UNIVERSITA' DI BOLOGNA - CAMPUS DI CESENA

DIPARTIMENTO DI INGEGNERIA DELL'ENERGIA ELETTRICA E DELL'INFORMAZIONE "GUGLIELMO MARCONI"

CORSO DI LAUREA IN INGEGNERIA BIOMEDICA

Titolo dell'elaborato:

STATO DELL'ARTE SULLE PRINCIPALI ONTOLOGIE IN AMBITO MEDICO

Elaborato in:

CALCOLATORI ELETTRONICI

Relatore:
Paolo Castaldi

Candidata:
Elena Sarti

Correlatori:
Luca Roffia,
Adelina Vivaldo

Anno Accademico 2024/2025

Abstract

La crescente complessità e quantità dei dati biomedici a disposizione ha reso indispensabile l'adozione di strumenti in grado di favorire l'interoperabilità semantica tra sistemi informativi eterogenei. In questo contesto si inserisce il Web Semantico che, grazie all'impiego di tecnologie come **RDF** (Resource Description Framework) e **OWL** (Web Ontology Language), consente una rappresentazione strutturata e condivisa della conoscenza.

Questa tesi analizza il ruolo delle **ontologie** come elemento chiave per la modellazione semantica dell'informazione, con particolare attenzione alla loro implementazione mediante triple RDF e al loro impiego nel dominio biomedico. Dopo una trattazione teorica delle tecnologie semantiche e del concetto di ontologia, vengono esaminate in dettaglio alcune delle principali ontologie utilizzate in medicina e bioinformatica: **SNOMED CT, Gene Ontology, MeSH, UMLS e NCIT**. Per ciascuna di esse si analizzano la struttura, gli obiettivi e le modalità di integrazione nei sistemi informativi sanitari.

Attraverso questa analisi, si evidenzia come le ontologie costituiscano uno strumento fondamentale per migliorare la qualità, la coerenza e la riusabilità dei dati in ambito medico, contribuendo in modo significativo allo sviluppo di soluzioni intelligenti per la gestione della conoscenza biomedica.

Indice

Elenco delle figure

Elenco delle tabelle

Elenco dei Codici

Capitolo 1

Introduzione

1.1 Panoramica del contesto

Nel contesto dell'attuale evoluzione tecnologica, la gestione della conoscenza assume un ruolo sempre più centrale, in particolare nei settori caratterizzati da un'elevata complessità informativa, come quello biomedico. La crescente disponibilità di dati clinici, genetici e farmacologici richiede strumenti che non solo ne consentano l'organizzazione e l'accesso, ma anche l'integrazione semantica tra fonti eterogenee. In tale scenario si inserisce il Web Semantico [1], un'estensione del Web tradizionale che mira a rendere i contenuti accessibili non solo agli esseri umani, ma anche alle macchine, attraverso rappresentazioni strutturate della conoscenza.
Elemento fondante del Web Semantico sono le ontologie [9], strumenti formali che permettono di definire concetti, relazioni e vincoli di un dominio specifico, rendendo possibile una comprensione condivisa delle informazioni. L'implementazione delle ontologie si basa su tecnologie standard come RDF [15] (Resource Description Framework), che consentono di descrivere dati secondo una struttura semantica basata su triple soggetto-predicato-oggetto. Tali tecnologie sono fondamentali per costruire sistemi intelligenti capaci di interrogare, inferire e aggregare informazioni in maniera coerente.

1.2 Scopo della tesi

La presente tesi si propone di analizzare il ruolo delle ontologie nel contesto del Web Semantico, focalizzandosi sia sugli aspetti teorici che ne permettono l'implementazione che sulle applicazioni pratiche in ambito medico. L'obiettivo è duplice: da un lato, fornire una panoramica delle tecnologie semantiche alla base della modellazione della conoscenza; dall'altro, esaminare alcune

delle principali ontologie biomediche attualmente in uso, evidenziandone caratteristiche, ambiti di impiego e benefici in termini di interoperabilità, standardizzazione e supporto alla ricerca clinica.

1.3 Articolazione in capitoli

La presente tesi è articolata in due capitoli principali:

1. nel primo capitolo vengono introdotti i concetti fondanti del Web Semantico, con particolare attenzione al modello RDF e ai linguaggi per la definizione di ontologie, in particolare OWL [4]. Viene inoltre analizzato il concetto stesso di ontologia e il suo ruolo nella rappresentazione della conoscenza;

2. il secondo capitolo è dedicato a un approfondimento delle principali ontologie impiegate in ambito medico, quali SNOMED CT [25] (Systematized Nomenclature of Medicine – Clinical Terms), Gene Ontology [7], MeSH (Medical Subject Headings) [26], UMLS (Unified Medical Language System) [27] e NCIT (National Cancer Institute Thesaurus) [17]. Per ciascuna di esse si illustrano struttura, finalità e modalità di utilizzo all'interno di sistemi informativi sanitari, motori di ricerca scientifici e applicazioni di supporto decisionale.

1.4 Conclusione

Attraverso questo percorso, si intende mettere in luce come l'integrazione tra modelli ontologici e tecnologie semantiche possa offrire un contributo significativo alla gestione efficiente e intelligente dei dati biomedici, promuovendo un'evoluzione verso sistemi sanitari sempre più interoperabili, precisi e orientati alla conoscenza.

Capitolo 2

Concetti di base

In questo capitolo, verranno illustrati i riferimenti teorici che costituiscono le fondamenta per l'implementazione dell'oggetto della tesi. L'obiettivo è fornire al lettore i concetti di base necessari.

2.1 Semantic Web

"Most of the web's content today is designed for humans to read, not for computer programs to manipulate meaningfully."

Così Tim Berners-Lee, inventore del World Wide Web (WWW), si espresse a proposito della funzionalità della sua stessa invenzione [1]: il suo scopo iniziale era, infatti, la creazione di una piattaforma in cui documenti e dati potessero essere condivisi, e a cui si potesse accedere globalmente. Si tratta del **Web di Documenti**, implementato per uso umano.

2.1.1 Tecnologie chiave del Document Web

È possibile ricondurre le fondamenta del Web che conosciamo a tre tecnologie principali:

1. URL (Uniform Resource Locator): indirizzo univoco orientato alla posizione della risorsa che identifica, è un localizzatore. Ogni documento disponibile nel web può essere identificato tramite un URL;

2. HTTP (HyperText Transfer Protocol): protocollo alla base della comunicazione dei web server, definisce la struttura dei messaggi trasmessi attraverso il web per la ricerca di pagine HTML, come illustrato in figura 2.1;

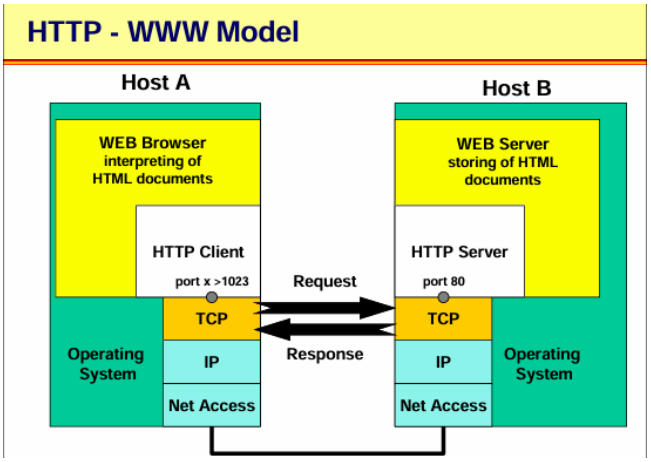

Figura 2.1: HTTP nel WWW.

Fonte: `https://www.ict.tuwien.ac.at/lva/384.081/infobase/L63-WWW_v4-3.pdf`

3. HTML (HyperText Markup Language): linguaggio che struttura il contenuto delle pagine web. Fornisce il framework per presentare testo, immagini, links sul web.

Se da un lato queste tecnologie creano un web che dispone di una grandissima quantità di informazioni, dall'altro strutturano i dati in formati che non ne permettono l'apprendimento ai computer senza intervento umano. Infatti, i contenuti sono pensati per essere letti e interpretati da essere umani; gli agenti automatici sono in grado di analizzare la struttura di una pagina, ma non comprendono il significato di ciò che elaborano, siano essi titoli, testo, links o immagini. Affinchè ciò sia possibile, l'informazione online deve essere corredata da metadati[1] comprensibili anche alle macchine, permettendo che dati esistenti su piattaforme diverse vengano automaticamente correlati ed elaborati dai calcolatori.

2.1.2 Il Web Semantico: un Web di dati

L'interoperabilità[2] è resa possibile dal Semantic Web, termine coniato da Tim Berners-Lee stesso [1]. Esso rappresenta un'evoluzione del World Wide Web: non si limita più ad associare un URL a un documento, bensì a ogni risorsa, che viene così identificata in modo univoco. Nel Semantic Web, i dati sono descritti tramite metadati che ne specificano il contesto semantico, in un formato adatto all'interpretazione e all'elaborazione automatica. L'informazione assume così un significato preciso, e le relazioni tra le risorse possono essere esplicitate e formalizzate. Questo approccio permette ai computer di comprendere, collegare e inferire nuova conoscenza a

[1] Treccani: *"insieme di dati accessori che contribuiscono a descrivere in modo dettagliato e completo un oggetto o un soggetto"*.

[2] L'interoperabilità è definita come la capacità di due o più sistemi diversi di comunicare tra loro in maniera efficiente e di potere interpretare la stessa informazione.

partire da dati disponibili. Tale approccio strutturato permette una comunicazione efficace tra sistemi che adottano una stessa strutturazione dei dati e vocabolari standardizzati, garantendo interoperabilità, quindi integrazione e analisi di dati provenienti da fonti diverse. In ambito medico, i dati spesso sono memorizzati in formati diversi e mancano degli standard universali: le tecnologie del Semantic Web sono perciò cruciali affinchè professionisti del campo della sanità abbiano accesso a dati più completi, aiutandoli a prendere migliori decisioni e migliorando la qualità del sistema sanitario.

2.1.2.1 Wikipedia Vs DBpedia

Un esempio del passaggio dal Web tradizionale al Semantic Web è la differenza tra Wikipedia e DBpedia, progetto nato nel 2007 con lo scopo di estrarre informazioni strutturate da Wikipedia e pubblicarle sul Web come Linked Open Data [2.2].
Su Wikipedia le informazioni relative alle entità (e.g. malattie, ad esempio la malattia di Alzhaimer) sono presentate sotto forma di pagine HTML pensate per essere lette da esseri umani. Se un utente cerca la malattia di Alzhaimer, si troverà una pagina con una spiegazione dettagliata - informazioni testuali, non strutturate - che contiene dei links, collegamenti ipertestuali ad argomenti correlati, per esempio alla demenza. Questi links, utili agli utenti, non sono altro che collegamenti tra documenti HTML. Se un programma dovesse estrarre informazioni sulla malattia, dovrebbe processare tutte le pagine HTML, ricorrendo a complesse tecniche di interpretazione del testo per trovare dettagli rilevanti.
Al contrario, DBpedia estrae informazioni strutturate da Wikipedia e le trasforma in triple RDF[3], permettendo l'instaurazione di relazioni tra entità, come mostrato in figura 2.2.

Listing 2.1: Esempio di triple RDF per la Malattia di Alzheimer

```
@prefix dbo: <http://dbpedia.org/ontology/> .
@prefix dbpedia: <http://dbpedia.org/resource/> .

dbpedia:Alzheimer's_disease a dbo:Disease ;
  dbo:symptom dbpedia:Memory_loss ;
  dbo:complication dbpedia:Dementia ;
  dbo:field dbpedia:Neurology .
```

[3]La conoscenza all'interno del Semantic Web è modellata tramite triple RDF; il modello viene approfondito nel paragrafo 2.2

Se un utente cerca la malattia di Alzhaimer su DBpedia, gli è restituita non una pagina HTML, ma una risorsa che descrive la malattia in termini *machine-readable*. Mentre su Wikipedia è presente un link alla pagina dedicata alla demenza, una delle complicazioni a cui può portare la malattia, nella risorsa RDF è presente il predicato *dbo:complication*, che collega l'Alzhaimer alla risorsa della demenza, permettendo al computer di comprendere direttamente la connessione. Cliccando su *dbpedia:Dementia*, utenti e macchine sono reindirizzati alla risorsa della demenza: questo link rappresenta una relazione semantica tra due entità in maniera interpretabile dai computer.

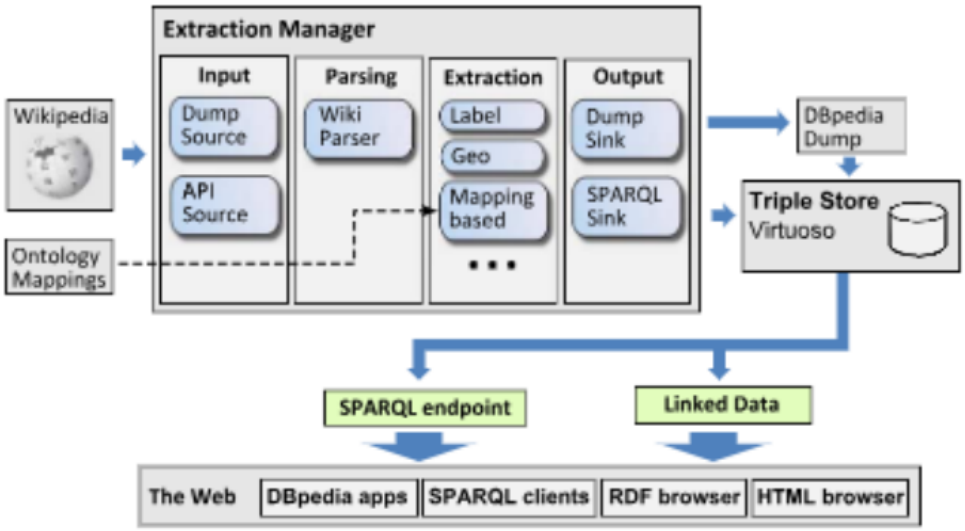

Figura 2.2: Processo di estrazione dei dati da Wikipedia in DBpedia. Il diagramma illustra le fasi di input, parsing, estrazione ed esportazione verso un triple store RDF.
Fonte: `https://journals.sagepub.com/doi/abs/10.3233/SW-140134`

2.1.3 Semantic Web Stack

W3C[4] ha definito un modello dell'architettura del Web Semantico, il cosiddetto *Semantic Web Stack*, illustrato in figura [2.3]. Esso rappresenta i vari strati necessari per costruire il Web Semantico; dimostra come diverse tecnologie, molte delle quali prese da standard già esistenti e riadattate, interagiscano tra loro per formare un web machine-readable.

1. URI/IRI: gli URI (Uniform Resource Identifier) sono indirizzi univoci per identificare una risorsa web, che comprendono URL e URN (Uniform Resource Name). Un URI è sia human-readable che machine-readable; un IRI(International Resource Identifier) estende

[4]World Wide Web Consortium: organizzazione internazionale che sviluppa standard e linee guida per il web, con il fine di garantire accessibilià, privacy, sicurezza e globalizzazione.

la validità di un URI, permettendo anche l'utilizzo di caratteri non in codifica ASCII - nel Semantic Web, spesso i dati hanno origine da ambienti culturali e linguistici diversi, perciò questa estensione è fondamentale.

