Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Grounding Medical Terms in MeSH RDF: Resolution Coveage and Retrieval Efficiency

Bentounsi Imene[1], Menidjel Ramy Mohamed Amine[2], Zahri Ibrahim El Khalil[2], Kenidra Billel[3]

[1]*LIRE Laboratory, University of Constantine 2 Abdelhamid Mehri, Constantine, ALGERIA*

[2]*Department of Software Technologies and Information Systems, University of Constantine 2 Abdelhamid Mehri, Constantine, ALGERIA*

[3]*LIRE Laboratory, University of Constantine 1, Constantine, Algeria*

ARTICLE INFO

Received: 08 May 2026

Revised: 01 July 2026

Accepted: 14 July 2026

ABSTRACT

Introduction: Linked datasets built with semantic-web technologies are large and growing in number, which makes querying them expensive and makes prior knowledge of their content valuable for targeted retrieval.

Objectives: This paper addresses two coupled problems in the medical domain using Linked Data principles: grounding medical term mentions in the Medical Subject Headings (MeSH) vocabulary — resolving a surface mention to its concept — and mapping the resulting terms, including compound candidate terms (multi-word expressions that may contain two or more candidate terms), by their most specific broader descriptor.

Methods: We exploit the Medical Subject Headings (MeSH) controlled vocabulary published by the U.S. National Library of Medicine (NLM) as RDF, and its public SPARQL endpoint, to assign each input term the most specific broader descriptor(s) that subsume it in the MeSH polyhierarchy. Because a term may appear in several MeSH hierarchies, the method naturally returns one or more class labels. We then propose an extension that classifies compound terms absent from MeSH by decomposing them into components and computing their least common subsumer over MeSH tree numbers. We describe a three-stage pipeline (input validation, SPARQL interrogation over HTTP, and descriptor extraction), the auto-generated SPARQL query at its core, and an evaluation against the live MeSH endpoint.

Results: Measured against a tree-number-walking alternative, the single direct query returns the same classes about four times faster (warm median 543 ms vs. 2178 ms) using one HTTP round-trip instead of three to five. On a constructed benchmark spanning four MeSH domains (cardiology, oncology, pulmonology, neurology), the LCS extension assigns a correct class to all 3570 intra-domain concept pairs and correctly abstains on all 8207 cross-domain pairs, whereas on a free-text clinical corpus it recovers none of eleven terms — results that delimit the method's scope precisely: it classifies compounds built from indexed concepts, not arbitrary free-text phrases. On a standard disease-mention corpus (NCBI Disease), enhancing the lookup with synonyms, case variants, and abbreviation expansion raises resolution coverage from 1.5% to 48.8%, while classification remains reliable once a term is resolved.

Conclusions: Overall, the direct SPARQL-based method classifies medical terms in MeSH quickly and reliably, The main challenge remains resolving terms from real clinical text, and synonym and abreviation matching appear to be the most promising way to improve this.

Keywords: Linked Data; MeSH RDF; SPARQL; term classification; compound terms; biomedical informatics; semantic web; controlled vocabulary; least common subsumer.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

1. INTRODUCTION

The Web began as a collection of human-readable documents identified by URLs and interlinked by hypertext. These documents describe real-world resources in natural language that machines cannot interpret directly. The *Web of Data* extends this model by publishing data in machine-readable formats and connecting them through Uniform Resource Identifiers (URIs), so that both people and machines can collect, combine, and reuse them [1], [2]. *Linked Data* is the set of design principles that make this interlinking possible: use URIs to name things, use HTTP URIs so they can be dereferenced, return useful information using standards such as RDF and SPARQL, and include links to other URIs [3].

In the biomedical field, Linked Data has become a practical necessity. Clinical and medical information is overwhelmingly expressed in natural language and therefore cannot be processed reliably by standard web technologies without prior structuring [4]. Linked Data supports the construction of reference repositories, the alignment of medical ontologies, the handling of legal constraints around personal data, and the qualification and reuse of data across institutions [2], [5].

Among the pioneers of medical term organization is the U.S. National Library of Medicine (NLM), which created the *Medical Subject Headings* (MeSH) thesaurus to index and retrieve the biomedical literature [6], [7]. MeSH is now published as Linked Data (MeSH RDF) and is queryable through a public SPARQL endpoint, which makes it an exceptionally rich source for resolving and organizing medical terminology [8]. MeSH RDF itself comprises on the order of 154.7 million triples [8], and querying large public SPARQL endpoints is both costly and unreliable: across 427 endpoints monitored over 27 months, even simple lookups averaged 270–400 ms (some taking several seconds), generic-query performance varied by 3–4 orders of magnitude, and only 32.2% of endpoints reached high monthly availability [20]. Knowing the content of such datasets in order to issue *targeted* queries is therefore valuable.

