propio verbo sin conjugar. Lo mismo sucede para el resto de palabras del idioma. Mientras mayor sea la cantidad de palabras normalizadas, hay una mejor representación en el grafo de su conocimiento expuesto en el corpus. Obviamente, es de vital importancia la normalización de familias de palabras a su primitiva, para poder dar continuidad al significado que ellas representan, puesto que no tiene sentido normalizar «glóbulos rojos» a «globo rojo» o «glóbulo blanco».

Una deficiencia clara sucedió a la hora de hallar los resultados de la tabla 4.5. Para este tipo de evaluación, lo ideal es tener un grafo de conocimiento formado a partir de una ontología y de un corpus preferentemente grande. A su vez, el grafo

debe ser revisado con otro corpus perteneciente al mismo tema y de mediano o gran tamaño.

Para llevar a cabo esta tarea, se dividieron las oraciones anotadas en dos grupos, un grupo de *training* (*entrenamiento* en español) con el cual se construyó el grafo de conocimiento y un grupo de *testing* (*verificación* en español), con el que se revisó la existencia de las anotaciones de texto y las relaciones respecto a las que ya están construidas en el grafo.

Como puede verse en la tabla 4.5, a medida que el corpus de entrenamiento tiene más entidades y relaciones, se logró un mayor porcentaje de coincidencia respecto a la información con la que se comprobaba la base de conocimiento ya formada. Esto se traduce en que la base de conocimiento es capaz de condensar lo aprendido, logrado a través de la normalización de palabras y frases. Además, este resultado también evidencia que mientras más grande sea el corpus anotado inicial, mayor y de mejor calidad será el conocimiento representado en el grafo creado siguiendo los pasos propuestos en esta investigación.

En adición a esto, las palabras que participan en una mayor cantidad de relaciones en la base de conocimiento generada se muestran a continuación de manera descendente.

Representados como *Concept* en el corpus anotado, los nodos del grafo de conocimiento que más relaciones tienen son:

- tratamiento
- persona
- cuerpo
- problema
- médico
- sangre
- síntoma
- niño

Representados como *Action* en el corpus anotado, los nodos del grafo de conocimiento que más relaciones tienen son:

- tener
- usar
- dolor
- riesgo
- utilizar
- ayudar
- causar
- dañar

Estas palabras evidentemente están ampliamente correlacionadas con el dominio médico, al que pertenece el corpus usado. De este resultado se interpreta que los nodos con mayor cantidad de relación en el grafo son nodos con conocimiento válido, consistentes y de calidad en el dominio específico y no con conocimiento espurio.

# Conclusiones

Esta investigación propone un conjunto de elementos orientados al descubrimiento de conocimiento en textos del lenguaje natural. La propuesta se centra en el idioma español y el dominio de la salud, pero es generalizable en ambos aspectos. Entre las contribuciones fundamentales de esta investigación destacan:

- (1) La definición de un modelo de anotación de propósito general que logra capturar los rasgos semánticos más relevantes contenidos en documentos de texto plano. El mismo es usado como base en la construcción de la ontología propuesta.
- (2) La definición de un formato de anotación de archivos para el esquema conceptual previamente definido.
- (3) Se diseñó una propuesta de ontología donde se puede representar un corpus de documentos escritos en lenguaje natural.
- (4) Se implementó un algoritmo computacional para representar un corpus anotado como grafo de conocimiento a través de dicha ontología.

A menudo, una ontología de un dominio no es un objetivo en sí misma. Desarrollarla es similar a definir un conjunto de datos y su estructura para que los utilicen otros programas. Los métodos de resolución de problemas, las aplicaciones independientes del dominio y los usuarios, a menudo las utilizan como datos, en conjunto con bases de conocimiento creadas a partir de las mismas. Por ejemplo, en esta investigación se desarrolla una ontología de dominio médico, la cual se puede utilizar como base para algunas aplicaciones que ofrezcan un conjunto de herramientas de gestión de la salud.