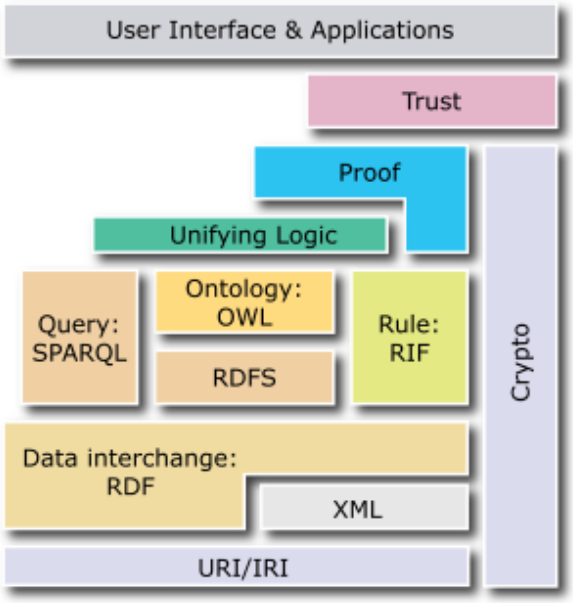

Figura 2.3: Il Semantic Web Stack

Fonte: `https://www.researchgate.net/figure/Semantic-Web-technology-stack_fig1_309201011/actions#reference`

2. XML (eXstensible Markup Language): linguaggio che utilizza tags per definire e organizzare le informazioni, permettendo un rappresentazione ordinata di complessi dataset. Un esempio di semplice codice XML per descrivere il diabete mellito potrebbe essere:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<Malattia>
  <Nome>Diabete di tipo 2</Nome>
  <Descrizione>
    Malattia cronica caratterizzata da alti livelli di zucchero
      ↪ nel sangue.
  </Descrizione>
  <Sintomo>Sete eccessiva</Sintomo>
  <Causa>Vita sedentaria</Causa>
</Malattia>
```

3. RDF (Resource Description Framework): modello che rappresenta l'informazione tramite triple, instaurando relazioni tra le entità. Ogni tripla è composta da tre elementi: soggetto, predicato, oggetto; permette di rappresentare le informazioni sulle risorse nella forma di grafo, un *Knowledge Graph*.

4. RDFS (RDF Schema): espansione di RDF che fornisce una semantica addizionale, permettendo la formazione di gerarchie e relazioni tra le risorse.

5. OWL - Web Ontology Language: linguaggio che aggiunge a RDFS costrutti più avanzati per descrivere la semantica delle dichiarazioni RDF. Consente di indicare vincoli come la cardinalità o la transitività; basandosi su una logica descrittiva, conferisce al Semantic Web il potere di ragionamento.

6. SPARQL: linguaggio che permette di effettuare query RDF, recuperare informazioni da dataset RDF, e quindi facilitare l'estrazione di conoscenza dai dati tramite un set di regole; un esempio di query SPARQL è illustrato al paragrafo 2.4.5.

7. Logica: inferenza e deduzione. Questo strato del Semantic Web riproduce i processi cognitivi umani, permettendo ai sistemi di dedurre nuove informazioni dai dati esistenti tramite regole strutturate [5]. Ciò porta alla scoperta di nuove relazioni e pattern, quindi maggiore automazione nei processi decisionali.

8. Conferma: meccanismi per convalidare la correttezza delle deduzioni, assicurando che le conclusioni tratte dalla macchina siano rintracciabili e verificabili. Vengono documentati tutti gli step logici effettuati per arrivare a una determinata conclusione.

9. Crittografia e Firma Digitale: meccanismi per proteggere i dati, garantendo che rimangano autentici e accessibili solo agli utenti autorizzati. Firmando dati RDF, viene autenticata la fonte dei dati, assicurando che questi non siano stati alterati durante la trasmissione.

2.2 RDF Data Model

W3C dà una definizione di RDF [15]: si tratta di un framework per esprimere informazioni riguardo a *risorse*[6], affinchè possano essere scambiate tra applicazioni diverse senza perdita di

[5] Per esempio, se A $>$ B e B $>$ C, allora A $>$ C.

[6] Qualsiasi cosa può essere una risorsa: documenti, persone, concetti, oggetti fisici.

significato. In particolare, RDF è usato per pubblicare e collegare dati sul web.
Per esempio, cercando `http://www.example.org/bob#me` , l'utente ha come risposta dati riguardo a Bob, magari comprendenti il fatto che soffre di diabete mellito[7]. Recuperando l'IRI della patologia, all'utente sono forniti più dati riguardo ad essa, inclusi i link ad altri dataset, che ne identificano per esempio le cause e le conseguenze. Tali usi del modello RDF sono qualificati come *Linked Data*.

2.2.1 Triple RDF

RDF permette di effettuare dichiarazioni che esprimono relazioni tra due risorse: un **soggetto** e un **oggetto** [8]. Il formato di una tripla[9] ha la seguente struttura:
<subject> <predicate> <object>, dove soggetto e oggetto sono risorse, mentre il predicato è descritto da una proprietà.
Esempio: $< http://example.org/Bob > < http://example.org/hasDesease > < http://example.org/Diabetes >$[10] .

2.2.2 RDFS

Da solo, il modello RDF non è in grado di effettuare assunzioni riguardo a cosa gli IRI delle risorse significhino; per questo, è spesso usato in combinazione con dei vocabolari che forniscono informazione semantica riguardo alle risorse. Per definire le caratteristiche semantiche dei dati, RDF fornisce un linguaggio, RDF Schema, che utilizza dei costrutti principali per modellare l'informazione[11]:

- *rdfs:class*: è possibile definire delle classi a cui appartengono le risorse;
- *rdf:type*: proprietà che dichiara la relazione tra un'istanza e la sua classe; [12]
- *rdfs:domain, rdfs:range*: specificano a che classe devono appartenere, rispettivamente, il soggetto e l'oggetto di un predicato.

[7] Come identificato dall'IRI che identifica la patologia.
[8] È il predicato che stabilisce la natura di una relazione.
[9] Dato che consistono di tre elementi, le dichiarazioni sono chiamate triple.
[10] Bob è il soggetto, hasDesease è il predicato, Diabetes è l'oggetto.
[11] *rdf* è il prefisso usato per definire elementi RDF base, come *rdfs* è il prefisso per definire elementi RDFS. Il prefisso usato per le risorse di esempio è *ex*.
[12] È possibile creare gerarchie tra classi e sottoclassi e tra proprietà e sottoproprietà: *rdfs:subClassOf, rdfs:subPropertyOf*.

2.2.3 Knowledge Graphs

La stessa risorsa è spesso referenziata in molte triple: nasce così la possibilità di rappresentare l'informazione tramite dei grafi. Si consideri l'esempio:

<Bob> <is a> <person>.

<Bob> <hasAge> <22>.

<Bob> <hasDeasease> <Diabetes>.

<Diabetes> <is a> <ChronicDisease>.

<Hypertension> <is a> <Chronic Deasease>.

Possiamo rappresentare le suddette relazioni tramite un grafo, come quello in figura 2.4; ogni risorsa è rappresentata come un **nodo** e i predicati come **archi**. Ogni tripla forma un segmento del grafo. Questa rappresentazione rende più facile la comprensione di complesse reti di dati, evidenziando le relazioni tra le risorse. Grazie a RFDS, è possibile fare inferenze: per esempio, data la tripla ex:Bob ex:hasDesease ex:Diabetes è possibile derivare la seguente tripla: ex:Bob rdf:type foaf:Person.[13]

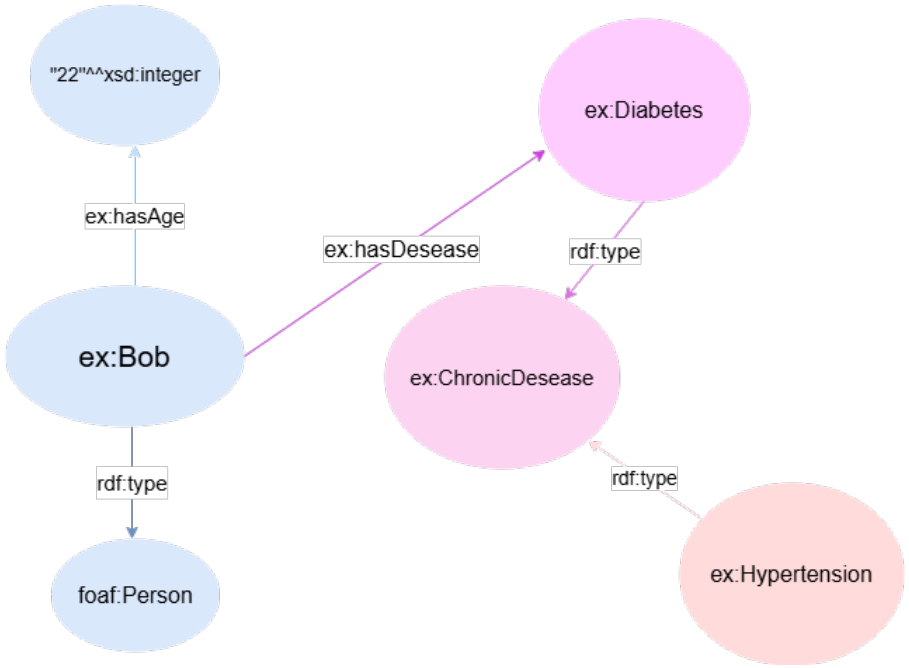

Figura 2.4: Grafo delle triple in esempio.

2.2.4 Sintassi Turtle per RDF

Per rappresentare grafi RDF, è possibile usare sintassi diverse. Alcuni esempi sono:

1. RDF/XML: usa la sinstassi XML per descrivere triple RDF;

[13]Infatti, ex:hasDesease rdfs:domain foaf:Person.

2. JSON-LD: rappresenta i dati tramite JSON, rendendo più facile l'integrazione con applicazioni web;

3. Turtle family of RDF languages: comprende N-Triple[14] e Turtle.

Turtle (Terse RDF Triple Language) è una sintassi più human-readable, che si basa sull'uso di prefissi e semplici dichiarazioni. Se più triple hanno lo stesso soggetto, Turtle lo omette, evitando ridondanza e mantenendo il codice pulito. In sintassi Turtle, l'esempio del grafo 2.4 risulta il seguente:

Listing 2.2: Sintassi Turtle per RDF

```
@prefix ns1: <http://example.org/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

ns1:Bob a ns1:Person ;
    ns1:hasAge 22 ;
    ns1:hasDisease ns1:Diabetes .

ns1:Hypertension a ns1:ChronicDisease .