This paper studies the following problem: given a compound candidate term in the medical domain, classify it according to its lowest (most specific) common ancestor in the MeSH hierarchy. A compound candidate term is a term that contains at least two candidate terms; candidate terms themselves are the longest sequences of nouns and adjectives extracted from a corpus [9]. Because a term can be classified only once it has been *resolved* (grounded) to a MeSH concept, we also study this upstream resolution step on real-text mentions. Our contributions are:

1. A taxonomic assignment method that issues a single, directly targeted SPARQL query against the NLM MeSH endpoint to retrieve the broader descriptor(s) of an input term, overcoming the incomplete-hierarchy limitation of the interactive MeSH browser and the implementation overhead of tree-number manipulation;
2. An extension that classifies compound terms absent from MeSH by decomposition and least common subsumer (LCS) over tree numbers, thereby generalizing classification beyond verbatim vocabulary membership;
3. A quantitative evaluation against the live MeSH endpoint: the single direct query returns the same classes about four times faster than tree-number walking using one HTTP round-trip instead of three to five, and the LCS extension classifies 3570/3570 intra-domain concept pairs across four MeSH domains while correctly abstaining on 8207/8207 cross-domain pairs;
4. A study of *resolution coverage* on a real corpus (NCBI Disease): exact preferred-label matching resolves only 1.5% of disease mentions, whereas matching against synonyms (entry terms), case variants, and expanded abbreviations raises coverage to 48.8% (45.4% to the correct concept), localizing the bottleneck at resolution rather than classification.

The remainder of the paper is organized as follows. Section 2 reviews related work. Section 3 introduces the necessary background on MeSH RDF and SPARQL. Section 4 presents the proposed approach and its extension. Section 5 reports case studies, and Section 6 reports the evaluation. Section 7 discusses limitations, and Section 8 concludes.

2. RELATED WORK

Terminology and term extraction. A *term* is a lexical unit whose meaning is defined relative to a domain of specialty rather than a free linguistic context [9], [10]. Automatic extraction of multi-word terms from text is a long-standing task; the C-value/NC-value method of Frantzi, Ananiadou, and Mima remains a reference for recognizing

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

nested and compound terms by combining statistical and linguistic evidence [9]. Our work begins *after* extraction: the compound candidate terms are inputs to be classified.

Controlled biomedical vocabularies. MeSH is a controlled vocabulary maintained by the NLM for indexing and retrieving the biomedical literature [6], [7]. It is one component of the broader Unified Medical Language System (UMLS), which integrates MeSH, SNOMED CT, and many other sources into a common metathesaurus [11]. Mapping free biomedical text onto such vocabularies is the role of tools such as MetaMap [12]. Whereas MetaMap focuses on recognizing concepts in text, our problem is the *organization* of already-identified compound terms into the MeSH hierarchy.

Biomedical concept normalization and entity linking. Resolving a free-text mention to its concept in a controlled vocabulary — *normalization*, or *entity linking* — is the upstream task on which our grounding step builds. Dictionary- and learning-to-rank approaches such as DNorm normalize disease mentions against MeSH and outperform string-matching baselines like MetaMap on the NCBI Disease corpus [23], and TaggerOne models recognition and normalization jointly [24]. More recent work learns dense representations of concept names: BioSyn aligns mentions with synonyms by contrastive synonym marginalization [25], and SapBERT self-aligns biomedical entity representations and reaches state-of-the-art entity linking against UMLS [26]. Our resolution step is deliberately lightweight — exact, synonym, and abbreviation matching expressed directly as SPARQL over MeSH RDF — and we regard these neural normalizers as the natural route to close the residual coverage gap (Section 7).

Linked Data and the semantic web. The semantic web envisions machine-interpretable links between datasets [1], realized through Linked Data best practices [3] and surveyed in [2] and [13]. In the life sciences, Bio2RDF was an influential effort to expose heterogeneous bioinformatics databases as interlinked RDF and to query them with SPARQL [14], demonstrating the value of a Linked-Data substrate for biomedical knowledge integration — the same substrate on which MeSH RDF is built.

Taxonomy-based classification and semantic similarity. Methods that exploit the structure of a taxonomy frequently rely on the *least common subsumer* (the lowest common ancestor of two concepts). Resnik's information-content measure and the Wu–Palmer measure both derive concept relatedness from this shared ancestor [15], [16], and these ideas have been adapted to the biomedical domain over ontologies such as SNOMED CT [17]. Our approach is related in spirit — we also classify by the most specific subsuming descriptor — but it is *structural and explicit*: rather than computing a numeric similarity score, we read the MeSH polyhierarchy directly through SPARQL and return the broader descriptor(s) as class labels.