En los resultados se demostró el descubrimiento de conocimiento implícito en el corpus. Esto trae consigo disímiles ventajas, desde el propio descubrimiento de

este conocimiento, hasta la interpretación y aprendizaje de un gran número de textos escritos en lenguaje natural, en apenas segundos, mediante el uso de un equipo de cómputo y las propuestas ofrecidas en esta investigación. El conocimiento extraído podría ser usado posteriormente por especialistas en el tema o usuarios, y de esta manera ahorrar el tiempo que tomaría la lectura e interpretación del propio corpus usado para esta tarea.

Teniendo en cuenta que un humano debe tener una gran capacidad de memorización para poder recordar todo lo aprendido en un corpus de documentos, la traba que puede ocasionar tenerlo escrito en un idioma que no se domine, y el factor de no olvidar lo aprendido de él al pasar el tiempo, es clave la utilización de un equipo de cómputo para la creación de la base de conocimiento respectiva al corpus. La misma puede ser fácilmente guardada, leída y usada en el propio sistema o incluso en otros, aportando gran versatilidad al uso de las técnicas mostradas en este estudio.

El descubrimiento automático de conocimiento en el dominio médico tiene especial importancia, pues permitiría identificar interacciones ocultas en la literatura. Además, a pesar de que los recursos médicos disponibles en idioma español son abundantes, los recursos necesarios para la creación de sistemas de extracción automática son más escasos que en otros idiomas, por lo cual la construcción de una ontología y un grafo de conocimiento basado en un corpus del propio dominio, constituyen un hecho relevante para el desarrollo de nuevos sistemas y la continuación de esta investigación en un futuro.

## Recomendaciones

A pesar de que esta investigación está orientada hacia el descubrimiento de conocimiento en documentos médicos y del idioma español, el modelo de anotación y la ontología propuesta son de propósito general. Esto permite su aplicación en otros dominios e idiomas. Se propone la anotación de corpus de otros dominios en el modelo de anotación definido en este estudio. Al mismo tiempo, tendría gran connotación la creación de grafos de conocimiento mediante la utilización de estos corpus y basados en el modelo ontológico ofrecido.

Se propone comprobar la efectividad de la propuesta de solución ofrecida, pues como se mencionó anteriormente, una deficiencia clara a la hora de calcular las estadísticas expuestas en la tabla 4.5 es que se usó un corpus muy pequeño para crear el grafo de conocimiento y la no existencia de uno independiente pero del mismo dominio y anotado con el formato de modelo propuesto para la posterior validación de la base de conocimiento creada.

La resolución de referencias y correferencias mejoraría en gran medida el descubrimiento de conocimiento implícito en el grafo y debe mejorar los resultados obtenidos. Esta es una tarea que usualmente se intenta resolver usando inteligencia artificial y es un reto que se propone para trabajo futuro, dando continuidad a la línea de investigación presentada en este trabajo.

Otra de las propuestas consideradas es la creación de un grafo de conocimiento a partir de un corpus de dominio específico, siguiendo la línea de investigación de este estudio. A su vez, fomentar el análisis de este corpus en un grupo de expertos en el dominio, y de esta manera corroborar cuán relevante es el conocimiento implícito descubierto a través del grafo resultante en comparación al extraído por los especialistas.

Además se propone la creación de una aplicación para computadoras, móviles y/o páginas web, la cual podría ofrecer sugerencias de enfermedades dados los síntomas especificados por el usuario, y a su vez, posibles tratamientos para

las mismas. Esto sería posible mediante la utilización del grafo de conocimiento creado a partir del corpus usado en esta investigación, el cual es de dominio médico.