ns1:Diabetes a ns1:ChronicDisease
```

2.3 Ontologie

2.3.1 Definizione

Nel Semantic Web, oltre alle triple RDF, sono le ontologie ad assumere un ruolo cruciale. Il termine viene dalla filosofia, dove indica lo studio dell'essere in quanto tale, e delle sue categorie fondamentali; è poi entrato in uso nel campo dell'intelligenza artificiale e della rappresentazione della conoscenza.
La definizione di ontologia risale al 1992 [9]: un'ontologia è una *"specificazione formale ed esplicita di una concettualizzazione condivisa"*.

1. Concettualizzazione: visione astratta e semplificata di ciò che si vuole rappresentare.[15]

[14] Ottimo per grandi dataset, fornisce una rappresentazione riga per riga delle triple.
[15] Oggetti, concetti ed entità di qualsiasi tipo.

2. Esplicita: i concetti utilizzati e i vincoli al loro utilizzo sono definiti in modo chiaro e comprensibile [23].

3. Formale: deve essere comprensibile dalle macchine e, quindi, deve essere espressa in modo formale.

4. Condivisa: la conoscenza che si vuole rappresentare deve essere accettata consensualmente dalle diverse comunità.

Questa definizione si riferisce, quindi, a un tentativo di formulare una concettualizzazione esaustiva e rigorosa nell'ambito di un determinato dominio.
Si tratta generalmente di una struttura dati gerarchica che contiene tutte le entità rilevanti, e le relazioni esistenti tra esse, le regole, gli assiomi e i vincoli specifici del dominio. L'obiettivo principale è la rappresentazione dell'informazione in modo strutturato, permettendo interoperabilità tra sistemi diversi, e la nascita di reti di conoscenze che replichino la complessità del mondo reale. Ogni informazione del dominio di pertinenza sarà mappata, attraverso la definizione dei metadati, dalla propria ontologia ed inserita in un contesto che la relazioni ad altre ontologie.
Un'ontologia utilizza strutture che permettono ai calcolatori di interpretare e processare dati complessi: questi non vengono solo memorizzati, ma anche capiti e usati in modo efficiente dalle macchine.
Per avere applicazioni che possano interagire automaticamente, è necessario avere un modo comune di interpretare le collezioni di informazioni, per ridurre o eliminare la confusione concettuale e terminologica presente in uno specifico settore. L'ontologia elimina l'ambiguità, fornendo una base semantica e un vocabolario concettuale condiviso.

2.3.2 Classificazione

Si possono distinguere diverse tipologie di ontologie[16]. Una prima distinzione si basa sul livello di generalità usato per la descrizione dei domini:

- **ontolgie generiche o top-level**: descrivono concetti molto generali[17] che sono indipendenti da un particolare dominio;

[16]Tuttavia, la distinzione tra le varie categorie non è mai netta, in quanto una stessa ontologia potrebbe appartenere a più categorie.
[17]Quali spazio, tempo, materia, eventi.

- **ontologie di dominio**: descrivono un determinato dominio (e.g.: la medicina) o un generico task (e.g.: imaging cardiovascolare) specializzando i concetti e le relazioni introdotti nell'ontologia top-level;

- **ontologie applicative**: contengono tutte le definizioni necessarie a modellare la conoscenza richiesta per una specifica applicazione (e.g.: fMRI cardiaca). A tale scopo, possono combinare concettualizzazioni proprie sia delle ontologie top-level che di quelle di dominio; l'ontologia risultante è particolare per una determinata applicazione e non sempre può essere riutilizzata per un compito differente [8].

Le ontologie possono differire anche per il formalismo usato per esprimere i termini e i loro significati: un livello di formalismo maggiore non implica un'ontologia maggiormente sviluppata, bensì una con un livello di specificazione maggiore. Si distinguono ontologie altamente informali, ontologie semi-informali, ontologie semi-formali e ontologie rigorosamente formali. Il livello di formalismo impiegato dipende dal campo di utilizzo: se l'ontologia è pensata per la comunicazione tra persone, la rappresentazione può essere molto più informale; se è pensata per sistemi automatici, è necessario un rigoroso formalismo, affinchè il linguaggio possa essere comprensibile alle macchine.
Un'ulteriore classificazione può essere condotta sulla base dell'espressività, cioè sulla quantità di vincoli sulle proprietà: maggiore la quantità assiomatica interna, maggiore l'espressività.

2.3.3 Elementi costitutivi

Un'ontologia è costuituita da:

1. **classi** (o concetti): qualsiasi tipo di entità del dominio, siano esse astratte o concrete. Esempi di concetti, nel settore medico, sono l'apparato cardiovascolare, il processo di vascolarizzazione, il flusso informativo ospedaliero [16].
   Le classi possono essere organizzate in gerarchie di superclassi e sottoclassi: per esempio, dalla classe *patologie* è possibile derivare le sottoclassi *Croniche* e *NonCroniche*;

2. **istanze**: attuali oggetti presenti nel dominio, ereditano attributi e relazioni dalle classi. Ad esempio, definita la classe *Patologie*, una istanza possibile sarà *Diabete*.

3. **relazioni**: interazioni tra classi del dominio, formalmente rappresentate come un qualsiasi sottoinsieme del prodotto di N insiemi[18] [16]. Dati due insiemi A e B, con $A = \{a_1, a_2\}$ e $B = \{b_1, b_2\}$, il prodotto cartesiano $A \times B$ è:

$$A \times B = \{(a_1, b_1), (a_1, b_2), (a_2, b_1), (a_2, b_2)\}.$$

Una relazione tra A e B è un sottoinsieme di questo prodotto. Ad esempio, $R = \{(a_1, b_2), (a_2, b_1)\}$ è una relazione binaria tra elementi di A e B.

Se considero la classe *Paziente* = {Paz1, Paz2} e la classe *Patologia* = {Diabete, Neoplasia}, la relazione *affetto_ da* potrebbe essere:

$$affetto\_\,da = \{(\text{Paz1}, \text{Diabete}), (\text{Paz2}, \text{Neoplasia})\};$$

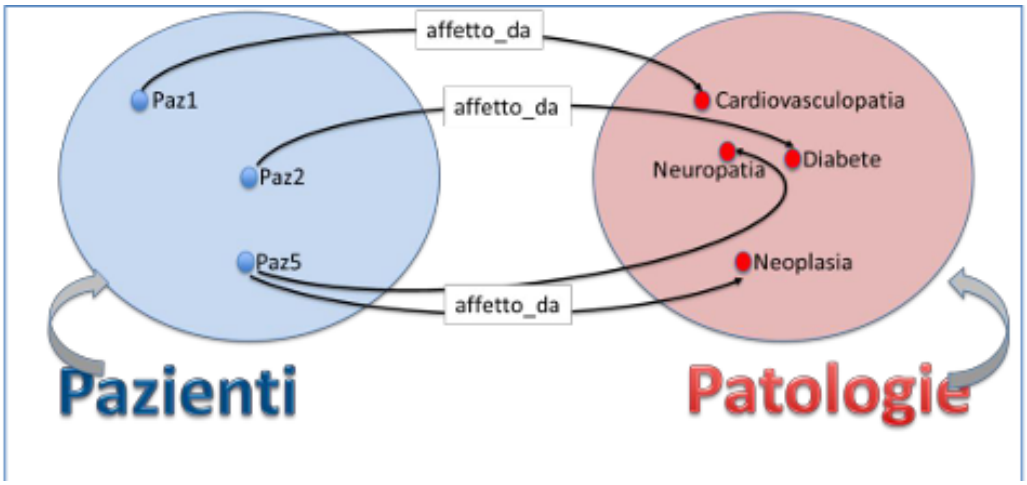

Figura 2.5: La figura mostra un esempio dove sette istanze, ovvero *Paz1, Paz2, Paz5, Cardiovasculopatia, Diabete, Neoplasia e Neuropatia* sono raggruppate in due classi, *Pazienti* e *Patologie*, e relazionati attraverso la proprietà *affetto_ da*.

Fonte: `https://core.ac.uk/download/pdf/37830677.pdf`

4. **attributi**: proprietà che descrivono le caratteristiche delle classi;

5. **assiomi**: regole logiche e vincoli che rendono coerente l'ontologia; affermazioni sempre vere sul dominio definito dell'ontologia[19], necessarie per fare inferenza.

[18] Dove N è il numero delle classi del dominio.

[19] Esempio di assioma: AIDS conclamato è HIV positivo.

Il tipo e la specificità delle inferenze che si possono desumere dalle ontologie sono strettamente legati alla qualità descrittiva dell'ontologia, che dipende soprattutto da proprietà e assiomi. Inoltre, per essere considerata tale, un'ontologia deve disporre di:

i. un vocabolario finito e controllato;

ii. rigide relazioni gerarchiche di sottoclassi tra le classi;

iii. interpretazione non ambigua delle classi e delle relazioni tra le varie entità.

Una ontologia popolata con istanze e regole di inferenza pone i fondamenti di una vera e propria base di conoscenza[20], strumento dinamico e riusabile.

2.3.4 Costruire un'ontologia

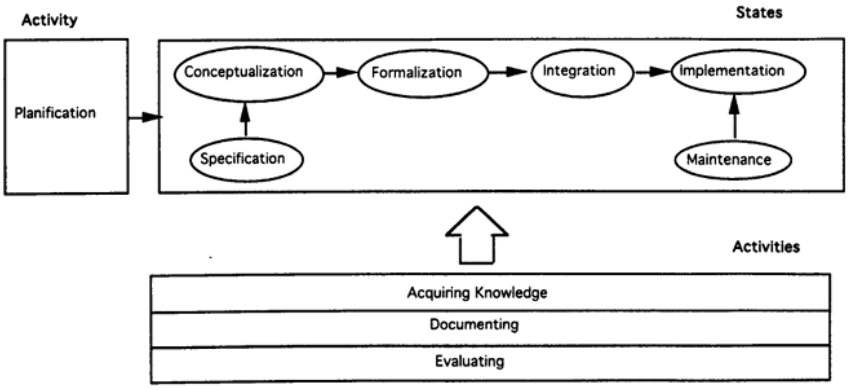

Figura 2.6: Fasi della costituzione e del ciclo di vita di un'ontologia.
Fonte: `http://oa.upm.es/5484/1/METHONTOLOGY_.pdf`

Lo sviluppo di un'ontologia è un processo dinamico che parte da un'ontologia iniziale approssimata, successivamente rivista, raffinata e definita nei dettagli [19]. Non esiste dunque un'unica metodologia di creazione corretta; la soluzione migliore dipende sempre dall'uso che si deve fare dell'ontologia.

Prima di analizzare i passi dello sviluppo di un'ontologia (illustrazione in figura 2.6), è bene approfonfire la differenza tra due concetti che verranno enunciati in seguito:

2.3.4.1 Flat Glossary Vs Structured Glossary

Un *flat glossary* è una semplice lista di concetti associati a una definizione, in cui ogni termine è indipendente dagli altri. Fornisce definizioni condivise, non è gerarchico e non struttura la

[20]Spesso, questo termine viene confuso con il termine database, ma il database si concentra sul dato ed è specifico di una realtà, non ha caratteristiche di riusabilità.

conoscenza; non vengono esplicitate le relazioni tra i concetti. Un esempio è riportato in tabella 2.1.

Termine	**Definizione**
Diabete	Malattia metabolica cronica
Paziente	Persona che riceve assistenza medica
Terapia	Trattamento per una patologia

Tabella 2.1: Esempio di flat glossary.

Uno *structured glossary* è un glossario in cui i termini sono organizzati secondo relazioni semantiche: facilita l'inferenza e il ragionamento autonomo. Non ci sono solo termini e definizioni, ma anche classi, individui, proprietà, domini e range. Struttura la conoscenza per renderla leggibile non solo da umani, ma anche da agenti automatici: si avvicina alla forma di una ontologia. Un esempio è riportato in tabella 2.2.

ex:Diabetes	rdf:type	ex:Desease
ex:Patient	rdf:type	foaf:Person
ex:hasDeasease	rdfs:domain	foaf:Person
ex:hasDeasease	rdfs:range	ex:Desease

Tabella 2.2: Esempio di structured glossary.

2.3.4.2 Fasi dello sviluppo

1. **Esame del dominio ed acqusizione della conoscenza**: fase iniziale in cui si determinano il dominio di interesse[21], lo scopo dell'ontologia, i tipi di domande a cui l'informazione può fornire risposte[22], chi fruirà dell'ontologia. Si raccolgono poi quante più informazioni possibili sul dominio di interesse e si analizzano i termini usati in maniera consistente per descrivere le entità.

2. **Considerare il riuso di risorse esistenti**: rifinire ed estendere risorse già esistenti, quali glossari, dizionari di termini e sinonimi, tassonomie ed altre ontologie è un vantaggio,

[21]Per esempio, in ambito sanitario il dominio clinico e quello dei servizi sanitari sono concettualmente molto diversi.

[22]Si stila una lista di *competency questions*, che serviranno come test ultimo per verificare che l'ontologia contenga le informazioni necessarie, e quanto queste vadano nel dettaglio.

non solo in termini di tempo, ma anche perchè permette l'interazione del sistema con altre applicazioni che hanno già adottato una particolare ontologia. Sul web e nella letteratura sono presenti librerie di ontologie riutilizzabili, come Ontolingua ontology library[23], oppure DAML ontology library[24].

3. **Pianificare lo svluppo dell'ontologia**: si progetta la struttura concettuale complessiva del dominio, identificando i principali concetti, le loro proprietà e le relazioni tra essi. Si sviluppano progressivamente un *flat glossary* e uno *structured glossary* [2.3.4.1] e, infine, si identificano tutte le relazioni concettuali tra gli oggetti.

4. **Definire le classi e l'ordine gerarchico** - ci sono tre possibili approcci per lo sviluppo delle gerarchie [19]:

   (a) sviluppo **top-down**: inizialmente sono definite le classi più generali, e successivamente le sottoclassi e i concetti più specifici;

   (b) sviluppo **bottom-up**: inizia con la definizione delle classi più specifiche, le foglie della gerarchia[25], con un successivo raggruppamento di queste classi in concetti più generali;

   (c) approccio **middle-out** o combinato: sviluppo che combina il top-down e il bottom-up. Sono definiti inizialmente i concetti più salienti, e successivamente vengono generalizzati in superclassi e specializzati in sottoclassi. L'approccio combinato risulta spesso quello preferito, in quanto i concetti salienti sono tendenzialmente quelli che descrivono meglio il dominio. In figura 2.7 un esempio dei diversi livelli della tassonomia delle patologie.

   Le classi vengono organizzate in una tassonomia gerarchica chiedendosi se un oggetto che è istanza di una determinata classe sarà necessariamente anche istanza di qualche altra classe:
   Se una classe A è una superclasse della classe B, allora ogni istanza di B è anche una istanza di A.

[23] `http://www.ksl.stanford.edu/software/ontolingua/`

[24] `http://www.daml.org/ontologies/`

[25] In informatica, un albero è una struttura dati gerarchica, composta da nodi collegati tra loro da archi. C'è sempre un nodo da cui si diramano tutti gli archi, detto radice; i nodi da cui non partono archi sono detti foglie dell'albero.

Figura 2.7: *Patologia* è la classe più generale, *Cardiovasculopatia* è una classe middle-level, da cui si può partire nel metodo combinato, e le foglie sono bottom-level.

Fonte: `https://core.ac.uk/download/pdf/37830677.pdf`

5. **Dichiarare le istanze e definire le propreietà**: si assegnano alle proprietà cardinalità[26], tipo[27], dominio e range. In figura 2.8, una rappresentazione delle possibili proprietà di una classe.

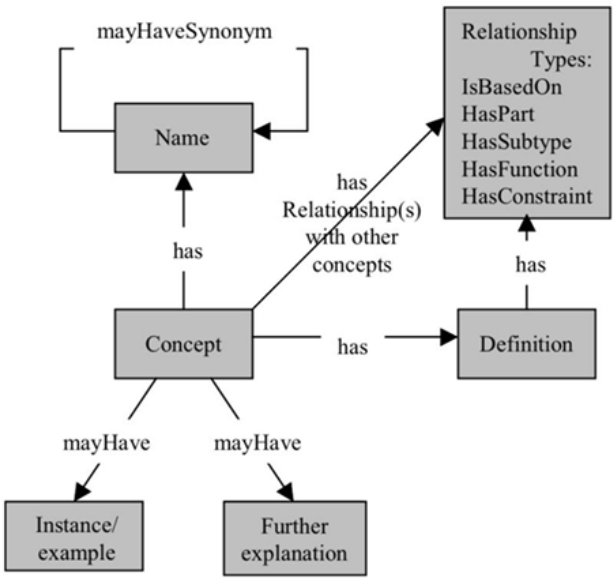

Figura 2.8: Rappresentazione delle possibili proprietà di una classe.

Fonte: `https://core.ac.uk/download/pdf/11310019.pdf`

[26]Numero di valori che una proprietà può assumere.

[27]Ad esempio: intero, stringa, booleano.

6. **Verificare l'ontologia**: una volta sviluppata, occorre analizzare l'ontologia per individuare eventuali inconsistenze sintattiche, logiche e semantiche tra i suoi elementi. Vanno previste diverse tipologie di verifica: verifica di completezza rispetto al dominio da modellare, verifica di concisione per evitare di fare assunzioni implicite e ridondanti, verifica di consistenza per evitare definizioni in contraddizione, verifica di coerenza affinché tutte le relazioni siano consistenti con le definizioni dei concetti.

7. **Rilasciare l'ontologia**: al termine dello sviluppo dell'ontologia, è fondamentale corredarla di annotazioni attraverso la scrittura di etichette[28] da abbinare ai concetti: sia il tag di default, quello preferito per il concetto, sia uno o più tag alternativi utilizzati per sinonimi o varianti, con il fine di consentire l'individuazione di tutti i dati semanticamente relazionati. Se l'ontologia deve essere multilingua, ogni etichetta dovrà avere specificata una lingua (e.g.: @it).

Listing 2.3: *rdfs:label* è l'etichetta di default, per dare un nome leggibile al concetto o alla proprietà.

```
:Diabete a :Patologia ;
    rdfs:label "Diabete"@it ;
    rdfs:label "Diabetes"@en .
```

Listing 2.4: *skos:altlabel* è un'etichetta alterntiva che serve per aggiungere sinonimi, abbreviazioni o varianti linguistiche; utile per migliorare la ricerca semantica e il matching tra i dati.