3. BACKGROUND: MESH RDF AND SPARQL

3.1 MESH RDF

MeSH RDF is a Linked-Data representation of the biomedical vocabulary, designed to facilitate sharing and linking of data that were historically distributed as XML [8]. Every statement is encoded as an RDF *triple* (subject, predicate, object); a collection of triples forms a knowledge graph that can be queried with SPARQL.

MeSH is organized into 16 top-level categories (A–Z), each divided into sub-categories that contain descriptors arranged from the most general (highest in the tree) to the most specific (lowest), with up to about thirteen levels of depth. The vocabulary is structured on three levels: descriptors (main headings) and qualifiers (subheadings); concepts, which are groups of synonymous terms; and terms, the lowest nodes, grouped into concepts.

Each descriptor carries one or more *tree numbers* that encode its position(s): an alphabetical prefix denotes the category (e.g., "C" for *Diseases*) and a dotted numeric suffix denotes the location in the hierarchy (e.g., *Heart Diseases* [C14.280]). Because a descriptor can belong to several hierarchies, it can have several tree numbers — this is the polyhierarchy. For example, *Nose Diseases* is identified by both [C08.460] and [C09.603]. The vertical relation between a descriptor and its immediate broader descriptor is expressed by the property meshv:broaderDescriptor.

3.2 SPARQL

SPARQL ("SPARQL Protocol and RDF Query Language") is the W3C standard query language and protocol for RDF data [18], [19]. It treats data as a directed, labeled graph of triples; a query consists of triple patterns whose variables

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

are bound by matching them against the dataset. A SELECT query has three parts: a *prologue* (BASE/PREFIX declarations), a *head* (SELECT and the projected variables), and a *body* (the WHERE graph pattern, with optional FROM, FILTER, OPTIONAL, ORDER BY, LIMIT, OFFSET). MeSH RDF exposes a SPARQL endpoint over HTTP at http://id.nlm.nih.gov/mesh; queries can be transported via HTTP GET/POST and results returned as HTML, XML, CSV, or JSON. In this work, results are consumed as JSON.

4. PROPOSED APPROACH

4.1 OBJECTIVE

The goal is to assign to each compound candidate medical term its most specific broader descriptor(s**)** in MeSH — the lowest ancestor that subsumes it. Because MeSH is a polyhierarchy, a term may be subsumed in several places; the method therefore returns a *set* of class labels rather than a single one.

4.2 DESIGN ALTERNATIVES CONSIDERED

We explored three retrieval strategies before settling on the final one:

5. **MeSH browser.** Convenient for manual lookup, but it returns only a *partial* hierarchy. For the compound term *Conjunctival Neoplasms*, the browser roots the returned subtree at *Neoplasms*, hiding the higher ancestor that actually exists. This truncation is problematic when the sub-terms of a compound are distant in the hierarchy.
6. **Tree-number truncation.** Because tree numbers are unique positional identifiers, one can ascend the hierarchy by removing trailing digit groups. For *Conjunctival Neoplasms* [C11.187.169], removing the last group yields [C11.187] (the immediate broader descriptor in that hierarchy); removing two groups yields [C11]. This is feasible but implementation-heavy, especially across multiple hierarchies.
7. **Direct** broaderDescriptor query (chosen). Issuing a single SPARQL query that retrieves the broader descriptor(s) of the input term is the most direct path: it returns exactly the class label(s) we need, across all hierarchies, in one round trip.

4.3 Classification Pipeline

The method has two stages:

8. **MeSH RDF interrogation.** A SPARQL SELECT query is directly formulated and submitted by the user. This query is encapsulated as a parameter within an HTTP request sent to the MeSH RDF endpoint. If the query successfully matches existing concepts, a JSON response containing the broader descriptor(s) is returned; otherwise, a 'not found' message is generated
9. **Classification.** The application parses the JSON, extracts the broader descriptor label(s) — the most specific parent(s) — and displays them as the term's semantic category/categories.

4.4 Core Sparql Query

The query at the heart of the method binds the input term by its English label and retrieves the labels of its broader descriptors:

```
PREFIX rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
PREFIX meshv: <http://id.nlm.nih.gov/mesh/vocab#>
PREFIX mesh:  <http://id.nlm.nih.gov/mesh/>
SELECT DISTINCT ?result
FROM <http://id.nlm.nih.gov/mesh>
WHERE {
 ?child  rdfs:label          "<TERM_TO_CLASSIFY>"@en .
```

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

?child meshv:broaderDescriptor ?parent .

?parent rdfs:label ?result .

}

The placeholder <TERM_TO_CLASSIFY> is substituted at run time with the validated input term. The DISTINCT modifier removes duplicate labels, and FROM <http://id.nlm.nih.gov/mesh> scopes the query to the MeSH graph.