# Bibliografía

- [1] ALANI, H., KIM, S., MILLARD, D. E., WEAL, M. J., HALL, W., LEWIS, P. H., AND SHADBOLT, N. R. Automatic ontology-based knowledge extraction from web documents. *IEEE Intelligent Systems* 18, 1 (2003), 14–21. (Citado en la página 9).
- [2] ASIM, M. N., WASIM, M., GHANI KHAN, M. U., MAHMOOD, W., AND ABBASI, H. M. A survey of ontology learning techniques and applications. *Database* 2018 (2018). (Citado en las páginas 12, 13 y 15).
- [3] BARBU, E. Property type distribution in Wordnet, corpora and Wikipedia. *Expert Systems with Applications* 42, 7 (2015), 3501–3507. (Citado en la página 9).
- [4] BIAN, H., AND HA, S. Conceptual extraction of domain knowledge graph in different data sources. *Conference of DEStech Transactions on Computer Science and Engineering* (2017). (Citado en la página 9).
- [5] BRANK, J., MLADENIC, D., AND GROBELNIK, M. A survey of ontology evaluation techniques. *Proceedings of the conference on data mining and data warehouses (SiKDD 2005)* (2005), 166–170. (Citado en las páginas 11 y 14).
- [6] BRANK, J., MLADENIC, D., AND GROBELNIK, M. Gold standard based ontology evaluation using instance assignment. In *Workshop on Evaluation of Ontologies for the Web, EON* (2006). (Citado en la página 57).
- [7] BREWSTER, C., ALANI, H., DASMAHAPATRA, S., AND WILKS, Y. Data driven ontology evaluation, 2004. (Citado en la página 14).



- [8] BREWSTER, C., CIRAVEGNA, F., AND WILKS, Y. User-centred ontology learning for knowledge management. *International Conference on Application of Natural Language to Information Systems* (2002), 203–207. (Citado en la página 9).
- [9] BRICKLEY, D., AND GUHA, R. V. Resource Description Framework (RDF) Schema Specification. World Wide Web Consortium: <http://www.w3.org/TR/PR-rdf-schema>. (Citado en la página 10).
- [10] BUITELAAR, P., CIMIANO, P., RACIOPPA, S., AND SIEGEL, M. Ontology-based information extraction with soba. *Proceedings of the International Conference on Language Resources and Evaluation (LREC)* (2006). (Citado en la página 9).
- [11] BUITELAAR, P., AND SINTEK, M. OntoLT version 1.0: Middleware for ontology extraction from text. *Proceedings of the Demo Session at the International Semantic Web Conference* (2004). (Citado en la página 9).
- [12] CASTANO, S., ESPINOSA, S., FERRARA, A., KARKALETSIS, V., KAYA, A., MELZER, S., MÖLLER, R., MONTANELLI, S., AND PETASIS, G. Ontology dynamics with multimedia information: The boemie evolution methodology. *International Workshop on Ontology Dynamics (IWOD-07)* (2007), 41. (Citado en la página 9).
- [13] CHALENDAR, G. D., AND GRAU, B. SVETLAN’: a system to classify nouns in context. *Proceedings of the First International Conference on Ontology Learning 31* (2000), 19–24. (Citado en la página 8).
- [14] CHAPULSKY, H., HOVY, E., AND RUSS, T. Progress on an automatic ontology alignment methodology. ANSI Ad Hoc Group on Ontology Standards. (Citado en la página 7).
- [15] CIMIANO, P., AND STAAB, S. Learning concept hierarchies from text with a guided agglomerative clustering algorithm. *Proceedings of the ICML 2005 Workshop on Learning and Extending Lexical Ontologies with Machine Learning Methods* (2005). (Citado en la página 9).
- [16] CIMIANO, P., AND VÖLKER, J. text2onto. *International Conference on Application of Natural Language to Information Systems, Springer* (2005), 227–238. (Citado en la página 9).