```
:Diabete skos:altLabel "Malattia diabetica"@it ;
         skos:altLabel "Diabete mellito"@it ;
         skos:altLabel "Sugar disease"@en .
```

Le ontologie sono sul Web. Le applicazioni possono utilizzare ontologie differenti, oppure le stesse ma espresse in lingue diverse: le equivalenze tra termini e le relazioni intercorrenti tra loro possono diventare un problema non banale. Perciò, un passo fondamentale nella costruzione di un'ontologia è la scelta del linguaggio più opportuno da usare per definirne in maniera esplicita i componenti, in modo che siano condivisibili sul web. Di seguito, viene illustrata l'implementazione degli elementi costitutivi tramite OWL.

[28] Annotazioni associate ai concetti, usate per fornire nomi umani leggibili.

2.4 OWL: Web Ontology Language

OWL (Ontology Web Language) è un liguaggio realizzato appositamente per lo sviluppo delle ontologie e per la loro diffusione sul World Wide Web [23], implementato dal *W3C Web Ontology Working Group*[29]. Attualmente, il linguaggio è distribuito in versione 2, (informalmente, *Owl2*); per definirlo, sono stati utilizzati RDF e RDFS, al fine di permettere la descrizione di sofisticate basi di conoscenza, interpretabili sia da uomini che da agenti automatici. OWL mette a disposizione classi e sottoclassi, proprietà e sottoproprietà, vincoli sulle proprietà e regole di inferenza, che lo rendono adatto a modellare domini complessi. Rispetto a RDFS, OWL migliora significativamente il supporto al ragionamento autonomo, grazie ad avanzate regole logiche: diventa così uno strumento essenziale per applicazioni che richiedono una sofisticata rappresentazione della conoscenza, tra cui l'intelligenza artificiale.

2.4.1 OWL Lite, DL, Full

Esistono tre varianti del linguaggio OWL [4]: OWL Lite, OWL DL, OWL Full.

1. OWL Lite: versione semplificata di OWL, implementata per ontologie meno complesse, per cui sono sufficienti skills di ragionamento di base. Caratterizzata da una minore espressività, non è adottata su larga scala, dal momento che sono utilizzabili meno tipi di costrutti; resta comunque un linguaggio computazionalmente efficiente.

2. OWL Full: fornisce il massimo livello di espressività e di libertà - classi, istanze e proprietà possono essere combinate liberamente, senza restizioni. Questo sottolinguaggio rende quindi possibile la generazione di modelli di conoscenza molto complessi, ma contemporaneamente introduce potenziale ambiguità: una stessa entità può essere, allo stesso tempo, sia una classe che un'istanza[30]. Possono nascere perciò scenari in cui il processo di ragionamento non riesce a terminare, generando problemi decisionali.

3. OWL DL (Description Logic): in questo sottolinguaggio è raggiunto un equilibrio tra espressività e capacità decisionale; OWL DL permette l'espressione di relazioni complesse, controllate da restrizioni che garantiscono che il processo di ragionamento giunga sempre a una conclusione.

OWL DL è la variante più adatta per modellare complesse basi di conoscenza, in cui è necessario un rigoroso controllo sulla coerenza; si basa sulla logica descrittiva [9], con architettura illustrata

[29] `http://www.w3.org/2001/sw/WebOnt/`

[30] Per esempio, l'entità ex:Patient può rappresentare simultaneamente una classe ed un'istanza.

in figura 2.9. Un sistema basato sulla logica descrittiva è costituito da molteplici componenti: la base di conoscenza, costituita da TBox e ABox, e il meccanismo di inferenza che le lega.

1. TBox (Terminological Box): si tratta della componente terminologica che descrive un dominio, definendone classi e proprietà, come un vocabolario.

2. ABox (Assertional Box): asserzioni, fatti specifici del dominio in termini di istanze, ruoli e proprietà.

3. Linguaggio Descrittivo: strumento formale per modellare le ontologie, che definisce le relazioni tra le entità.

4. Ragionamento: permette al sistema di inferire nuove relazioni e fatti dalle definizioni esistenti nella TBox e dalle asserzioni della ABox, anche se non espressi esplicitamente, grazie alla logica che deriva dall'ontologia.

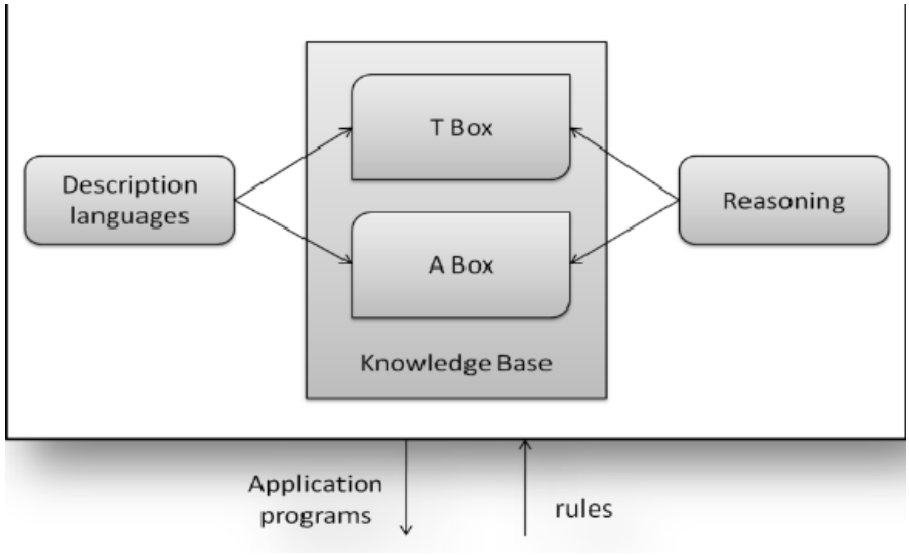

Figura 2.9: Architettura di un sistema basato sulla logica descrittiva.

Fonte: `https://www.researchgate.net/figure/The-Architecture-of-a-knowledge\protect\penalty\z@-representation-system-As-seen-in-Figure-3-the-author_fig5_234689081`

2.4.2 Documenti OWL

I documenti OWL sono chiamati *ontologie OWL* e sono di tipo RDF: l'elemento radice è un elemento *rdf:RDF*. L'header è costuito da una collezione di asserzioni, raggruppate sotto elementi *owl:Ontology*, contenenti commenti, la versione di controllo e le inclusioni di altre ontologie

tramite l'elemento *owl:imports*[31]

Sono presenti elementi che permettono di effettuare una gestione delle diverse versioni delle ontologie e la definizione di specificatori di compatibilità e incompatibilità fra le version[23]:

- *owl:priorVersion*: indica la versione corrente dell'ontologia;
- *owl:versionInfo*: contiene informazioni circa l'attuale versione in formato stringa;
- *owl:backwardCompatibleWith*: elemento che informa sulle versioni precedenti dell'ontologia con cui questa resta compatibile;
- *owl:incompatibleWith*: elemento che informa sulle versioni precedenti dell'ontologia con cui questa non è più compatibile.

2.4.3 Elemento classe

La definizione delle classi avviene tramite l'elemento *owl:Class*. Per esempio, è possibile definire la classe *Desease* nel seguente modo:

Listing 2.5: Definizione della classe OWL *Desease*, sottoclasse di *MedicalEntity*

```
@prefix : <http://www.example.org/ontology#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

:Disease rdf:type owl:Class ;
    rdfs:label "Disease"@en ;
    rdfs:subClassOf :MedicalEntity .
```

In OWL sono state prefinite due classi basi: *owl:Thing*, classe più generale che contiene tutte le altre, e *owl:Nothing*, classe vuota. Ogni oggetto di OWL appartiene a alla classe *owl:Thing*: nell'esempio precedente sarebbe quindi possibile sostituire *:MedicalEntity* con *:Thing*.

L'assioma ***owl:equivalentClass*** esprime equivalenza tra classi[32]; relazioni che esprimono combinazioni booleane di classi[33] sono invece ***owl:unionOf, owl:complementOf, owl:disjointWith, owl:intersectionOf***. Risulta così possibile continuare l'esempio precedente:

[31] Per *owl:imports* vale la proprietà transitiva: se l'ontologia A importa l'ontologia B e questa a sua volta importa l'ontologia C, allora A importa C.

[32] Classi equivalenti sono costitutite dagli stessi elementi.

[33] Unione, intersezione, disgiunzione, complemento.

Listing 2.6: Esempio di relazioni OWL tra classi.

```
:Patologia a owl:Class ;
    owl:equivalentClass :Disease ;

:Therapy a owl:Class ;
:Disease owl:disjointWith :Therapy .

:MentalDisease a owl:Class .
:PhysicalDisease a owl:Class .
:CombinedDisease a owl:Class ;
    owl:unionOf ( :MentalDisease :PhysicalDisease ) ;
    rdfs:label "Malattia Mentale o Fisica"@it .

:HereditaryDisease a owl:Class .
:ChronicCondition a owl:Class .
:InheritedChronicDisease a owl:Class ;
    owl:intersectionOf ( :HereditaryDisease :ChronicCondition ) ;
    rdfs:label "Malattia Ereditaria e Cronica"@it .

:NonDisease a owl:Class ;
    owl:complementOf :Disease ;
    rdfs:label "Non-Patologia"@it .
```

2.4.4 Proprietà

Le proprietà vengono distinte [4] in:

- ***owl:datatypeProperty***: assegna relazioni tra elementi classe e tipi di dati;
- ***owl:objectProperty***: assegna relazioni tra elementi di classi distinte.

Di seguito, è illustrata una tabella contenente le principali relazioni tra oggetti utilizzate:

Relazione	Significato	Esempio
owl:inverseOf	Definisce l'inverso di una proprietà	$hasSymptom \leftrightarrow isSymptomOf$
owl:TransitiveProperty	Se A è collegato a B, e B a C, allora A è collegato a C	isAncestorOf
owl:SymmetricProperty	Se A è collegato a B, allora B è collegato ad A	isSiblingOf
owl:AsymmetricProperty	Se A è collegato a B, B non può essere collegato ad A	isParentOf
owl:ReflexiveProperty	Ogni istanza è collegata a sé stessa con questa proprietà	isEqualTo
owl:IrreflexiveProperty	Nessun istanza può essere collegata a sé stessa	isTallerThan
owl:FunctionalProperty	Ogni soggetto può avere al massimo un oggetto per quella proprietà	hasSocialSecurityNumber
owl:InverseFunctionalProperty	Ogni oggetto è collegato a massimo un soggetto	isIdentifiedBy
owl:propertyChainAxiom	Permette di definire catene di proprietà	$hasFather, hasBrother \rightarrow hasUncle$
owl:equivalentProperty	Due proprietà sono semanticamente equivalenti	$hasDisease \equiv suffersFrom$
owl:disjointWith	Due proprietà non possono valere per la stessa coppia	isParentOf and isChildOf

Tabella 2.3: Relazioni tra oggetti in OWL

2.4.4.1 Vincoli sulle proprietà

OWL mette a disposizione dei costrutti di restrizione, usati per definire condizioni e regole sui valori di una proprietà. Per imporre una restrizione si ricorre al costrutto ***owl:Restriction***, in combinazione con ***owl:onProperty***, che specifica su quale proprietà si vuole imporre la restrizione. I vincoli imposti possono essere del tipo ***owl:allValuesFrom***, ***owl:someValuesFrom***, oppure ***owl:hasValue***. Si consideri il seguente esempio:

Listing 2.7: Esempio dell'uso dei costrutti di restrizione.

```
@prefix : <http://example.org/ontology#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

### Definizione di classi ed istanze.
:Disease a owl:Class .
:Symptom a owl:Class .
:Fever a :Symptom .

:hasSymptom a owl:ObjectProperty ;
    rdfs:domain :Disease ;
    rdfs:range :Symptom .

### Restrizione: hasSymptom deve avere come oggetto tutti i valori
```

```
    ↪ della classe Symptoms.
:StrictDisease a owl:Class ;
    owl:equivalentClass [
        a owl:Restriction ;
        owl:onProperty :hasSymptom ;
        owl:allValuesFrom :Symptom
    ] .

### Restrizione: hasSymptom deve avere almeno un oggetto
    ↪ appartenente alla classe Symptom.
:ObservableDisease a owl:Class ;
    owl:equivalentClass [
        a owl:Restriction ;
        owl:onProperty :hasSymptom ;
        owl:someValuesFrom :Symptom
    ] .

### Restrizione: l'oggetto di hasSymptom deve essere esattamente l'
    ↪ istanza Fever.
:FeverDisease a owl:Class ;
    owl:equivalentClass [
        a owl:Restriction ;
        owl:onProperty :hasSymptom ;
        owl:hasValue :Fever
    ] .
```

Altri tipi di vincoli sono quelli *di cardinalità*, che si ottengono mediante i costrutti:

- ***owl:minCardinality***: specifica il numero minimo di valori per una proprietà;
- ***owl:maxCardinality***: specifica il numero massimo di valori per una proprietà;
- ***owl:Cardinality***: specifica il numero esatto di valori per una proprità.[34]

[34]Per esempio, potrebbero essere riferite sempre alla proprietà hasSymptom.

2.4.5 Inferenza in OWL

Le proprietà che OWL mette a disposizione sono fondamentali, in quanto permettono ai sistemi automatici di dedurre relazioni e classificazioni implicite, rendendo possibile l'inferenza.
OWL adotta l'*Open World Assumption*(OWA) [10]: l'assenza di informazione non implica falsità - si assume che la conoscenza sia incompleta. Per esempio, la mancanza di dati che indicano che Bob è affetto da qualche patologia non indica che Bob non sia malato - potrebbe essere semplicemente un'informazione non ancora nota. Al contrario, i dataset tradizionali adottano la Closed World Assumption (CWA), dove la mancanza di informazione ne simboleggia la non veridicità.
Com'è possibile, nota un'ontologia in OWL, passare da dati strutturati[35] a dati del Web Semantico, su cui è possibile fare inferenza automatica?
Si consideri l'architettura [2] illustrata in figura 2.10. Si parte da un database relazionale, una base di dati tradizionale[36] in cui i dati sono organizzati in tabelle, righe, colonne. I dati in formato relazionale vengono convertiti in formato RDF da un Convertitore RDB to RDF, diventando così compatibili con il Web Semantico. Il file *.owl* è il file che contiene l'ontologia OWL vera e propria, quindi la definizione di classi, proprietà, vincoli e logica descrittiva; questo viene caricato, insieme ai dati in formato RDF, nell'OWL API [11]: libreria in Java che consente di caricare un file OWL contenente un'ontologia, interrogare, modificare o creare ontologie, fondere dati RDF con concetti dell'ontologia - è qui che entrano in gioco le classi, proprietè e restrizione definite nel file *.owl*. In questo modo, viene fornita una semantica ai dati: i dati vengono annotati semanticamente grazie all'ontologia e sono pronti sia per inferenza automatica, sia per essere interrogati con strumenti come SPARQL. Di seguito un esempio di quary SPARQL base:

Listing 2.8: Esempio di query SPARQL per trovare pazienti affetti da diabete.