4.5 Base Algorithm

Input: Q (a SPARQL SELECT query, as a string)

Output: the set of broader-descriptor labels, or an error

1.response <- httpRequest(endpoint, Q) // GET/POST to the MeSH SPARQL endpoint

2.if response is empty then

3. return "term not found in MeSH"

4.return parseDescriptors(response) // Extract labels from the JSON response

4.6 Extension: Classifying Unseen Compounds Via Decomposition And Least Common Subsumer

The base method classifies a term only if that exact term exists in MeSH as a descriptor or entry term. To organize compound terms that are *absent* from MeSH — yet whose components are present — we add a second route based on the *least common subsumer* (LCS), the lowest node that subsumes all components.

The key observation is that MeSH tree numbers encode the path from the root, so the least common subsumer of several components is simply the longest common prefix of their tree numbers. For example, if two components resolve to C14.280.647.250 and C14.280.647.500, their longest common prefix is C14.280.647 (*Myocardial Ischemia*), which becomes the class of the compound. If the components share no common prefix (different category letters), the term is genuinely cross-category and has no common ancestor — a meaningful outcome rather than an error.

Input: C (The set of terms that compose a compound term absent from MeSH)

Output: the LCS descriptor label(s), or "no common ancestor"

1. for each component c in C:

2. TN[c] <- treeNumbers(c) // SPARQL: meshv:treeNumber / rdfs:label

3. prefix <- argmax over (one tree number per component) of longest common dotted-segment prefix // Compute the argmax of the longest common dotted prefix

4. if prefix is empty then

5. return "no common ancestor (cross-category term)"

6. return descriptorsAt(prefix) // SPARQL: descriptor whose tree number = prefix

Under polyhierarchy a component may have several tree numbers; step 3 selects the combination that yields the deepest (most specific) common prefix. For compounds of more than two components, the LCS is obtained by reducing pairwise.

5. CASE STUDIES

We tested the base method on medical terms drawn from medical texts, from the simplest (single-word) to four-word compounds and terms with special characters, cross-checking each returned descriptor against the MeSH browser.

- **Single-word term — *Troponin*.** Confirms a non-compound term is correctly looked up and its broader descriptor returned.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

- **Two-word compound — *Gray Matter*.** Returns *Brain* and *Spinal Cord*, because the term appears in two distinct MeSH hierarchies — correct handling of polyhierarchy.
- **Multi-word compound — *Superficial Musculoaponeurotic System*.** Returns *Face* and *Neck*, consistent with the term's hierarchies.
- **Term with a special character — *17-Ketosteroids*.** The validation step accepts the hyphenated input, and the method returns *Ketosteroids* and *Adrenal Cortex Hormones*.
- **Four-word compound — *Cardio Ankle Vascular Index*.** Confirms behavior on longer compounds.

For terms registered in MeSH, the method retrieves the correct most-specific ancestor(s) and correctly returns multiple class labels under polyhierarchy. MeSH is extensive but not exhaustive, and it is updated on weekdays; some clinically attested expressions are not yet registered, and the base method returns no result for them. Table I lists the compound terms encountered during testing that the base method could not classify.

Table I. Compound candidate terms not classifiable by the base method (absent from MeSH).

Term	Term
appropriate treatment	conventional anti-ischemic treatment
circumflex occlusion	inter-ventricular anterior distal
cardiac consultation	coronary artery radiography exploration
frontal cutting	heterogeneous thinning anterior wall
collateral occlusion	irregular inter-ventricular
transverse cutting	negative clinical test
sagittal cutting	normal systolic function
persantine injection	ambulatory inferior myocardial
moderate concentric	inter-ventricular anterior
multifactorial atheromatous	long occlusion

Most of these are free-text clinical phrases rather than standardized headings, which explains their absence from a curated vocabulary; many, however, contain a component that *is* present in MeSH and are therefore candidates for the LCS route of Section 4.6.

6. EVALUATION

6.1 Experimental Setup

We evaluate the approach directly against the public NLM MeSH SPARQL endpoint (https://id.nlm.nih.gov/mesh/sparql), querying the current (2026) MeSH release. Because MeSH RDF is republished on weekdays, we fix and report the release year to make the experiment reproducible. All queries were issued from a single client machine over a standard internet connection; we therefore report latency as a property of the *public-endpoint setting* the method is designed for, not of a local triple store.

The test set has two parts. The first is a set of four terms registered in MeSH (Section 5); on these we compare the two retrieval strategies, the direct broaderDescriptor query and the tree-number ascent. The second is a set of eleven clinically attested compound terms absent from MeSH (Table I); on these the direct method returns nothing, and we measure how many the decomposition-plus-LCS method (Section 4.6) recovers.

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Each (term, method) pair is executed ten times: one *cold* run (first request) and nine *warm* runs. We report cold latency, the mean, median, and 95th-percentile warm latency (the percentile guards against outliers the mean hides), and the number of HTTP round-trips each method requires, as well as *coverage* (fraction of terms for which a class is returned). Correctness is assessed by checking each returned descriptor against the term's hierarchy in the MeSH browser.