- [17] CONSUEGRA AYALA, J. P. Descubrimiento de Conocimiento en Documentos Clínicos: Un Enfoque Basado en Aprendizaje Profundo, Dec. 2019. (Citado en la página 37).
- [18] CORCOGLIONITI, F., ROSPOCHER, M., AND APROSIO, A. P. Frame-based ontology population with PIKES. *IEEE Transactions on Knowledge and Data Engineering* 28, 12 (2016), 3261–3275. (Citado en la página 13).
- [19] CRAVEN, M., DIPASQUO, D., FREITANG, D., MCCALLUM, A., MITCHELL, T., NIGAM, K., AND SLATTERY, S. Learning to extract symbolic knowledge from the World Wide Web. *AAAI’98* (1998), 509–516. (Citado en la página 8).
- [20] CRAVEN, M., DIPASQUO, D., FREITANG, D., MCCALLUM, A., MITCHELL, T., NIGAM, K., AND SLATTERY, S. Learning to construct knowledge bases from the World Wide Web. *Artificial Intelligence* 118 (2000), 69–113. (Citado en la página 8).
- [21] CULLEN, J., AND BRYMAN, A. The knowledge acquisition bottleneck: time for reassessment? *Expert Systems* 5, 3 (1988), 216–255. (Citado en la página 6).
- [22] DE SILVA, T. S., MACDONALD, D., PATERSON, G., SIKDAR, K. C., AND COCHRANE, B. Systematized nomenclature of medicine clinical terms (SNOMED CT) to represent computed tomography procedures. *Computer methods and programs in biomedicine* 101, 3 (2011), 324–329. (Citado en la página 11).
- [23] DEB, C. K., MARWAHA, S., ARORA, A., AND DAS, M. A framework for ontology learning from taxonomic data. *Big Data Analytics* (2018), 29–37. (Citado en la página 8).
- [24] DELLSCHAFT, K., AND STAAB, S. Strategies for the Evaluation of Ontology Learning. *Ontology Learning and Population* 167 (2008), 253–272. (Citado en las páginas 14 y 15).
- [25] DRYMONAS, E., ZERVANOU, K., AND PETRAKIS, E. G. Unsupervised Ontology Acquisition from Plain Texts: The OntoGain System. *International Conference on Application of Natural Language to Information Systems* (2010), 277–287. (Citado en la página 10).

- [26] ESTEVEZ-VELARDE, S., GUTIÉRREZ, Y., MONTOYO, A., PIAD-MORFFIS, A., MUÑOZ, R., AND ALMEIDA-CRUZ, Y. Gathering object interactions as semantic knowledge. *Proceedings on the International Conference on Artificial Intelligence (ICAI)* (2018), 363–369. (Citado en las páginas 10, 11 y 17).
- [27] ESTEVEZ-VELARDE, S., PIAD-MORFFIS, A., ALMEIDA-CRUZ, Y., GUTIÉRREZ, Y., MUÑOZ, R., AND MONTOYO, A. Framework for Continuous Knowledge Discovery using Ontology Enrichment and Population. 2020. (Citado en las páginas 7, 9, 12, 13 y 14).
- [28] ETZIONI, O., CAFARELLA, M., DOWNEY, D., KOK, S., POPESCU, A., SHAKED, T., SODERLAND, S., WELD, D. S., AND YATES, A. Web-scale information extraction in knowitall: (preliminary results). *Proceedings of the 13th international conference on World Wide Web* (2004), 100–110. (Citado en la página 9).
- [29] EXPLOSION AI. spaCy. <https://spacy.io>. (Citado en la página 52).
- [30] FAURE, D., AND POIBEAU, T. First experiments of using semantic knowledge learned by ASIUM for information extraction task using INTEX. *Ontology Learning ECAI-2000 Workshop*, Citeseer, IOS Press. (Citado en la página 8).
- [31] GANGULY, P., CHATTOPADHYAY, S., PARAMESH, N., AND RAY, P. An ontology-based framework for managing semantic interoperability issues in e-health. *HealthCom 2008-10th International Conference on e-health Networking, Applications and Services* (2008), 73–78. (Citado en la página 11).
- [32] GRUBER, T. R. A Translation Approach to Portable Ontology Specification. *Knowledge Acquisition* 5 (1993), 199–220. (Citado en la página 3).
- [33] GUARINO, N. Formal ontology in information systems. *Proceedings of the first international conference (FOIS'98)*, June 1998. (Citado en la página 2).
- [34] GUREVYCH, I., MALAKA, R., PORZEL, R., AND ZORN, H. Semantic coherence scoring using an ontology. *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics* (2003), 88–95. (Citado en la página 13).