```
1     PREFIX ex: <http://example.org/ontology#>
2
3 SELECT ?patient
4 WHERE {
5   ?patient a ex:Patient .
6   ?patient ex:hasDisease ex:Diabetes .
7 }
```

[35] Si tratta di dati organizzati in tabelle.
[36] Ad esempio, MySQL.

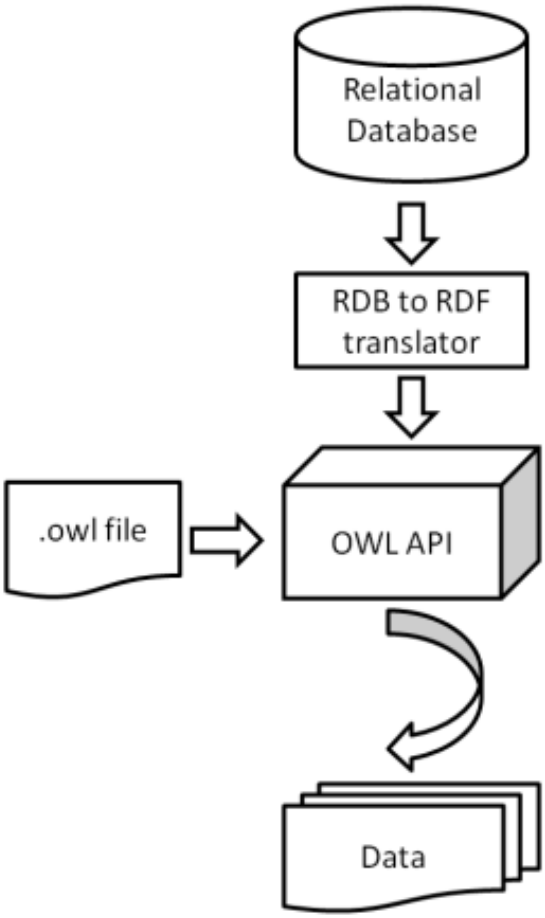

Figura 2.10: Esempio di architettura per l'integrazione di basi di dati relazionali con un'ontologia in OWL, tramite l'uso di una OWL API.

Fonte: `https://www.researchgate.net/figure/An-architecture-that-uses-OWL\protect\penalty\z@-API_fig2_301546678/download?_tp=eyJjb250ZXh0Ijp7I\protect\penalty\z@-mZpcnN0UGFnZSI6Il9kaXJlY3QiLCJwYWdlIjoiX2RpcmVjdCJ9fQ`

Capitolo 3

Principali ontologie in ambito medico

Questo capitolo tratterà l'analisi di alcune delle più utilizzate ontologie in ambito medico, implementate grazie ai fondamenti teorici illustrati nel capitolo precedente.

3.1 Criteri di scelta delle ontologie da trattare

L'ambito biomedico vanta un numero elevato di risorse ontologiche. Il portale BioPortal (`http://bioportal.bioontology.org/`), sviluppato dal National Center for Biomedical Ontology (NCBO) [20], funge da repository on line delle principali ontologie biomediche: permette di esplorare e visualizzare contemporaneamente più ontologie collegate fra loro, navigare all'interno di una specifica ontologia e recuperare risorse ontologiche inerenti un termine specifico. Sul portale è presente una classificazione delle ontologie in base al rispettivo numero di visite, aggiornata mensilmente.
Seppur BioPortal sia un'importante repository, non sarebbe opportuno considerare le sole ontologie con più visualizzazioni sul portale per risalire alle più utilizzate in ambito sanitario. Bisogna, infatti, anche considerare le ontologie utilizzate attraverso altre piattaforme o integrate direttamente nei database bioinformatici; un altro criterio di individuazione può essere il numero di citazioni in articoli scientifici. In figura 3.1, due grafici a barre in cui si riportano, rispettivamente, le stime[1] del numero di citazioni su Google Scholar e PubMed[2], e del numero di accessi e download su BioPortal, per ontologia.
Di seguito, verranno illustrate le cinque ontologie che vantano numeri più alti.

[1] Le stime sono aggiornate al 2025.
[2] Repository più diffuse al mondo di articoli scientifici.

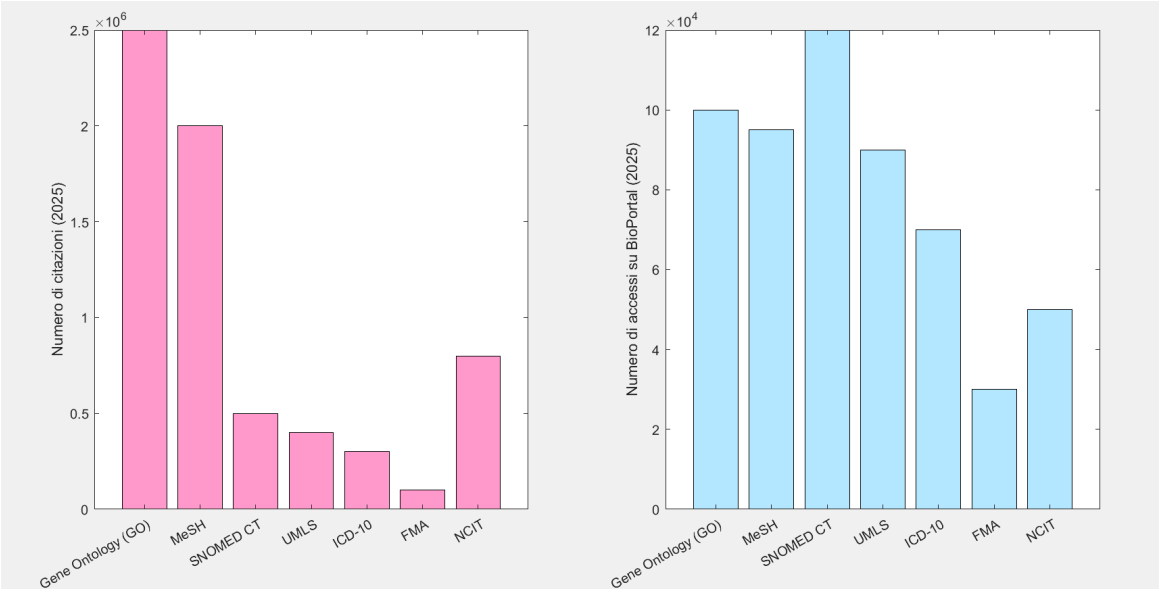

Figura 3.1: Nel grafico di sinistra: numeri di citazioni su PubMed e Google Scholar per ontologia. Nel grafico di destra: numeri di accessi per ontologia su BioPortal. Da notare che i due grafici non sono in scala.

3.2 SNOMED CT

SNOMED CT (Systematized Nomenclature of Medicine – Clinical Terms), risultato di uno sviluppo congiunto dell'NHS (National Healt Service), in Inghilterra, e del CAP (College of American Pathologists), è l'ontologia sanitaria più completa e multilingue al mondo [25]. Consente una rappresentazione coerente del contenuto clinico nelle EHR (Electronic Healt Records[3]) ed è in uso in più di ottanta paesi al mondo. SNOMED CT ha una profondità senza pari ma rimane, comunque, un prodotto in crescita e in evoluzione, con una SNOMED CT International Edition rilasciata mensilmente. Supporta lo sviluppo di contenuti clinici completi e di alta qualità nelle cartelle cliniche elettroniche, fornisce un modo standardizzato per rappresentare le frasi cliniche acquisite dal medico e ne consente l'interpretazione automatica. Permette, quindi, di:

i. standardizzare i termini medici usati nei sistemi sanitari;

ii. supportare la codifica di diagnosi, procedure, sintomi, farmaci e altri concetti clinici;

iii. facilitare l'interoperabilità tra sistemi informativi sanitari diversi, a livello nazionale e internazionale;

[3]Cartelle cliniche elettroniche.

iv. migliorare la qualità dei dati clinici, aiutando anche nell'analisi, nella ricerca medica e nella documentazione elettronica del paziente.

3.2.1 Componenti principali

I componenti principali di SNOMED CT, illustrati in figura 3.2, sono **concetti**, **descrizioni** e **relazioni**.

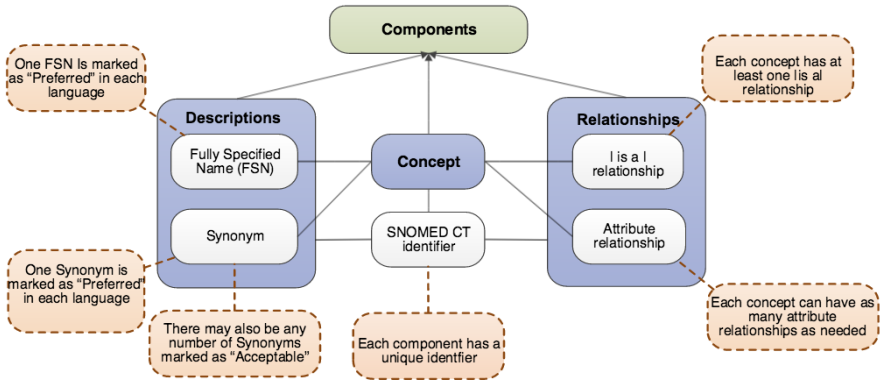

Figura 3.2: Schema di delle componenti fondamentali del modello SNOMED CT. Fonte: `https://jmeter-ci.blogspot.com/2017/05/understanding-snomed-ct-data-model.html`

1. *Concetti*: SNOMED CT contiene più di 311.000 concetti unici; ogni concetto rappresenta un termine clinico tramite uno *SNOMED CT identifier*, identificatore numerico univoco e machine-readable. Si consideri, per esempio, il codice 233604007: esso rappresenta la diagnosi di *Polmonite acquisita in comunità*; se riportato in EHR, permette di identificare in modo univoco la condizione diagnostica, facilitando la comunicazione tra professionisti della salute e la gestione elettronica dei dati sanitari.

2. *Relazioni*: una relazione rappresenta un'associazione tra due concetti, ed è usata per definire in modo formale il significato di un concetto affinchè possa essere capito da un computer. Esistono circa 1360000 relazioni semantiche in SNOMED CT; la relazione più usata è *is a*, che costituisce la spina dorsale tassonomica del sistema [12]; altri esempi sono *si trova in, fa parte di*. Un *relationship type* è inoltre usato per rappresentare il significato della relazione stessa.

3. *Descrizioni*: termini human-readable associati ai concetti. Ogni descrizione ha un *description type*, e può essere contrassegnata come *preferred for use* in specifici linguaggi. Un *FSN (Fully Specified Name)* è un tipo di descrizione che identifica completamente e

univocamente un concetto. Esistono anche sinonimi, descrizioni che permettono di descrivere lo stesso concept ID in modo differente. In ogni linguaggio, è marcato come *preferred* uno specifico sinonimo.

3.2.2 Struttura gerarchica

I concetti sono organizzati in una struttura gerarchica [24], dal generale al più specifico. Ciò offre la possibilità di registrare dati medici dettagliati e di visualizzarli, combinarli o esaminarli in un secondo momento. Sono presenti 19 categorie principali, le ***top-level hierarchies***; in tabella 3.1, sono presenti descrizioni ed esempi per ciascuna delle 19 categorie principali. In figura 3.3, un esempio di classi middle-level e bottom-level della categoria *Clinical finding*.

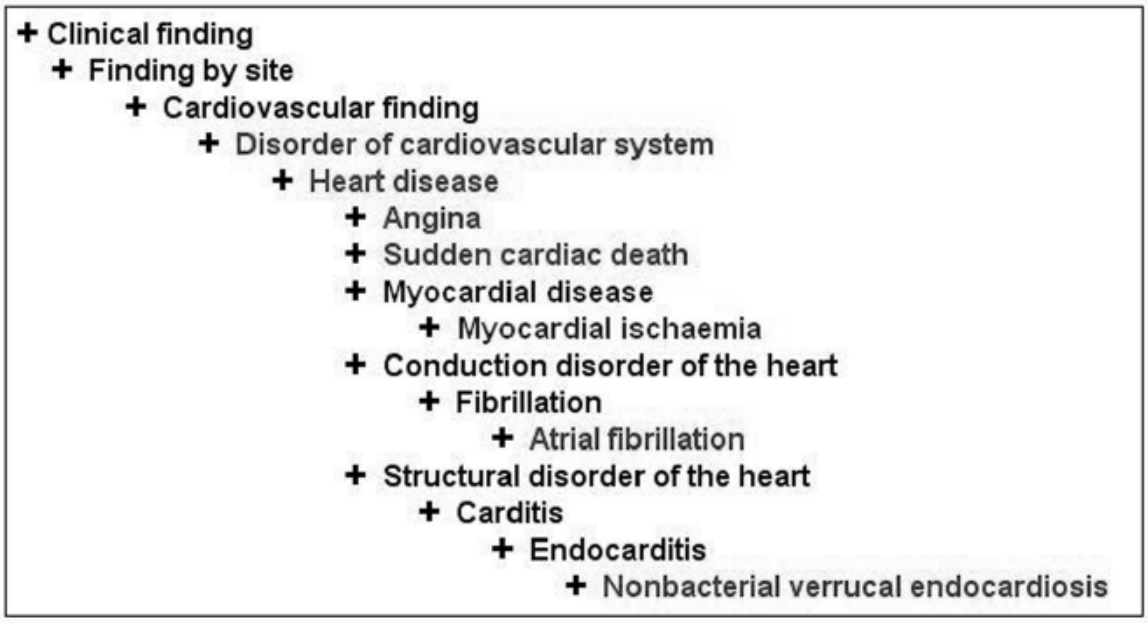

Figura 3.3: Esempio delle sottoclassi della categoria *Clinical Finding*. Si può notare la struttura gerarchica.
Fonte: `https://www.researchgate.net/figure/Hierarchical-structure-of-SNOMED-CT_fig2_200057553`

Inoltre, uno stesso concetto può appartenere a più categorie diverse, come si può notare in figura 3.4.
Di seguito, un esempio: il concetto SNOMED CT 82272006 definisce la classe di tutti i singoli casi di malattia che soddisfano i criteri per raffreddore: si individuano i sinonimi *Acute Coryza, Acute Nasal Catarrh, Acute Rhinitis, Common Cold*, oltre agli spagnoli *Resfrío Común, Rhinitis Infecciosa*. Perciò, un paziente con diagnosi di *Raffreddore Nasale* e un altro paziente con

diagnosi di *Coyza Acuta* sono entrambi considerati istanze di *Raffreddore Comune*. Tutte le singole istanze del raffreddore sono collegate dalla relazione di superclasse *is a*, che le include nella superclasse del *Raffreddore Comune*.

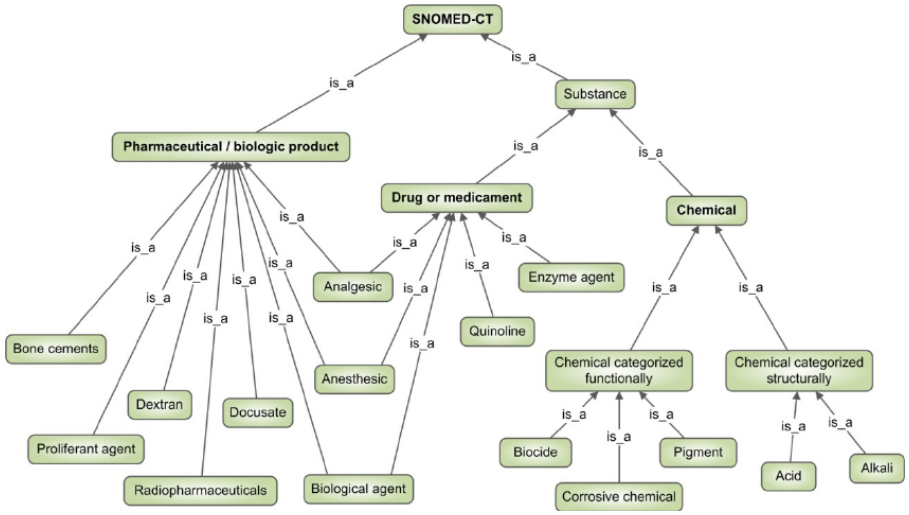

Figura 3.4: Illustrazione di alcuni concetti del vocabolario SNOMED CT. *Analgesic, Anesthetic, Biological agent* sono appartengono sia alla categoria *Drug or medicament*, che a *Biologic product*.

Fonte: `https://www.researchgate.net/profile/Marcos_Martinez\protect\penalty\z@-Romero/publication/236099124/figure/fig6/AS:299640834273330@1448409209476/Part-of-the-hierarchy-of-classes-of-SNOMED-CT-showing-some-of-the-relevant\protect\penalty\z@-classes-in.png`

3.2.3 DL in SNOMED CT

SNOMED CT usa vincoli semantici impliciti per descrivere i concetti [12]; questo consente ragionamenti automatici, come inferenza su ereditarietà, equivalenza o incoerenza. Infatti, le condizioni per appartenere a una classe sono implicitamente basate sulla Description Logic.
Si consideri un concetto come *Raffreddore*: esso ha il vincolo:

$$\text{«Common cold : 246075003 | Causative agent | = «Virus}$$

che corrisponde all'espressione logica del primo ordine:
forall x:istanza di $(x, Raffreddore) \rightarrow$ *esiste y:istanza di (y, Virus) e agente eziologico(y,x).*
Cioè: *per ogni istanza x che è un Raffreddore, esiste almeno un y, istanza di Virus, che causa* x . Formalmente:

$$\forall x\,(Raffreddore(x) \implies \exists y\,(Virus(y) \wedge agenteEziologico(y,x)))$$

Categoria	Descrizione	Esempio
Clinical finding	Condizioni cliniche, malattie, sintomi.	Raffreddore Comune
Procedure	Interventi chirurgici, trattamenti, esami.	Chirurgia cardiaca
Observable entity	Concetti osservabili, come misure o stati osservabili.	Pressione sanguigna
Body structure	Strutture anatomiche.	Cuore
Organism	Organismi come batteri, virus, funghi.	Escherichia coli
Substance	Sostanze chimiche, farmaci, materiali biologici.	Paracetamolo
Pharmaceutical/biologic product	Prodotti farmaceutici e biologici.	Insulina
Specimen	Campioni biologici.	Campione di sangue
Special concept	Concetti speciali.	Inesprimibile
Environment or geographical location	Luoghi, ambienti.	Ospedale
Physical object	Dispositivi, strumenti medici.	Elettrocardiografo
Physical force	Forze fisiche rilevanti in contesto medico.	Gravità
Event	Eventi - traumi o nascite.	Frattura ossea
Situation with explicit context	Situazioni che includono contesto, come la storia passata di una malattia.	Storia di infarto miocardico
Staging and scales	Stadi e scale di valutazione clinica.	Scadio || del cancro al seno
Social context	Fattori sociali, occupazioni, relazioni.	Fumatore attivo
Attribute	Attributi usati per definire altri concetti.	Laterality (laterale sinistro)
Qualifier value	Valori qualificatori utilizzati per specificare attributi.	Moderato
Linkage concept	Concetti che rappresentano relazioni tra altri concetti.	Complicazione di

Tabella 3.1: Elenco delle top-level hierarchies, con descrizione ed esempi.

Di seguito, un esempio di come rappresentare in RDF/Turtle la classe *Raffreddore* come sottoclasse di una restrizione, con *owl:someValuesFrom*. La restizione si applica alla proprietà *agenteEziologico*, e afferma che c'è almeno un valore per quella proprietà che è un'istanza della classe *Virus*:

Listing 3.1: Esempio di vincoli semantici in SNOMED CT, tramite l'utilizzo di OWL.