6.2 Efficiency: Direct Query Vs. Tree-Number Ascent

Table II reports latency and round-trip counts averaged over the four in-MeSH terms. The direct method needs a single round-trip; the tree-number ascent needs one request to obtain the term's tree number(s) plus one request per parent position, which is the source of its higher cost.

Table II. Efficiency on the four terms registered in MeSH (averaged; warm = steady-state).

Method	HTTP round-trips	Cold (ms)	Warm mean (ms)	Warm median (ms)	Warm p95 (ms)
Direct (broaderDescriptor)	1	552	545	543	556
Tree-number ascent	3–5 (avg. 3.5)	2202	2182	2178	2217

Both methods returned identical, correct classes for all four terms (e.g., Troponin → Microfilament Proteins; Multiprotein Complexes; Muscle Proteins), so the comparison is on cost for the same output. The direct cold figure excludes the very first request of the run, a one-time connection warm-up of ≈2.9 s.

The direct strategy returns the same classification roughly four times faster (warm median 543 ms vs. 2178 ms) and with far fewer requests (1 vs. 3–5 round-trips). Notably, the per-request latency is similar for both (≈0.5–0.6 s); the entire difference is driven by the *number of HTTP round-trips* the tree-number ascent requires — one to read the tree number(s), then one per parent position. This confirms the design choice of Section 4.2: collapsing the classification into a single targeted query avoids the cumulative round-trip latency of walking the hierarchy level by level. It matters precisely because querying public SPARQL endpoints is costly [20].

6.3 Proof Of Concept: The Lcs Mechanism On Real Mesh Data

We first validate the mechanism on *proof-of-concept* cases, built by pairing genuine MeSH descriptors drawn from the *Myocardial Ischemia* branch (C14.280.647). Each pair stands for a compound expression (e.g., "coronary occlusion thrombosis") that is not registered as a single MeSH descriptor, so the direct broaderDescriptor method returns nothing; the LCS method, by contrast, assigns a class. The LCS is computed as the longest common prefix of the components' full tree numbers and was verified against the live MeSH endpoint (querying the prefix returns the named descriptor). The ten pairs in Table III are ordered from the most specific common ancestor to the most generic, to make the depth behaviour visible.

Table III. The LCS mechanism on descriptor pairs from the Myocardial Ischemia branch (verified against the live endpoint). Each pair represents an out-of-MeSH compound; the assigned class is the descriptor at the longest common prefix of the two tree numbers.

Component A (tree number)	Component B (tree number)	Common prefix (LCS)	Assigned class

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Coronary Restenosis (C14.280.647.250.285.200)	Coronary Stenosis (C14.280.647.250.285)	C14.280.647.250.285	Coronary Stenosis
Coronary Occlusion (C14.280.647.250.272)	Coronary Thrombosis (C14.280.647.250.290)	C14.280.647.250	Coronary Disease
Coronary Aneurysm (C14.280.647.250.250)	Coronary Vasospasm (C14.280.647.250.295)	C14.280.647.250	Coronary Disease
Coronary Artery Disease (C14.280.647.250.260)	Coronary Stenosis (C14.280.647.250.285)	C14.280.647.250	Coronary Disease
Coronary Occlusion (C14.280.647.250.272)	Coronary-Subclavian Steal Syndrome (C14.280.647.250.647)	C14.280.647.250	Coronary Disease
Anterior Wall Myocardial Infarction (C14.280.647.500.093)	Inferior Wall Myocardial Infarction (C14.280.647.500.187)	C14.280.647.500	Myocardial Infarction
ST Elevation Myocardial Infarction (C14.280.647.500.875)	Non-ST Elevated Myocardial Infarction (C14.280.647.500.469)	C14.280.647.500	Myocardial Infarction
Angina, Stable (C14.280.647.187.362)	Microvascular Angina (C14.280.647.187.575)	C14.280.647.187	Angina Pectoris
Coronary Stenosis (C14.280.647.250.285)	Anterior Wall Myocardial Infarction (C14.280.647.500.093)	C14.280.647	Myocardial Ischemia
Angina, Stable (C14.280.647.187.362)	Coronary Aneurysm (C14.280.647.250.250)	C14.280.647	Myocardial Ischemia

The table makes the key property visible: the assigned class climbs from specific to generic as the two components grow farther apart in the hierarchy. Components in the same sub-family share a deep prefix and yield a specific class (*Coronary Stenosis*, *Coronary Disease*, *Myocardial Infarction*, *Angina Pectoris*); components in different sub-families share only the branch prefix and yield the generic *Myocardial Ischemia*. The first row also shows that when one component is an ancestor of the other, the LCS is that ancestor itself. The method is therefore *depth-sensitive*: it returns the most specific common ancestor available, not a fixed shortcut toward the root. These cases confirm the pipeline (component lookup → tree numbers → common prefix → ancestor descriptor → class) produces correct, medically sensible classes for compounds the direct method cannot resolve.