- [35] HAHN, U., AND ROMACKER, M. Content management in the SYNDIKATE system - how technical documents are automatically transformed to text knowledge bases. *Data & Knowledge Engineering* 35, 2 (2000), 137–159. (Citado en la página 8).
- [36] HAHN, U., AND ROMACKER, M. The SYNDIKATE text Knowledge base generator. *Proceedings of the First International Conference on Human Language Technology Research* (2001), 1–6. (Citado en la página 8).
- [37] HEARST, M. A. Automatic acquisition of hyponyms from large text corpora. *Association for Computational Linguistics* (1992), 539–545. (Citado en la página 23).
- [38] HENDLER, J., AND MCGUINNESS, D. L. The DARPA Agent Markup Language. *IEEE Intelligent Systems* 16, 6 (2000), 67–73. (Citado en la página 10).
- [39] HUMPHREYS, B. L., AND LINDBERG, D. A. B. The UMLS project: making the conceptual connection between users and the information they need. *The UMLS project: making the conceptual connection between users and the information they need. Bulletin of the Medical Library Association* 81, 2 (1993), 170. (Citado en la página 11).
- [40] IVANOVIĆ, M., AND BUDIMAC, Z. An overview of ontologies and data resources in medical domains. *Expert Systems with Applications* 41, 11 (2014), 5158–5166. (Citado en la página 10).
- [41] JAIN, S., JAIN, N. K., AND MISHRA, S. EHCPRS system as an ontology learning system. *2015 Second International Conference on Computing for Sustainable Global Development (INDIACom)* (2015), 978–984. (Citado en la página 9).
- [42] JIANG, X., AND TAN, A. CRCTOL: A semantic-based domain ontology learning system. *Journal of the American Society for Information Science and Technology* 61, 1 (2010), 150–168. (Citado en la página 10).
- [43] KIM, K., AND CHOI, H. Design of a clinical knowledge base for heart disease detection. *7th IEEE International Conference on Computer and Information Technology (CIT 2007)* (2007), 610–615. (Citado en la página 11).

- [44] KIM, S., SON, J., PARK, S., PARK, S., CHANGKI, L., WANG, J., JANG, M., AND PARK, H. Optima: An ontology population system. *3rd Workshop on Ontology Learning and Population* (July 2008). (Citado en la página 9).
- [45] KNIGHT, K., AND LUK, S. K. Building a large-scale knowledge base for machine translation. *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI'94)* (1994), 773–778. (Citado en la página 7).
- [46] KÖHLER, S., DOELKEN, S. C., MUNGALL, C. J., BAUER, S., FIRTH, H. V., ISABELLE, B., BLACK, G. C., BROWN, D. L., BRUDNO, M., CAMPBELL, J., FITZPATRICK, D. R., EPPIG, J. T., JACKSON, A. P., FRESON, K., GIRDEA, M., HELBIG, I., HURST, J. A., JÄHN, J., JACKSON, L. G., KELLY, A. M., LEDBETTER, D. H., MANSOUR, S., MARTIN, C. L., MOSS, C., MUMFORD, A., OUWEHAND, W. H., PARK, S., RIGGS, E. R., SCOTT, R. H., SISODIYA, S., VOOREN, S. V., WAPNER, R. J., WILKIE, A. O., WRIGHT, CAROLINE F., V. S. A. T., LEEUW, N. D., DE VRIES, B. B., WASHINGTON, N. L., SMITH, C. L., WESTERFIELD, M., SCHOFIELD, P., RUEF, B. J., GKOUTOS, G. V., HAENDEL, M., SMEDLEY, D., LEWIS, S. E., AND ROBINSON, P. N. The Human Phenotype Ontology project: linking molecular biology and disease through phenotype data. *Nucleic acids research* 42, D1 (2014), D966–D974. (Citado en la página 11).
- [47] KRUMMENACHER, R., SIMPERL, E., CERIZZA, D., VALLE, E. D., NIXON, L. J., AND FOXVOG, D. Enabling the European patient summary through triplesaces. *Computer methods and programs in biomedicine* 95, 2 (2009), S33–S43. (Citado en la página 11).
- [48] LABROU, Y., AND FININ, T. Yahoo! as an ontology: using Yahoo! categories to describe documents. *Proceedings of the eighth international conference on Information and knowledge management* (1999), 180–187. (Citado en la página 10).
- [49] LACHER, M. S., AND GROH, G. Facilitating the exchange of explicit knowledge through ontology mappings. *Proceedings of FLAIRS'2001* (2001), 305–309. (Citado en la página 7).
- [50] LEHMANN, J., ISELE, R., JAKOB, M., JENTZSCH, A., KONTOKOSTAS, D., MENDES, P. N., HELLMANN, S., MORSEY, M., KLEEF, P. V., AUER, S., AND BIZER, C. DBpedia: a large-scale, multilingual knowledge base