```
:Virus a owl:Class .
:agenteEziologico a owl:ObjectProperty .
:Raffreddore a owl:Class ;
    rdfs:subClassOf [
        a owl:Restriction ;
        owl:onProperty :agenteEziologico ;
        owl:someValuesFrom :Virus
    ] .
```

3.3 Gene Ontology (GO)

L'ontologia Gene Ontology (GO) [7] è la più completa fonte al mondo di informazione su geni, prodotti genici e loro funzioni. La conoscenza che rappresenta è sia human-readable che machine-readable, fungendo così da base per analisi computazionali di biologia molecolare su

larga scala e per esperimenti sulla genetica nel campo della ricerca biomedica. GO è un progetto bioinformatico atto a unificare la descrizione delle caratteristiche dei prodotti genici in tutte le specie.
In particolare, il progetto si propone di:

1. mantenere e sviluppare un vocabolario controllato, atto a descrivere i geni e i prodotti genici per ogni organismo vivente;

2. annotare i geni e i prodotti genici, e diffondere tali dati;

3. fornire strumenti per un facile accesso ai dati scaturiti dal progetto.

Il framework computazionale di GO consente quindi il confronto delle funzioni tra gli organismi e l'integrazione delle conoscenze tra diversi database biologici.

3.3.1 Macrocomponenti

GO è organizzata in tre sotto-ontologie, chiamate *aspects*: ***Molecular Function (MF), Cellular Component (CC)*** e ***Biological Process (BP)*** [7].

- Molecular Function: le funzioni molecolari (MFs) rappresentano attività a livello molecolare svolte da prodotti genici, come il processo di catalisi o l'attività di regolazione della trascrizione. Corrispondono sia ad attività che sono svolte da singoli prodotti genici, come proteine o RNA[4], sia ad attività svolte da complessi molecolari composti da più prodotti genici. I termini MF rappresentano attività, non le entità che le portano avanti; per questo, terminano sempre con il termine *activity*. *Protein kinase activity* individua un esempio di MF. Inoltre, non specificano il contesto, il luogo e lo spazio temporano in cui l'azione ha luogo.

- Cellular Component: localizza dove si trova il gene nella cellula; i CC includono strutture anatomiche cellulari, come *citoscheletro, mitocondrio, membrana plasmatica*, e componenti virali[5] come il *capside*.

- Biological process: i BP rappresentano operazioni o complessi di eventi molecolari con un inizio e una fine definiti, pertinenti al funzionamento di unità viventi integrate, quindi cellule, tessuti, organi e organismi. Un esempio è la *Riparazione del DNA* [7].

[4]RNA - acido ribonucleico: si tratta di una molecola polimerica implicata in vari ruoli biologici, quali la codifica, regolazione ed espressione dei geni, in particolare la sintesi proteica.

[5]I virus sono classificati a parte, in quanto non organismi cellulari.

3.3.2 Rappresentazione dei concetti

AmiGO[6] è il browser ufficiale della Gene Ontology, sviluppato dal Gene Ontology Consortium: si tratta di un'interfaccia web che permette di cercare termini GO, esplorare relazioni semantiche tra essi, visualizzare annotazioni GO per geni e proteine di molte specie, accedere a definizioni dettagliate, sinonimi, cross-reference e collegamenti a database esterni e scaricare annotazioni e termini GO in vari formati. Esistono più di 38000 termini GO; di seguito, è illustrata la rappresentazione di essi su AmiGO. Ogni termine è composto da molteplici elementi, alcuni obbligatori, altri facoltativi [7]. In figura 3.5, un esempio preso da AmiGO.

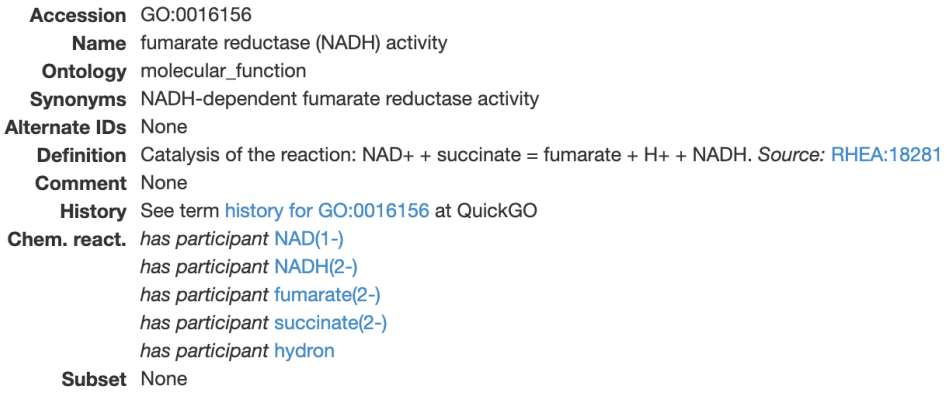

Figura 3.5: Screen Shot della pagina di AmiGO Browser relativa al termine *NADH activity*. Il nicotinammide adenina dinucleotide (NAD o NADH, a seconda dello stato di ossidazione) è un coenzima ossidoriduttivo: biomolecola il cui ruolo biologico consiste nel trasferire gli elettroni, quindi nel permettere le ossido-riduzioni.

Fonte: `https://www.geneontology.org/docs/ontology-documentation/`

Elementi obbligatori:

- **accession**, anche detto *Unique Identifier*: ogni termine ha un GO ID, identificativo univoco di sette numeri. Ad esempio, per identificare i tre aspetti principali MF, BP e CC, si ricorre rispettivamente agli identificativi *GO:0003674, GO:0008150, GO:0005575*;
- **term name**: nome human-readable assegnato ad ogni termine;
- **ontology (aspect)**: denota a quale delle tre sotto-ontologie appartiene il termine;
- **definition**: descrizione testuale di ciò che rappresenta il termine, insieme a una o più citazioni delle fonti da cui è presa l'informazione;

[6] http://amigo.geneontology.org

- **relationships to other terms**: è presente una *relations documentation page* apposita, in cui è descritto come il termine è relazionato ad altri termini. Tutti i termini[7] hanno almeno una relazione di sottoclasse *is a* che li lega ad un altro termine.

Possono, inoltre, essere presenti anche uno o più elementi opzionali, quali:

- **alternate ID**: si tratta di un ID secondario; l'esistenza di un ID secondario nasce quando due termini hanno lo stesso significato, e vengono quindi unificati. Tutti gli ID vengono conservati, in modo che nessuna informazione vada persa;

- **synonyms**: parole o frasi con significato molto vicino al nome del termine. La relazione tra il nome e un sinonimo è rappresentata dallo *scopo* del sinonimo:

Tipo di Sinonimo	**Scopo/Descrizione**
exact	Sinonimo esattamente equivalente al nome del termine; può essere usato in modo intercambiabile.
broad	Sinonimo con significato più ampio rispetto al termine.
narrow	Sinonimo con significato più specifico o dettagliato rispetto al termine.
related	Sinonimo correlato al termine, ma non equivalente in modo preciso.
custom types	Sinonimi con etichette personalizzate, ad es. *systematic synonym*, che sono considerati sinonimi exact.

Tabella 3.2: Tipi di sinonimi nella Gene Ontology e loro significato.

- **comment**: informazioni aggiuntive sul termine e il suo uso;

- **chem. react.**: per termini che presentanoriferimenti al *RHEA database of chemical reactions*[8], questa sezione elenca le reazioni in questione;

- **history**: traccia le modifiche che sono state apportate nel tempo al termine - quando è stato aggiunto, se è stato modificato, annotazioni come *was merged into*;

[7] Si intende, tutti i termini al di fuori dei tre aspetti principali.
[8] https://www.rhea-db.org/

- **obsolete tag**: valore booleano che indica se il termine è stato obsoleto, e non deve essere usato. Un termine diventa obsoleto quando è fuori ambito, nominato o definito in modo fuorviante, o descrive un concetto che sarebbe meglio rappresentato in un altro modo e deve essere rimosso dall'ontologia pubblicata. In questi casi, il termine e l'ID persistono ancora nell'ontologia, ma il termine viene etichettato come obsoleto e tutte le relazioni con altri termini vengono rimosse. Al termine, viene aggiunto un commento che descrive in dettaglio il motivo dell'obsolescenza e vengono suggeriti termini di sostituzione.

3.3.3 Struttura gerarchica

Ogni *aspect* ha un **root term**, un termine radice da cui si diramano tutti gli altri termini dell'aspetto: questo conferisce all'ontologia una struttura gerarchica, come è mostrato in figura 3.6.

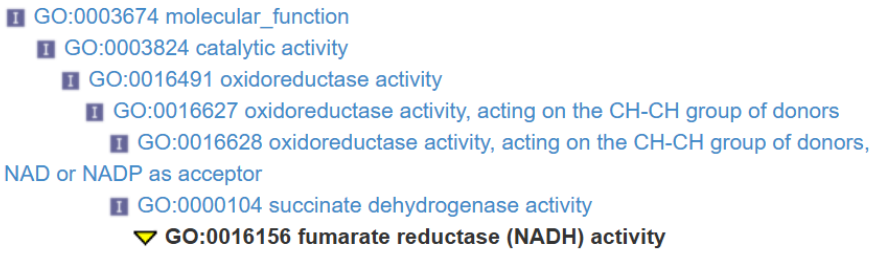

Figura 3.6: Albero gerarchico relativo al termine GO:00016156.
Fonte: `https://amigo.geneontology.org/amigo/term/GO:0035529`

I tre aspetti principali sono disgiunti, cioè non esistono relazioni *is a* tra termini appartenenti ad aspetti diversi. Un semplice esempio illustrativo: è corretta l'affermazione "*DNA repair* is a *biological process*", in quanto si tratta di due termini BP, mentre risulta scorretto "*DNA repair* is a *nucleus*", dato che si tratta di un BP e un CC. Nonostante ciò, è possibile instaurare altri tipi di relazioni tra termini di aspetti diversi, come *part of, occurs in*.
Inoltre, è possibile rappresentare i termini GO come nodi di un grafo, il **DAG - Directed Acyclic Graph**[9], in cui gli archi rappresentano le relazioni semantiche tra i concetti. Un esempio di DAG base è illustrato in figura 3.7.

[9]Si tratta di un grafo orientato, in cui cioè ogni arco ha un'origine e una destinazione, ed aciclico: seguendo il verso degli archi, è impossibile ritornare allo stesso nodo da cui si è partiti.

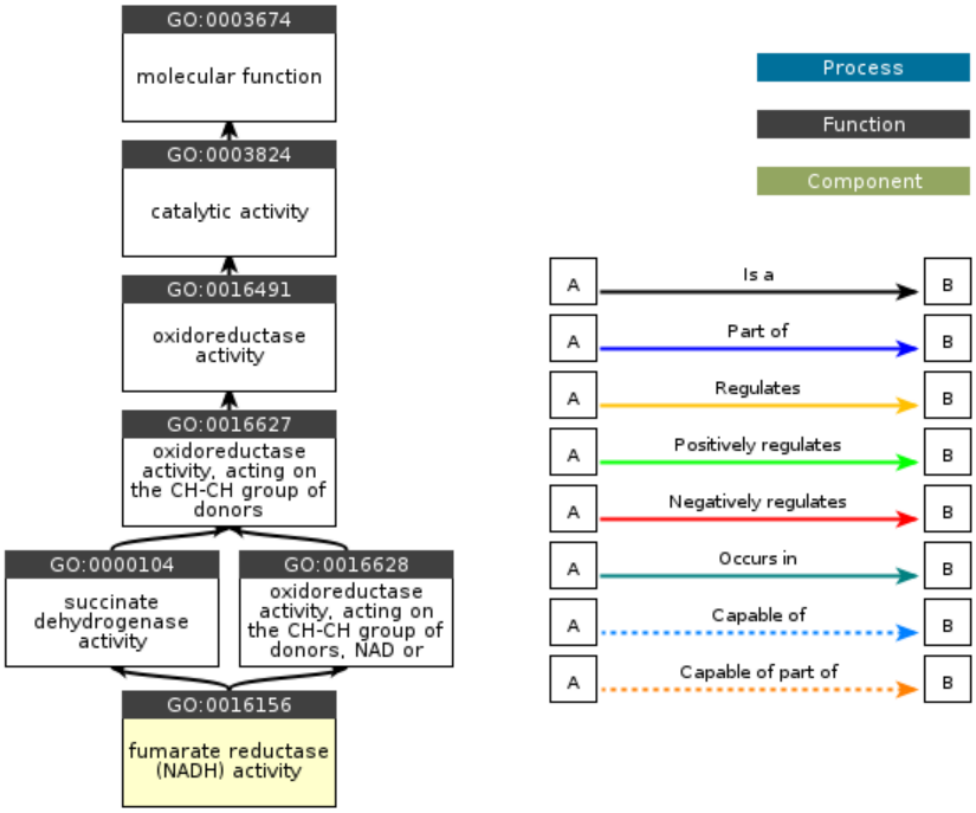

Figura 3.7: Un esempio di GO DAG, relativo all'esempio di figura 3.5. Si trova nella sezione *Graph Views* di AmiGO browser, ed è presente anche una legenda delle relazioni possibili.

Fonte: `https://www.geneontology.org/docs/ontology-documentation/`

3.3.4 GO Annotations e GO-CAM

Le annotazioni GO [5] sono collegamenti tra geni, o proteine, e termini GO.

Campo	**Valore**
Gene	TP53
GO term	GO:0003677 (DNA binding)
Aspect	Molecular Function
Evidence	IDA (Inferred from Direct Assay)
Reference	PMID:12345678

Tabella 3.3: Esempio di annotazione GO per il gene TP53. PMID sta per PubMed Identifier, codice numerico univoco assegnato a ogni pubblicazione scientifica indicizzata su PubMed.

Servono a descrivere cosa fa un gene, dove agisce e in quale processo biologico è coinvolto, in modo standard e computabile. Si tratta di un'informazione che contiene un gene, un GO term,

un'evidenza[10] e una fonte. Si consideri l'esempio in tabella 3.3: l'annotazione in questione indica che ci sono prove sperimentali dirette che il gene TP53 si lega al DNA. Un'evoluzione delle annotazioni sono le GO-CAM (Causal Activity Models): estensione della Gene Ontology che permette di modellare come le attività molecolari interagiscono tra loro all'interno di un processo biologico. Si tratta di una combinazione di annotazioni standard, per produrre una rete di annotazioni - un *modello*.

Le GO-CAM possono connettere informazioni diverse relative alla funzione di uno stesso gene, oppure funzioni di geni diversi, specificando come l'attività di un prodotto genico possa influire sull'attività di un altro. Sono rappresentate come grafi, dove i nodi corrispondo ad attività molecolari e gli archi a relazioni causali; usano OWL, quindi sono leggibili da software semantici e sono create e modificate tramite Noctua (http://noctua.geneontology.org/workbench/noctua-landing-page/)[11]. Si consideri l'illustrazione in figura 3.8: essa mostra come il modello permette di passare da annotazioni standard, riguardanti geni diversi, ad un'unica rete di attività interconnesse.

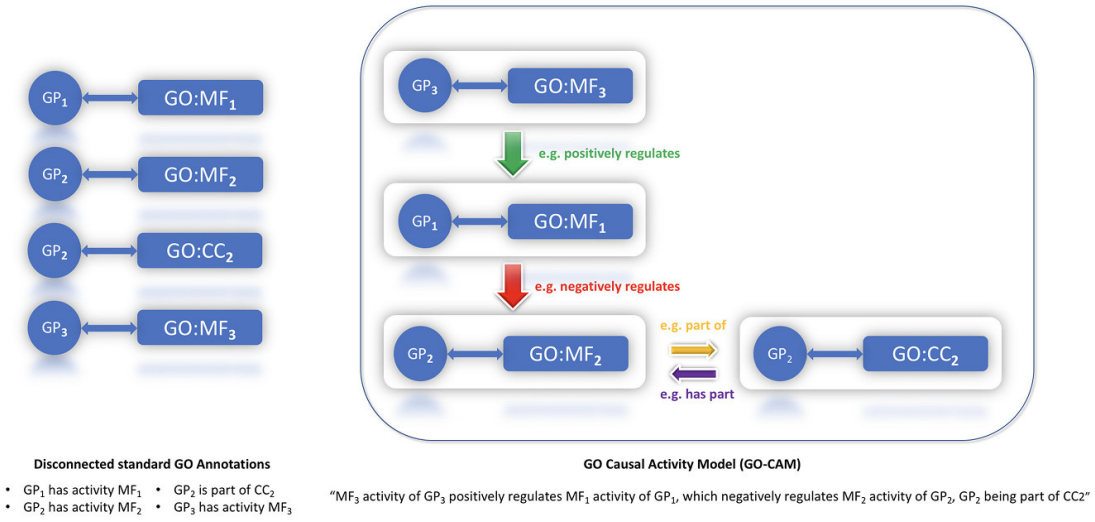

Figura 3.8: Principio del GO-CAM: sulla sinistra, quattro annotazioni standard; sulla destra, il modello che le unisce.

Fonte: `https://geneontology.cloud/home`

Esempi di relazioni che possono comparire in un GO-CAM sono: *enabled by/enables, part of, causally upstream of, occurs in, has substrate, positively/negatively regulates*.

Di seguito, è illustrato un esempio di GO-CAM rappresentato in RDF/Turtle; nell'esempio, il

[10] Come esperimenti, inferenze, letteratura.

[11] Strumento web-based sviluppato dal Gene Ontology Consortium, per la scrittura di annotazioni e la creazione e modifica di GO-CAM.

gene TP53 abilita l'attività molecolare DNA binding (GO:0003677), che è parte del processo di regolazione dell'espressione genica (GO:0010468).

Listing 3.2: Esempio di GO-CAM in RDF/Turtle.