6.4 Coverage On A Corpus Sample

The proof-of-concept cases above are *constructed* from descriptors known to share a branch; they validate the mechanism but are not drawn from text. The headline corpus experiment runs the pipeline on the eleven compound terms extracted from medical texts (Table I) and measures how many the LCS route recovers. On this sample the LCS method recovers 0 of 11 terms (0 %), as does the direct method. Error analysis shows why: these expressions are free-text clinical descriptions whose constituents are not MeSH descriptors (e.g., *circumflex*, *systolic*, *concentric*, *atheromatous*), so the decomposition never locates the two components needed to compute a common ancestor. This is not a failure of the mechanism — Section 6.3 demonstrates it on genuine MeSH-concept pairs — but a property of

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

this corpus: it contains clinical descriptions rather than compositions of indexed concepts. The result delimits the method's scope honestly: decomposition-plus-LCS recovers compounds built from indexed concepts, not arbitrary free-text phrases, and a richer terminological corpus (or normalization and entry-term matching, Section 7) would be required to raise corpus-level recovery.

6.5 Scaled Validation Across Mesh Domains

The corpus result above shows where the method does *not* apply; this section shows, at scale, where it does. We built a constructed benchmark by enumerating, within each of four large MeSH branches, every pair of leaf descriptors, and assigning each pair the descriptor at the longest common prefix of the two tree numbers (the LCS). Each pair stands for a compound expression made of two genuine MeSH concepts. The four branches span distinct clinical domains: cardiology (*Myocardial Ischemia*, C14.280.647), oncology (*Neoplasms by Site*, C04.588), pulmonology (*Respiratory Tract Diseases*, C08), and neurology (*Nervous System Diseases*, C10).

Table IV. Scaled validation of the LCS mechanism on intra-domain descriptor pairs from four MeSH branches.

Domain (MeSH branch)	Leaf descriptors	Pairs	LCS resolved
Cardiology (C14.280.647)	19	171	171 (100%)
Oncology (C04.588)	32	496	496 (100%)
Pulmonology (C08)	69	2344	2344 (100%)
Neurology (C10)	34	559	559 (100%)
Total	**154**	**3570**	**3570 (100%)**

Across the four domains, the LCS method assigns a class to 3570 of 3570 intra-domain pairs (100%). As a correctness control, we also formed all cross-domain pairs (one concept drawn from each of two different branches); the method correctly returns *no common ancestor* for 8207 of 8207 of them (100%), confirming that it does not fabricate spurious classes for unrelated concepts. The depth behaviour of Section 6.3 holds at scale: pairs from different sub-families resolve to the generic branch ancestor, while pairs from the same sub-family resolve to a specific descriptor (e.g., *Coronary Disease*, *Breast Neoplasms*, *Lung Diseases*, *Autoimmune Diseases of the Nervous System*).

This benchmark is *constructed*, not corpus-derived: it validates that the mechanism scales and behaves correctly across domains, not that such compounds are frequent in clinical text (Section 6.4 shows they are not). For the larger branches a subset of descriptors was sampled, which does not affect the outcome since LCS resolution within a branch is exact by construction. Taken together, the three experiments delimit the method precisely: 0/11 on free-text clinical descriptions, 3570/3570 on compounds of indexed concepts across four domains, and 8207/8207 correct abstentions on cross-domain pairs.

6.6 Direct-Method Resolution On A Real Corpus (Ncbi Disease)

The case studies of Section 5 used four hand-picked terms; here we evaluate the direct method's first stage — resolving a term string to its MeSH concept — at scale, on the NCBI Disease corpus, which annotates disease mentions in PubMed abstracts with gold MeSH concept identifiers [21]. We use the 328 unique test-set mentions that carry a single MeSH descriptor id. The corpus is a gold standard for *which concept a mention is*, so we measure resolution coverage and accuracy, and — using the gold id's broaderDescriptor as a silver reference — class agreement.

Exact matching on the descriptor *preferred* label (the original lookup) resolves only 1.5% of mentions, because real text uses lower case, synonyms, and abbreviations rather than canonical headings. We therefore add three lightweight strategies: matching against all of a descriptor's term labels (synonyms / entry terms) instead of only its preferred

Journal of Information Systems Engineering and Management
2026, 11(3s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

label; a small set of case variants; and expansion of in-abstract abbreviations via the Schwartz–Hearst algorithm [22]. Table V reports the effect.

Table V. Direct-method resolution on the NCBI Disease test set (328 unique disease mentions with gold MeSH ids).