- extracted from Wikipedia. *Semantic web* 6, 2 (2015), 167–195. (Citado en la página 10).
- [51] LENAT, D. B. CYC: a large-scale investment in knowledge infrastructure. *Communications of the ACM* 38, 11 (1995), 33–38. (Citado en la página 7).
- [52] LOZANO-TELLO, A. Ontologías en la Web Semántica. In *I Jornada de Ingeniería Web* (2001). (Citado en la página 1).
- [53] LOZANO-TELLO, A., AND GÓMEZ-PÉREZ, A. Ontometric: A method to choose the appropriate ontology. *Journal of Database Management (JDM)* 15, 2 (2004), 1–18. (Citado en la página 57).
- [54] MAEDCHE, A., AND STAAB, S. Ontology learning for the semantic web. *IEEE Intelligent systems* 16, 2 (2001), 72–79. (Citado en la página 6).
- [55] MANNING, C. D., SCHÜTZE, H., AND RAGHAVAN, P. Introduction to information retrieval. *Cambridge university press* (2008). (Citado en la página 11).
- [56] MCGUINNESS, D. L., FIKES, R., RICE, J., AND WILDER, S. An Environment for Merging and Testing Large Ontologies. (Citado en la página 3).
- [57] MCGUINNESS, D. L., FIKES, R., RICE, J., AND WILDER, S. The chimæra ontology environment. *AAAI/IAAI 2000* (2001), 1123–1124. (Citado en la página 2).
- [58] MCGUINNESS, D. L., AND NOY, N. F. Ontology development 101: A guide to creating your first ontology. *Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880* 15, 2 (2001). (Citado en la página 10).
- [59] MCGUINNESS, D. L., AND WRIGHT, J. Conceptual Modeling for Configuration: A Description Logic-based Approach. *Artificial Intelligence for Engineering Design, Analysis, and Manufacturing - special issue on Configuration* (1998). (Citado en la página 3).
- [60] MEDLINEPLUS. <https://medlineplus.gov>. (Citado en la página 4).
- [61] MISSIKOFF, M., NAVIGLI, R., AND VELARDI, P. Integrated approach to web ontology learning and engineering. *Computer* 35, 11 (2002), 60–63. (Citado en la página 9).