```
@prefix : <http://model.geneontology.org/> .
@prefix go: <http://purl.obolibrary.org/obo/GO_> .
@prefix enabled_by: <http://geneontology.org/lego/enabled_by> .
@prefix part_of: <http://purl.obolibrary.org/obo/BFO_0000050> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

:activity_TP53_DNA_binding
    rdf:type go:0003677 ;  # DNA binding (Molecular Function)
    enabled_by: <http://identifiers.org/ncbigene/7157> ;  #TP53 gene
    part_of: :process_gene_expression_regulation .

:process_gene_expression_regulation
    rdf:type go:0010468 .  # regulation of gene expression (
      ↪ Biological Process)
```

I GO-CAM servono per rappresentare processi biologici in modo più preciso e meccanicistico, consentire inferenze automatiche e query più complesse, favorire una migliore comprensione dei percorsi biologici.

3.4 MeSH - Medical Subject Headings

Medical Subject Headings (MeSH) [26] è un dizionario standardizzato e gerarchico sviluppato dalla National Library of Medicine (NLM), usato per indicizzare, catalogare e cercare informazioni nel campo della medicina e della sanità. Include termini usati per indicizzare articoli scientifici su PubMed/MEDLINE[12], per descrivere le risorse nel Catalogo NLM e altre banche dati NLM e per garantire coerenza nelle ricerche. MeSH, quindi, fornisce termini standardizzati per concetti medici e scientifici e permette ricerche più precise e coerenti, anche quando gli articoli usano termini diversi.

[12]Cioè, che li classificano secondo argomenti standard.

3.4.1 Componenti MeSH

1. **Descriptors (Main Headings)**: si tratta dei termini principali usati per indicizzare e descrivere di cosa parla un documento. Si possono raggruppare in quattro classi:

   - *class 1 - Main Headings*: termini principali usati per indicizzare articoli su MEDLINE/PubMed; rappresentano il contenuto del documento. Ad esempio, se si effettua una ricerca su PubMed tramite il termine MeSH *Breast Neoplasms*, saranno mostrati tutti gli articoli relativi all'argomento *tumore al seno*, anche se nei titoli compaiono espressioni diverse, come *carcinoma mammorio* o *brest cancer*;
   - *class 2 - Publication Types*: indicano la tipologia di documento (ad esempio, *Review, Clinical Trial)*, non il contenuto, quindi servono come metadati;
   - *class 3 - Check Tags*: etichette standardizzate per categorie comuni, come *Sesso*; il check tag indicherà quindi *Male* o *Female*;
   - *class 4 - Geographics*: termini usati per indicare località geografiche. Indicano la localizzazione, non il contenuto scientifico;

2. **Qualifiers (Subheadings)**: usati insieme ai descriptors per specificare un aspetto particolare. Esistono 83 qualifiers, e si usano per specificare aspetti come diagnosi, fisiologia, terapia, effetti collaterali. Ad esempio, dato il descriptor *Liver*, un qualifier può essere */drug effects*: *Liver/drug effects* indica non il fegato in generale, ma l'effetto dei farmaci sul fegato;

3. **Entry Terms**: sinonimi o termini collegati che indirizzano al descriptor ufficiale. Ad esempio, *Breast Cancer* è un Entry Term che rimanda a *Breast Neoplasms*;

4. **SCRS (Supplememtary Concept Records)**: usati per concetti che non rientrano tra i Descriptors, come farmaci, sostanze chimiche, malattie rare, gruppi etnici;

5. **Scope Note**: definizione esplicita che descrive il significato del termine e come usarlo.

Ad ogni termine del vocabolario MeSH è assegnato ad un *MeSH Unique ID*: codice alfanumerico univoco, costituito da una lettera[13] e sei numeri. Ad esempio, al descrittore *MRI* è associato l'ID D008279.

[13] D per descriptors, Q per qualifiers, C od M per SCRs.

3.4.2 Struttura ad albero

I descrittori MeSH sono organizzati in 16 categorie, illustrate in tabella 3.4 [18].

Lettera	**Categoria MeSH**	**Lettera**	**Categoria MeSH**
A	Anatomia	I	Educazione, sociologia e fenomeni sociali
B	Organismi	J	Tecnologia, industria, agricoltura
C	Malattie	K	Persone
D	Sostanze chimiche e farmaci	L	Informazione e comunicazione
E	Tecniche diagnostiche, terapeutiche e attrezzature	M	Gruppi di popolazione
F	Discipline mediche	N	Sanità pubblica
G	Fenomeni fisiologici	V	Tipi di pubblicazione
H	Fenomeni patofisiologici	Z	Localizzazioni geografiche

Tabella 3.4: Categorie principali della gerarchia MeSH.

Ogni categoria è suddivisa in sottocategorie, in cui i descrittori sono disposti gerarchicamente su tredici livelli: da qui, il nome di *MeSH Tree Structure*. Ogni descrittore compare in almeno una posizione nell'albero, e può apparire in tutte le posizioni in cui risulta adeguato. All'interno del MeSH Browser (https://meshb-prev.nlm.nih.gov/search), ogni descrittore è seguito da un codice che ne identifica la posizione nell'albero, che inizia con la lettera relativa alla categoria corrispondente[14].
Ad esempio, se si ricerca nel browser il termine *Cancer*, si è reindirizzati al descrittore *Neoplasms* e, accedendo alla sezione *MeSH Tree Structures*, è possibile vedere ogni foglia dell'albero, come mostrato in figura 3.9.

I codici non hanno alcun significato intrinseco: D12.776.641 e D12.644.641 contengono le medesime ultime tre cifre, ma ciò non implica nessuna caratteristica comune. Un descrittore può anche essere seguito da più numeri addizionali, in formato più piccolo, che indicano altre locazioni dello stesso descrittore nell'albero. Nella sua totalità, MeSH contiene circa 30000 voci.

[14]Nel codice, la lettera è seguita da tre cifre per ogni sottolivello, separate tramite punti fermi.

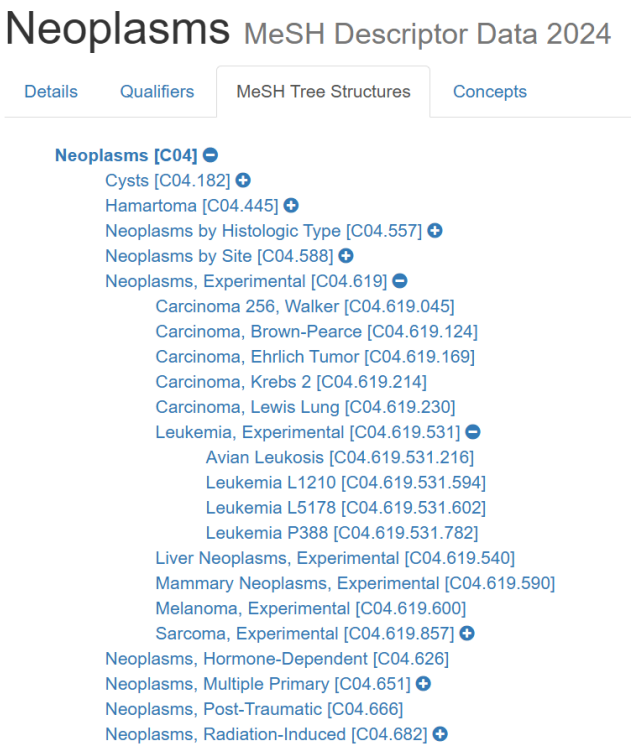

Figura 3.9: Esempio dei codici relativi all'albero del descrittore *Neoplasms*.
Fonte: `https://meshb-prev.nlm.nih.gov/record/ui?ui=D009369`

3.4.3 MeSH RDF

MeSH RDF [21] è una rappresentazione di MeSH che include un file scaricabile in formato N-Triples, un'interfaccia per query SPARQL, una SPARQL API e una interfaccia RESTful[15] per recuperare dati MeSH.

Nel vocabolario MeSH RDF, è utilizzato il prefisso *meshv:*[16]; all'interno, sono definite molteplici classi, alcune corrispondenti ad istanze di altre classi OWL più generiche, e molteplici proprietà. Un esempio di classe meshv è *meshv:DescriptorQualifierPair*, classe che accoppia un qualifier a un descrittore. Nel grafo in figura 3.10 vengono accoppiati i termini *mesh:D015242* e *mesh:Q000008*, e la coppia è un'istanza di *meshv:AllowedDescriptorQualifierPair*, sottoclasse di *meshv:DescriptorQualifierPair* insieme a *meshv:DisallowedDescriptorQualifierPair*.

3.5 UMLS Metathesaurus

UMLS [27] - Unified Medical Language System - è un'ontologia di concetti biomedici implementata dalla NLM, utilizzata principalmente dagli sviluppatori di sistemi di informatica medica.

[15]Basata sui principi dell'architettura REST (Representational State Transfer), in cui le risorse sono rappresentate da URI, usata per la comunicazione tra sistemi attraverso il protocollo HTTP.

[16]Ovvero, MeSH Vocabulary

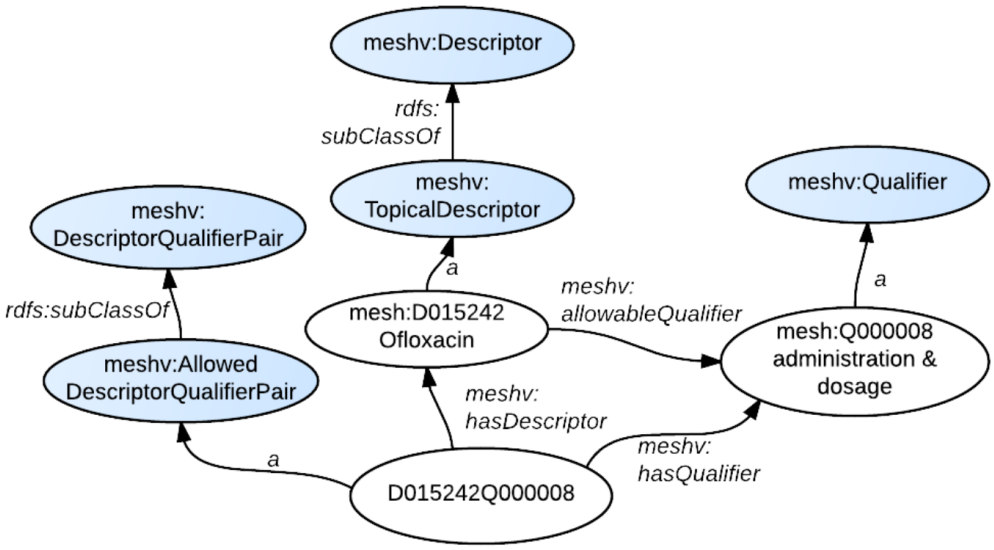

Figura 3.10: Esempio di grafo RDF che mostra come sono modellate le coppie descriptor-qualifier.

Fonte: `https://hhs.github.io/meshrdf/descriptor-qualifier-pairs`

Si tratta di un compendio di molteplici vcabolari nelle scienze biomediche; UMLS fornisce una struttura di mappatura tra questi e permette quindi una traduzione tra i vari sistemi terminologici. Inoltre, fornisce strutture per l'elaborazione automatica del linguaggio naturale (NLP). Lo scopo di UMLS è quello di mappare concetti equivalenti tra terminologie diverse: ogni concetto è infatti dotato di un *CUI (Concept Unique Identifier)*, che collega sinonimi e varianti provenienti da vocabolari diversi.
UMLS comprende più di un milione di concetti biomedici, che provengono da oltre 200 vocabolari, comprendenti SNOMED CT, Gene Ontology, MeSH, ICD-10[17] e LOINC[18].
L'ontologia non si basa nativamente su RDF, ma alcune sue componenti possono essere esportate o trasformate in RDF per applicazioni nel web semantico.

3.5.1 Semantic Network

La Semantic Network di UMLS [28] è una rete di concetti astratti, chiamati *Semantic Types*, e di relazioni semantiche:

[17]Ontologia utilizzata nel settore della sanità in Italia, ricopre solo concetti corrisponendti a diagnosi e procedure. Tutti i termini ICD sono compresi in SNOMED-CT, ragione per cui non è stata presa come oggetto di questa tesi.

[18]Vocabolario specializzato in risultati di test da laboratorio, misurazioni vitali, osservazioni cliniche e questionari.

- Semantic Types: categorie concettuali generali, identificate da un *TUI (Type Unique Identifier)*. Ogni concetto è mappato ad almeno un TUI, ma può averne anche più di uno. Nella versione attuale, sono presenti 127 Semantic Types. Ad esempio, il TUI T200 è riferito alla categoria *Disease or Syndrome*;

- Semantic Relationships: relazioni semantiche astratte, predefinite, che collegano i tipi semantici tra loro. Sono presenti 54 relazioni semantiche, come *treats, causes, associated_with, is_a*[19].

Una rappresentazione è riportata in figura 3.11.

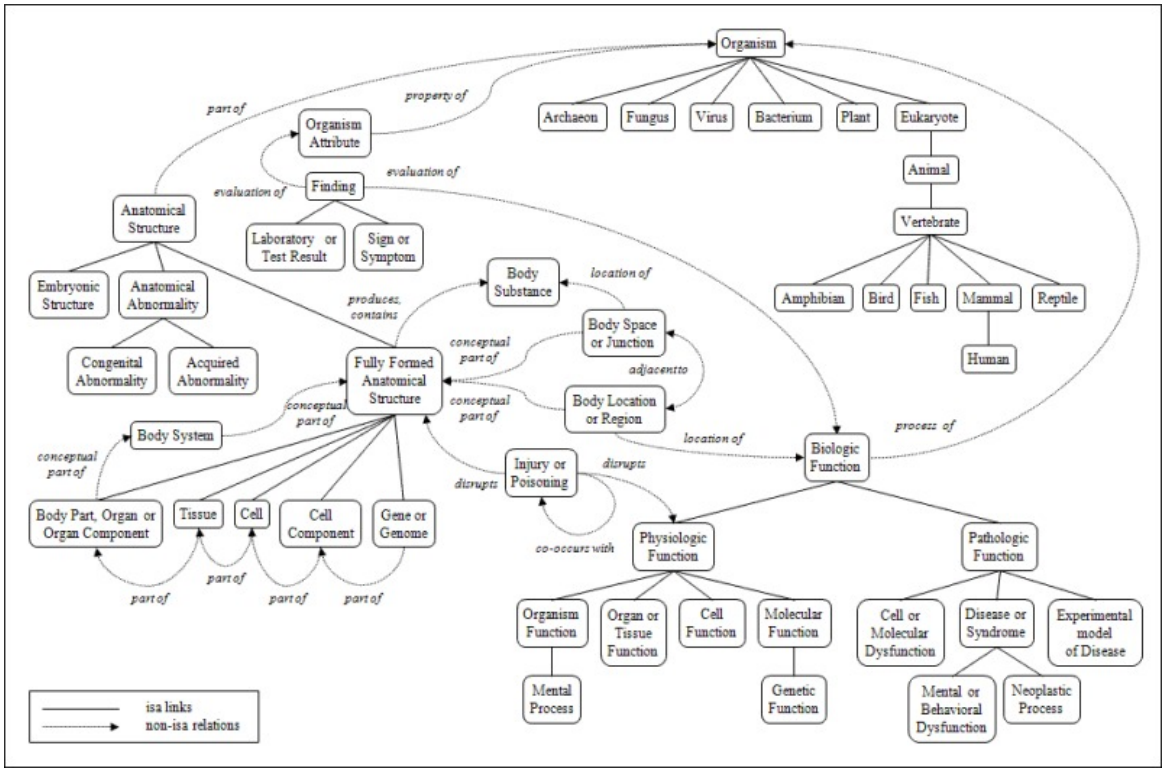

Figura 3.11: esempio di Semantic Network in UMLS, in cui i nodi sono tipi semantici e gli archi relazioni semantiche.

Fonte: `https://www.ncbi.nlm.nih.gov/books/NBK9679/`

Una semantic network permette la classificazione dei concetti e l'integrazione semantica[20], consente inferenze e controlli semantici tra entità mediche, favorendo il ragionamento autonomo, e supporta il data mining clinico, facilitando la ricerca di pattern e relazioni in grandi dataset.

[19] Is_a è una *subtype relation*

[20] Collega concetti da vocabolari diversi.

3.5.2 SPECIALIST Lexicon

UMLS adotta SPECIALIST Lexicon, risorsa linguistica computazionale progettata per supportare l'NLP - Natural Language Processing - in ambito biomedico e clinico. Si tratta di un lessico che fornisce informazioni morfologiche, sintattiche, ortografiche e semantiche su parole e frasi usate nella documentazione biomedica. Include abbreviazioni, forme irregolari e flesse (e.g.: *diagnose, diagnoses, diagnosed*), informazioni su derivazioni, categorie grammaticali e varianti ortografiche; permette di riconoscere la radice di una parola e ogni voce è accompagnata da una o più strutture sintattiche. Di seguito, un esempio di una voce SPECIALIST Lexicon, relativa all'entry *Diabetes*.

Listing 3.3: Esempio di voce SPECIALIST Lexicon