Matching strategy	Resolution coverage	Lookup accuracy (vs gold id)
Exact preferred label (baseline)	5 / 328 (1.5%)	—
Enhanced (synonyms + case + abbreviation)	160 / 328 (48.8%)	149 / 328 (45.4%)

The enhanced matching raises coverage from 1.5% to 48.8% — a roughly 32-fold gain (+155 mentions) — and 45.4% of all mentions are resolved to the *correct* gold concept, approaching the 63.7% F-measure reported for knowledge-based normalization on this corpus [21] with a substantially simpler method. Once a mention is resolved, the classification step is reliable: the assigned broaderDescriptor class agrees with the gold concept's class for 89.4% of resolved mentions (143/160), and 95.1% of gold concepts have a parent available to assign. The bottleneck is therefore *resolution*, not classification, and synonym/abbreviation matching addresses most of it.

The residual gap (mentions still unresolved) stems from composite-concept annotations, genuine lexical divergence (e.g., "cancer" vs. "neoplasms" where no entry term bridges them), and rare orthographic variants; closing it calls for approximate matching (edit-distance or embedding-based) against an enriched dictionary such as MEDIC, which we leave to future work.

7. DISCUSSION AND LIMITATIONS

Three factors bound these results. First, the bottleneck is resolution, not classification: exact preferred-label matching resolves only 1.5% of real disease mentions, and although synonym, case, and abbreviation matching raises this to 48.8% (Section 6.6), closing the remaining gap requires approximate (edit-distance or embedding-based) matching against an enriched dictionary such as MEDIC. Second, latency depends on a live public endpoint and varies with network conditions and server load; we mitigate this by separating cold and warm runs and reporting percentiles. Third, the assigned class is verified against the MeSH browser and, on the corpus, against the gold concept's own broader descriptor as a silver standard, rather than an independent clinician-annotated gold standard for the class itself. The LCS route also depends on the chosen decomposition strategy (longest word n-grams present in MeSH); alternative strategies (head-noun, all n-grams) may change coverage.

8. CONCLUSION AND FUTURE WORK

We presented a Linked-Data approach that classifies compound candidate medical terms by their most specific broader descriptor(s) in MeSH RDF, retrieved through a single targeted SPARQL query, and an extension that classifies unseen compounds via decomposition and the least common subsumer over tree numbers. Measured against the live MeSH endpoint, the direct query is about four times faster than a tree-number-walking alternative for identical output, and the LCS extension classifies 3570/3570 intra-domain concept pairs across four MeSH domains while correctly abstaining on 8207/8207 cross-domain pairs; on a free-text clinical corpus it recovers none of eleven terms, which delimits its scope to compounds of indexed concepts. On the NCBI Disease corpus, enhancing the lookup with synonyms, case variants, and abbreviation expansion raises resolution coverage from 1.5% to 48.8% (45.4% to the correct concept), while classification stays reliable once a term is resolved (89.4% class agreement). Future work will (i) close the residual resolution gap with approximate matching against an enriched dictionary such as MEDIC, (ii) compare against neural normalizers such as DNorm [23], BioSyn [25], and SapBERT [26] on the same benchmark, and (iii) build a clinician-annotated gold standard for the assigned classes.

REFERENCES

[1] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," Scientific American, vol. 284, no. 5, pp. 34–

43, 2001.http://web.dfc.unibo.it/buzzetti/IUcorso2006-07/materiali/bl-engl.pdf

[2] C. Bizer, T. Heath, and T. Berners-Lee, "Linked Data – The Story So Far," International Journal on Semantic Web and Information Systems, vol. 5, no. 3, pp. 1–22, 2009. https://dl.acm.org/doi/abs/10.1145/3591366.3591378

[3] T. Berners-Lee, "Linked Data," W3C Design Issues, 2006. https://www.w3.org/DesignIssues/LinkedData.html

[4] O. Bodenreider and R. Stevens, "Bio-ontologies: current trends and future directions," Briefings in Bioinformatics, vol. 7, no. 3, pp. 256–274, 2006. https://doi.org/10.1093/bib/bbl027

[5] W. Ghemmaz, "Introduction de la technologie Linked Open Data (LOD) dans l'intégration des sources de données : une approche de matching d'instances basée sur le lien ViewSameAs," Ph.D. dissertation, University Constantine 2 – Abdelhamid Mehri, 2019.