- [62] MITCHELL, T., COHEN, W., HRUSCHKA, E., TALUKDAR, P., YANG, B., BETTERIDGE, J., CARLSON, A., DALVI, BHAVANA, G. M., KISIEL, B., KRISHNAMURTHY, J., LAO, N., MAZAITIS, K., MOHAMED, T., NAKASHOLE, N., PLATANIOS, E., RITTER, A., SAMADI, M., SETTLES, B., WANG, R., WIJAYA, D., GUPTA, A., CHEN, X., SAPAROV, A., GREAVES, M., AND WELLING, J. Never-ending learning. *Communications of the ACM* 61, 5 (2018), 103–115. (Citado en la página 10).
- [63] MUSEN, M. A. Dimensions of knowledge sharing and reuse. *Computers and Biomedical Research* 25 (1992), 435–467. (Citado en la página 3).
- [64] NATURAL LANGUAGE TOOLKIT. NLTK. <https://www.nltk.org>. (Citado en la página 52).
- [65] NAVIGLI, R., AND VELARDI, P. Semantic interpretation of terminological strings. *Proceedings of the Sixth International Conference on Terminology and Knowledge Engineering* (2002), 95–100. (Citado en la página 9).
- [66] NAVIGLI, R., VELARDI, P., AND GANGEMI, A. Ontology learning and its application to automated terminology translation. *IEEE Intelligent Systems* 18, 1 (2003), 22–31. (Citado en la página 9).
- [67] NOY, N. F., AND MUSEN, M. A. An algorithm for merging and aligning ontologies: Automation and tool support. *Proceedings of the Workshop on Ontology Management at the Sixteenth National Conference on Artificial Intelligence (AAAI'99)* (1999), 1999–0799. (Citado en la página 7).
- [68] NOY, N. F., AND MUSEN, M. A. PROMPT: algorithm and tool for automated ontology merging and alignment. *Proceedings of Seventeenth National Conference on Artificial Intelligence (AAAI-2000)* (2000), 450–455. (Citado en la página 7).
- [69] OLIVEIRA, A., PEREIRA, F. C., AND CARDOSO, A. Automatic reading and learning from text. *Proceedings of the international symposium on artificial intelligence (ISAI)* (2001). (Citado en la página 8).
- [70] PEREIRA, F. C., OLIVEIRA, A., AND CARDOSO, A. Extracting concept maps with clouds. *Proceedings of the Argentine symposium of artificial intelligence (ASAI)* (2000). (Citado en la página 8).

- [71] PETASIS, G., KARKALETSIS, V., PALIOURAS, G., KRITHARA, A., AND ZAVITSANOS, E. Ontology population and enrichment: State of the art. *Knowledge-driven multimedia information extraction and ontology evolution* (2011), 134–166. (Citado en la página 12).
- [72] PIAD-MORFFIS, A., ALMEIDA-CRUZ, Y., GUTIÉRREZ, Y., AND MUÑOZ, R. A Computational Ecosystem to Support eHealth Knowledge Discovery Technologies in Spanish. *Journal of Biomedical Informatics* (2020), 103517. (Citado en la página 37).
- [73] PIAD-MORFFIS, A., ALMEIDA-CRUZ, Y., GUTIÉRREZ, Y., AND MUÑOZ, R. [dataset] eHealth-KD v2, Mar. 2020. (Citado en la página 38).
- [74] PIAD-MORFFIS, A., ESTEVEZ-VELARDE, S., ALMEIDA-CRUZ, Y., GUTIÉRREZ, Y., MUÑOZ, R., AND MONTOTO, A. Overview of the eHealth Knowledge Discovery Challenge at IberLEF 2020. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2020)* (2020). (Citado en la página 37).
- [75] PIAD-MORFFIS, A., ESTEVEZ-VELARDE, S., CONSUEGRA-AYALA, J. P., ALMEIDA-CRUZ, Y., GUTIÉRREZ, Y., MUÑOZ, R., AND MONTOTO, A. Overview of the eHealth Knowledge Discovery Challenge at IberLEF 2019. In *Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019). CEUR Workshop Proceedings, CEUR-WS.org* (2019). (Citado en la página 37).
- [76] PISANELLI, D. M., GANGEMI, A., AND STEVE, G. Ontologies and information systems: the marriage of the century? *New Trends in Software Methodologies, Tools and Techniques* (2002), 125–133. (Citado en la página 1).
- [77] PODGORELEC, V., AND PAVLIC, L. Managing diagnostic process data using semantic web. *Twentieth IEEE International Symposium on Computer-Based Medical Systems (CBMS'07)* (2007), 127–134. (Citado en la página 11).
- [78] PRICE, C., AND SPACKMAN, K. SNOMED Clinical Terms. *BJHC&IM-British Journal of Healthcare Computing & Information Management* 17, 3 (2000), 27–31. (Citado en la página 11).