```
{base=diabetes #Forma base della parola
entry=E0003456 #ID univoco dell'entry
cat=noun #Categoria grammaticale: sostantivo
variants=regular #Il plurale segue una forma regolare, anche se
    ↪ raramente usato: diabeteses.
infl=diabetes|diabeteses
uninflected=diabetes
citation=diabetes #Forma citazionale, usata per riferimento
number=singular|plural #esiste sia al singolare che al plurale.
noun_type=common #sostantivo comune, non proprio
}
```

Un sistema NLP che usa lo SPECIALIST Lexicon può identificare *diabetes* come sostantivo anche se appare in contesti diversi, per esempio al singolare o al plurale. La forma plurale *diabeteses* è molto rara, ma è viene comunque inclusa per gestire casi limite e supportare parsing[21] robusto anche in documenti clinici non standardizzati.
Un insieme di programmi Java utilizza il lessico per elaborare le variazioni nei testi biomedici, mettendo in relazione le parole con le parti del discorso in cui si trovano: ciò risulta utile nelle ricerche attraverso una EHR.

[21] Processo di analisi di una stringa di simboli, sia in linguaggio naturale che in strutture dati.

3.6 NCIT

NCIT - National Cancer Institute Thesaurus [17] - è un'ontologia sviluppata dal NCI per rappresentare in modo semantico e strutturato concetti relativi al cancro, alla medicina clinica e molecolare, ai farmaci, ai trattamenti e alle malattie in generale. Viene utilizzato soprattutto nelle EHR, in clinical trials e per annotazioni di dati genomici, e comprende circa 120000 concetti chiave biomedici e 400000 relazioni.

3.6.1 Struttura gerarchica

Anche NCIT, come tutte le ontologie precedentemente citate, ha una struttura gerarchica, con concetti organizzati in classi e sottoclassi. Una relazione di sottoclasse è *is_a*, mentre altre relazioni semantiche esistenti sono, per esempio, *part_of, associated_with,may_treat,has_finding*. Inoltre, un concetto può avere più di un genitore - la gerarchia è un DAG.
In tabella 3.5 è riportato l'elenco delle categorie principali di NCIT.

Codice	**Categoria**	**Codice**	**Categoria**
C17021	Anatomical Structure	C20181	Gene
C25218	Biologic Function	C64379	Gene Product
C7057	Cancer-Related Concept	C1909	Medical Device
C25488	Chemical	C1908	Pharmacologic Substance
C15695	Diagnostic/Prognostic Factors	C28007	Therapeutic Procedure
C16210	Disease/Disorder/Finding	C25190	Organism
C25701	Research Activity	C25616	Occupation or Discipline
C25700	Conceptual Entity	C25626	Role
C19486	Property or Attribute	C25292	Event

Tabella 3.5: Categorie principali del NCIT.

Si può osservare come ad ogni concetto sia assegnato un codice univoco; si consideri l'esempio:

```
1 Cancer (C9305)
2     Lung Cancer (C4872)
3         Small Cell Lung Cancer (C4873)
4         Non–Small Cell Lung Cancer (C4874)
```

In questo esempio, C9305 è la superclasse *Cancer*; C4872 è la sottoclasse *Lung Cancer* e C4873, C4874 sono figli della sottoclasse. In formato RDF/Turtle, è possibile osservare le gerarchie e le relazioni tra i concetti, come mostrato nell'esempio sottostante.

Listing 3.4: Rappresentazione RDF/Turtle della gerarchia relativa al concetto NCIT *Cancer*.

```
@prefix ncit: <http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#>
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

# Definizione del concetto Cancer
ncit:C9305 a ncit:Concept ;
    rdfs:label "Cancer" ;
    rdfs:subClassOf ncit:Disease .

# Definizione del concetto Lung Cancer
ncit:C4872 a ncit:Concept ;
    rdfs:label "Lung Cancer" ;
    rdfs:subClassOf ncit:C9305 ;  # Relazione is_a con Cancer
    rdfs:subClassOf ncit:Disease ;
    ncit:hasFinding ncit:C7523 .  # Relazione con un finding (e.g.: sintomo o segno associato)

# Definizione di un sottotipo di Lung Cancer
ncit:C4873 a ncit:Concept ;
    rdfs:label "Small Cell Lung Cancer" ;
    rdfs:subClassOf ncit:C4872 ;  # Sottotipo di Lung Cancer
    ncit:hasFinding ncit:C7524 .  # Altro finding associato

# Definizione di un altro sottotipo di Lung Cancer
ncit:C4874 a ncit:Concept ;
    rdfs:label "Non-Small Cell Lung Cancer" ;
    rdfs:subClassOf ncit:C4872 ;  # Sottotipo di Lung Cancer
    ncit:hasFinding ncit:C7525 .  # Altro finding associato
```

Capitolo 4

Conclusioni

L'adozione di tecnologie semantiche e, in particolare, delle ontologie rappresenta un passaggio fondamentale nella gestione e nell'organizzazione della conoscenza, soprattutto in ambiti complessi e fortemente strutturati come quello biomedico. La possibilità di descrivere concetti, relazioni e gerarchie in maniera formalmente rigorosa consente non solo di migliorare la qualità e la coerenza dei dati, ma anche di promuovere una reale interoperabilità tra sistemi e istituzioni diverse.
Attraverso l'utilizzo del modello RDF e dei linguaggi come OWL, il Web Semantico fornisce una base solida per la costruzione di sistemi intelligenti, capaci di elaborare, inferire e collegare informazioni provenienti da fonti eterogenee. Le triple RDF, in particolare, costituiscono un meccanismo funzionale per rappresentare dati in modo strutturato, mantenendo la semantica necessaria per il ragionamento automatico. Questo approccio si è rivelato particolarmente efficace nel campo biomedico, dove la grande quantità e varietà delle informazioni richiede un linguaggio comune e formalizzato per poter essere gestita efficacemente.

Nel corso di questo lavoro, è stato possibile analizzare alcune delle ontologie più rilevanti nel dominio sanitario, tra cui SNOMED CT, Gene Ontology, MeSH, UMLS e NCIT. Queste risorse, ciascuna con livelli diversi di dettaglio e obiettivi applicativi specifici, si sono affermate come standard internazionali per la classificazione, la ricerca e l'analisi dei dati medici. La loro integrazione nei sistemi informativi clinici e di ricerca consente di migliorare l'accuratezza nell'annotazione dei dati, di supportare decisioni cliniche più informate e di favorire lo sviluppo di strumenti computazionali avanzati per la scoperta di conoscenza. Tuttavia, nonostante i notevoli progressi, permangono alcune sfide aperte, come la necessità di garantire allineamento e coerenza tra ontologie diverse, la semplificazione dell'accesso per utenti non esperti e il bi-

lanciamento tra espressività semantica e performance computazionale. Inoltre, l'integrazione semantica dei dati richiede un impegno congiunto tra informatici, medici, biologi e altri esperti del dominio, per assicurare che le ontologie riflettano fedelmente la complessità del sapere biomedico.
In prospettiva futura, l'uso delle ontologie potrà essere ulteriormente potenziato grazie all'integrazione con tecniche di intelligenza artificiale: questo potrebbe aprire la strada a nuovi strumenti diagnostici e sistemi di supporto clinico sempre più personalizzati.

In conclusione, le ontologie costituiscono uno degli elementi cardine per la costruzione di un ambiente informativo sanitario più coerente, accessibile e intelligente. La loro diffusione, e continuo perfezionamento, rappresentano un passo essenziale per affrontare le sfide poste dall'attuale trasformazione digitale della medicina e della ricerca scientifica.
Lo studio delle ontologie e delle tecnologie semantiche applicate al dominio biomedico ha permesso di coniugare aspetti teorici dell'informatica con esigenze concrete della medicina e della ricerca scientifica, evidenziando quanto l'organizzazione della conoscenza non sia soltanto una questione tecnica, ma anche un elemento chiave per il progresso della medicina moderna. In un'epoca in cui i dati sono sempre più abbondanti, ma non necessariamente più comprensibili, strumenti come le ontologie possono davvero fare la differenza nel trasformare l'informazione in conoscenza vera, accessibile e utile sia a professionisti che pazienti.

Bibliografia

[1] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *ScientificAmerican.com*, 05 2001.

[2] Aivars Bumāns. Towards ontology web search engine. *Baltic Journal of Modern Computing*, 7(4):532–543, 2019. Accessed: 2025-04-25.

[3] Wikipedia contributors. Semantic web stack — wikipedia, the free encyclopedia, 2024. Accessed: 2025-04-16.

[4] Mike Dean and Dan Connolly. Owl web ontology language reference. `https://www.w3.org/TR/owl-ref/`, 2004. W3C Recommendation 10 February 2004.

[5] Gene Ontology Consortium. Go annotations overview, 2023. Accessed: 2025-05-01.

[6] Gene Ontology Consortium. Go-cam: Gene ontology causal activity models - overview, 2023. Accessed: 2025-04-30.

[7] Gene Ontology Consortium. Gene ontology resource, 2025. Accessed: 2025-04-30.

[8] Thomas R. Gruber. Ontolingua: A mechanism to support portable ontologies. *Knowledge Acquisition*, 5(2):199–220, 1993.

[9] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5(2):199–220, 1993.

[10] Matthew Horridge. Common errors in owl ontologies. Presented at the Protege Conference 2004, 2004. Accessed: 2025-04-25.

[11] Matthew Horridge, Sean Bechhofer, et al. Owl api - a java api for owl ontologies. `https://owlapi.sourceforge.net/`, 2025. Accessed: 2025-04-25.

[12] ItaliaWiki. Snomed ct: utilizzo, accesso e struttura dei gruppi di concetti. https://italiawiki.com/pages/assistenza-sanitaria/snomed-ct-utilizzo-accesso-a-snomed-ct-struttura-gruppi-di-concetti.html#Groepen_van_begrippen, 2025. Accesso: 30 aprile 2025.

[13] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. Dbpedia – a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

[14] D.I. Manfred Lindner. Lecture 6.3 – world wide web (v4.3), 2005. Institut für Computertechnik, TU Wien.

[15] Frank Manola and Eric Miller. Rdf primer. https://www.w3.org/TR/rdf-primer/, February 2004. W3C Recommendation.

[16] Multinazionali Tech. Ontologia informatica: significato, esempi, applicazioni, strutture, 2023. Accesso il 18 aprile 2025.

[17] National Cancer Institute. Evs resources. https://wiki.nci.nih.gov/spaces/EVS/pages/50694975/EVS+Resources, 2024. Accessed: 2025-05-02.

[18] National Library of Medicine. MeSH Tree Structures, 2023. Accessed: 2025-05-02.

[19] N. Noy and Deborah Mcguinness. Ontology development 101: A guide to creating your first ontology. *Knowledge Systems Laboratory*, 32, 01 2001.

[20] Natalya F. Noy, Nigam H. Shah, Patricia L. Whetzel, Benjamin Dai, Michael Dorf, Nicholas Griffith, Clement Jonquet, Daniel L. Rubin, Margaret-Anne Storey, Christopher G. Chute, and Mark A. Musen. Bioportal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, $37(\text{suppl}_2) : W170 - -W173, 2009$.

[21] National Library of Medicine. Mesh rdf technical documentation. https://hhs.github.io/meshrdf/, 2021. Accessed: 2025-05-02.

[22] Stefano Pieroni, Marco Franchini, Francesco Mariani, Luca Fortunato, and Sabrina Molinaro. Ontologie e modellazione di dati sanitari: Attività di ricerca nell'ambito del progetto odinet. Technical report, Consiglio Nazionale delle Ricerche – Istituto di Fisiologia Clinica, Pisa, 2013. Versione 1.0.

[23] Giovanni Pilato, Daniela Di Fatta, and Giovanni Canfora. Ontologie e linguaggi ontologici per il web semantico. Technical Report ICAR/PA-04-05, ICAR-CNR, Università di Palermo, 2004.

[24] SNOMED International. Snomed ct editorial guide. https://confluence.ihtsdotools.org/display/DOCEG/SNOMED+CT+Editorial+Guide, 2025. Accessed: 2025-04-30.

[25] SNOMED International. What is snomed ct? https://www.snomed.org/what-is-snomed-ct, 2025. Accessed: 2025-04-29.

[26] U.S. National Library of Medicine. Medical subject headings (mesh) - mesh browser, 2024. Accessed: 2025-04-30.

[27] U.S. National Library of Medicine. Unified medical language system (umls), 2025. Accessed: 2025-05-02.

[28] Wikipedia contributors. Unified medical language system — wikipedia, the free encyclopedia. https://en.wikipedia.org/wiki/Unified_Medical_Language_System, 2025. Accessed: 2025-05-02.

[29] Wikipedia contributors. Web semantico — wikipedia, l'enciclopedia libera, 2025. Ultimo accesso il 16 aprile 2025.