[6] C. E. Lipscomb, "Medical Subject Headings (MeSH)," Bulletin of the Medical Library Association, vol. 88, no. 3, pp. 265–266, 2000.https://pmc.ncbi.nlm.nih.gov/articles/PMC35238/

[7] F. B. Rogers, "Medical Subject Headings," Bulletin of the Medical Library Association, vol. 51, pp. 114–116, 1963. https://europepmc.org/api/getPdf?pmcid=PMC197951

[8] U.S. National Library of Medicine, "Medical Subject Headings RDF (dataset)." https://id.nlm.nih.gov/mesh/

[9] K. T. Frantzi, S. Ananiadou, and H. Mima, "Automatic recognition of multi-word terms: the C-value/NC-value method," International Journal on Digital Libraries, vol. 3, no. 2, pp. 115–130, 2000. https://link.springer.com/article/10.1007/s007999900023

[10] M. T. Cabré, Terminology: Theory, Methods and Applications. Amsterdam: John Benjamins, 1999. Vol 1.

[11] O. Bodenreider, "The Unified Medical Language System (UMLS): integrating biomedical terminology," Nucleic Acids Research, vol. 32, no. suppl_1, pp. D267–D270, 2004. https://doi.org/10.1093/nar/gkh061

A. R. Aronson and F.-M. Lang, "An overview of MetaMap: historical perspective and recent advances," Journal of the American Medical Informatics Association, vol. 17, no. 3, pp. 229–236, 2010.https://doi.org/10.1136/jamia.2009.002733

[12] T. Heath and C. Bizer, Linked Data: Evolving the Web into a Global Data Space. San Rafael, CA: Morgan & Claypool, 2011.

[13] F. Belleau, M.-A. Nolin, N. Tourigny, P. Rigault, and J. Morissette, "Bio2RDF: towards a mashup to build bioinformatics knowledge systems," Journal of Biomedical Informatics, vol. 41, no. 5, pp. 706–716, 2008. https://www.sciencedirect.com/science/article/pii/S1532046408000415

[14] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," in Proc. 14th International Joint Conference on Artificial Intelligence (IJCAI-95), 1995, pp. 448–453. https://arxiv.org/abs/cmp-lg/9511007

[15] Z. Wu and M. Palmer, "Verb semantics and lexical selection," in Proc. 32nd Annual Meeting of the Association for Computational Linguistics (ACL-94), 1994, pp. 133–138. https://aclanthology.org/P94-1019.pdf

[16] T. Pedersen, S. V. S. Pakhomov, S. Patwardhan, and C. G. Chute, "Measures of semantic similarity and relatedness in the biomedical domain," Journal of Biomedical Informatics, vol. 40, no. 3, pp. 288–299, 2007. https://www.sciencedirect.com/science/article/pii/S1532046406000645

[17] E. Prud'hommeaux and A. Seaborne, "SPARQL Query Language for RDF," W3C Recommendation, 15 Jan. 2008. https://www.w3.org/TR/rdf-sparql-query/

[18] J. Pérez, M. Arenas, and C. Gutierrez, "Semantics and complexity of SPARQL," ACM Transactions on Database Systems, vol. 34, no. 3, art. 16, pp. 1–45, 2009. https://doi.org/10.1145/1567274.15672

[19] C. Buil-Aranda, A. Hogan, J. Umbrich, and P.-Y. Vandenbussche, "SPARQL Web-Querying Infrastructure: Ready for Action?," in The Semantic Web – ISWC 2013, LNCS 8219, Springer, pp. 277–293. https://doi.org/10.1007/978-3-642-41338-4_18

[20] R. I. Doğan, R. Leaman, and Z. Lu, "NCBI disease corpus: a resource for disease name recognition and concept normalization," Journal of Biomedical Informatics, vol. 47, pp. 1–10, 2014. https://www.sciencedirect.com/science/article/pii/S1532046413001974

A. S. Schwartz and M. A. Hearst, "A simple algorithm for identifying abbreviation definitions in biomedical text," in Proc. Pacific Symposium on Biocomputing (PSB), 2003, pp. 451–462. https://doi.org/10.1142/9789812776303_0042

[21] R. Leaman, R. I. Doğan, and Z. Lu, "DNorm: disease name normalization with pairwise learning to rank," Bioinformatics, vol. 29, no. 22, pp. 2909–2917, 2013. https://doi.org/10.1093/bioinformatics/btt474

[22] R. Leaman and Z. Lu, "TaggerOne: joint named entity recognition and normalization with semi-Markov models," Bioinformatics, vol. 32, no. 18, pp. 2839–2846, 2016. https://doi.org/10.1093/bioinformatics/btw343

[23] M. Sung, H. Jeon, J. Lee, and J. Kang, "Biomedical entity representations with synonym marginalization," in Proc. 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020, pp. 3641–3650. https://aclanthology.org/2020.acl-main.335/

[24] F. Liu, E. Shareghi, Z. Meng, M. Basaldella, and N. Collier, "Self-alignment pretraining for biomedical entity representations," in Proc. 2021 Conf. of the North American Chapter of the ACL (NAACL-HLT), 2021, pp. 4228–4238. https://aclanthology.org/2021.naacl-main.334/