- 
- [79] PYTHON SOFTWARE FOUNDATION. Python. <https://www.python.org/downloads/release/python-382>. (Citado en la página 52).
- [80] ROSPOCHER, M., ERP, M. V., VOSSEN, P., FOKKENS, A., ALDABE, I., RIGAU, G., SOROA, A., PLOEGER, T., AND BOGAARD, T. Building event-centric knowledge graphs from news. *Journal of Web Semantics* 37 (2016), 132–151. (Citado en la página 14).
- [81] ROTHENFLUH, T. R., GENNARI, J., ERIKSSON, H., PUERTA, A., TU, S., AND MUSEN, M. Reusable ontologies, knowledge-acquisition tools, and performance systems: PROTÉGÉ-II solutions to Sisypheus-2. *International Journal of Human-Computer Studies* 44 (1996), 303–332. (Citado en la página 3).
- [82] RUSSELL, B. 2 ed. W. W. Norton & Company, Berlin, 1996. Reprint, first published in 1903. (Citado en la página 25).
- [83] SÁNCHEZ, D., BATET, M., MARTÍNEZ, S., AND DOMINGO-FERRER, J. Semantic variance: an intuitive measure for ontology accuracy evaluation. *Engineering Applications of Artificial Intelligence* 39 (2015), 89–99. (Citado en la página 13).
- [84] SANDUN, I., SUMATHIPALA, S., AND GANEGODA, G. U. Self-evolving disease ontology for medical domain based on web. *International Journal of Fuzzy Logic Intelligent Systems* 17, 4 (2017), 307–314. (Citado en la página 11).
- [85] SHAMSFARD, M. *Designing the ontology learning model, prototyping in a Persian text understanding system*. PhD thesis, AmirKabir University, Tehran, Iran, 2003. (Citado en la página 8).
- [86] SHAMSFARD, M., AND BARFOROUSH, A. A. A basis for evolutionary ontology construction. *Proceedings of 18th IASTED International Conference on Applied Informatics (AI'2000)* (2000), 433–438. (Citado en la página 8).
- [87] SHAMSFARD, M., AND BARFOROUSH, A. A. An introduction to Hasti: an ontology learning system. *Proceedings of the 6th Conference on Artificial Intelligence and Soft Computing (ASC'2002)* (2002), 242–247. (Citado en la página 8).
-

- [88] SHAMSFARD, M., AND BARFOROUSH, A. A. Ontology learning from natural language texts. *International Journal of Human-Computer Studies* (2002). (Citado en la página 8).
- [89] SHAMSFARD, M., AND BARFOROUSH, A. A. The state of the art in ontology learning: a framework for comparison. *The Knowledge Engineering Review* 18, 4 (2003), 293. (Citado en la página 7).
- [90] SOWA, J. F. Knowledge representation: logical, philosophical and computational foundations. Brooks/Cole Publishing Company. (Citado en la página 7).
- [91] STOERMER, H., PALMISANO, I., REDAVID, D., IANNONE, L., BOUQUET, P., AND SEMERARO, G. Contextualization of a RDF Knowledge Base in the VIKEF Project. *International Conference on Asian Digital Libraries* (2006), 101–110. (Citado en la página 9).
- [92] SUCHANEK, F., IFRIM, G., AND WEIKUM, G. LEILA: Learning to extract information by linguistic analysis. *Proceedings of the Second Workshop on Ontology Learning and Population: Bridging the Gap between Text and Knowledge* (2006), 18–25. (Citado en la página 9).
- [93] UNITED NATIONS DEVELOPMENT PROGRAM. United Nations Standard Products and Services Code. <https://www.unspsc.org>. (Citado en la página 10).
- [94] WEBER, N., AND BUITELAAR, P. Web-based ontology learning with ISOLDE. *Proceedings of the Workshop on Web Content Mining with Human Language at the International Semantic Web Conference 11* (2006). (Citado en la página 9).
- [95] WEISSTEIN, E. W. Grelling’s Paradox. <https://mathworld.wolfram.com/GrellingsParadox.html>. (Citado en la página 25).
- [96] YAMAGUCHI, T. Acquiring conceptual relationships from domain-specific texts. *Workshop on Ontology Learning* (2001). (Citado en la página 8).
- [97] ZHANG, D., WANG, B., WANG, N., YANG, Z., AND ZHAO, H. A new cognitive model for autonomous ontology learning. 2016 IEEE Eighth International Conference on Intelligent Systems (IS). (Citado en la página 8).