Hyperbolic Embedding of Subsumption Groups in Large-Scale Biomedical Ontologies with Containment Cone Constraints

Shervin Mehryar[1], Michel Dumontier[1]

[1]Maastricht University, Paul-Henri Spaaklaan 1, 6229 GT Maastricht, Netherlands

Abstract

Embedding methods are a key enabler of recent advances in language model and augmented retrieval systems. For biomedical ontologies in particular, embedding data in a geometric space with non-zero curvature can capture the structure of hierarchies more robustly. In this work, a hyperbolic embedding method is proposed which is equipped with containment constraints for modeling subsumption groups. On synthetically generated directed graphs, it is found that arbitrarily low distortions can be achieved. Through experimentation on a real-world biomedical ontology, it is shown that our method satisfies a larger portion of containment properties compared to existing models.

Keywords

Biomedical Ontologies, Clinical Applications, Hyperbolic Embedding, SNOMED CT

1. Introduction

Representation learning for biomedical data via embedding methods has been a key enabler of recent advances in health informatics [1]. In particular, graph-based embedding models exhibit desirable geometric properties in order to produce features across various tasks, such as link prediction over medical a knowledge base [2], subgraph retrieval for logical query answering [3], and disease classification in healthcare systems [4]. A common approach to feature learning is to embed symbolic objects in low-dimensional vector spaces (e.g. Euclidean) in order to capture semantic information by minimizing a similarity score on correlated items. Euclidean models however fail to capture latent hierarchical structures and heterogeneous patterns inherent in many biomedical sources used in healthcare applications [4].

Recently, hyperbolic spaces have been shown to produce high quality representations for entailment relations over tree-like and directed acyclic graphs (DAG). Compared with Euclidean methods (having zero curvature), the latent features from hyperbolic space better reflects the geometric properties of hierarchically structured data (trees, DAGs, etc). The hyperbolic version of the convolution graph neural networks [5] for instance has supplanted the state of the art performance among embedding models due to the power in representing hierarchical patterns. In fact, with the exponential growth in size of deep hierarchical graph structures, embeddings end up undesirably close to each other in bounded regions of the Euclidean space. A key insight from hyperbolic geometry is that tree-like structures (with large branching factors) can be embedded in hyperbolic spaces (e.g. Poincare ball) with arbitrarily low distortion [6].

Despite the potential for representing a large biomedical knowledge base in the hyperbolic space, the particular embedding method must overcome a number of technical challenges depending on the application in which its used. In the space of manifolds with negative curvature (e.g. Poincare ball) for instance, points tend to collapse inward (i.e. the origin) or outwardly (on the

SeWebMeDA-2026: 9th International Workshop on Semantic Web Solutions for Large-scale Biomedical Data Analytics, May 10, 2026, Dubrovnik , Croatia

shervin.mehryar@maastrichtuniversity.nl (S. Mehryar); michel.dumontier@maastrichtuniversity.nl (M. Dumontier)

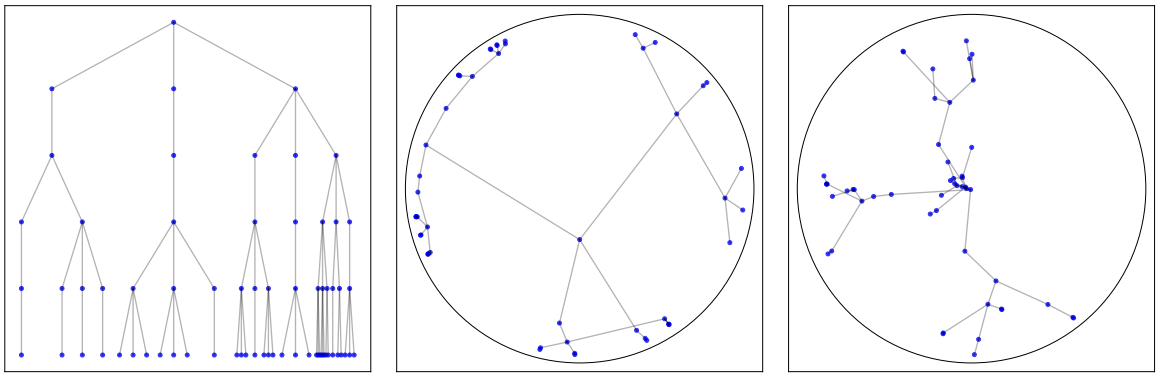

Figure 1: Embedding of a directed acyclic graph (DAG) with branching factor 3 and maximum depth 5: original discrete-space graph (left), hyperbolic space embeddings exhibiting boarder collapse (middle), and model equipped with distance regularization and cone constraints exhibiting containment violations (right).

boarder), if containment constraints are not imposed [7]. A second and equally important issue has to do with the fact that the hyperbolic distance alone is incapable of encoding asymmetric relations needed for certain entailments [6]. Specifically in the case of ontological graphs (with directed entailments), the partial ordering[1] of hierarchical paths may not be properly modeled with respect to the subsumption relationship [8]. In Figure 1, we demonstrate these artifacts by means of an example DAG embedded in the 2-dimensional hyperbolic space with curvature -1.0 (i.e. Poincare Ball).

In order to address these issues, in this work we propose an embedding approach for hierarchically structured graphs that represents input nodes with minimal distortion with respect to subsumption entailment. To address the issue of boarder collapse, we introduce radial depth regularization with respect to the subsumption relation. To address the issue of containment, we introduce hyperbolic cone entailment with half-aperture and rotation restrictions. In our experiments we show that on synthetically generated graphs with varying configurations, arbitrarily low distortion can be achieved. As a use case, we apply and report on the performance of the proposed method in a real-world setting using a large biomedical knowledge base, namely SNOMED CT[2].

2. Methodology

The proposed system learns hierarchical concept embeddings in the Poincare ball, leveraging hyperbolic geometry to naturally represent tree-like structures, as shown in Figure 2. Training combines three complementary objectives: a distance-based contrastive loss to preserve parent–child proximity, a radial ranking constraint to enforce depth ordering (parents closer to the origin than children), and a cone-based angular constraint to model subtree containment. Together, these components encode hierarchy through distance, radius, and angle, yielding embeddings that faithfully capture subsumption structure.

2.1. Poincare Contrastive Loss

We learn node embeddings in the d-dimensional Poincare ball of constant negative curvature ($c = -1$), $\mathbb{B}_c^d = \{x \in \mathbb{R}^d : \|x\| < 1\}$. This Riemannian manifold is particularly well-suited for hierarchical data: its volume grows exponentially toward the boundary, allowing tree-like structures (with exponentially increasing branching) to be embedded with low distortion compared to Euclidean space. The hyperbolic

[1] A partial ordering here means that nodes are ordered by the ancestor–descendant relationship.
[2] https://www.snomed.org/

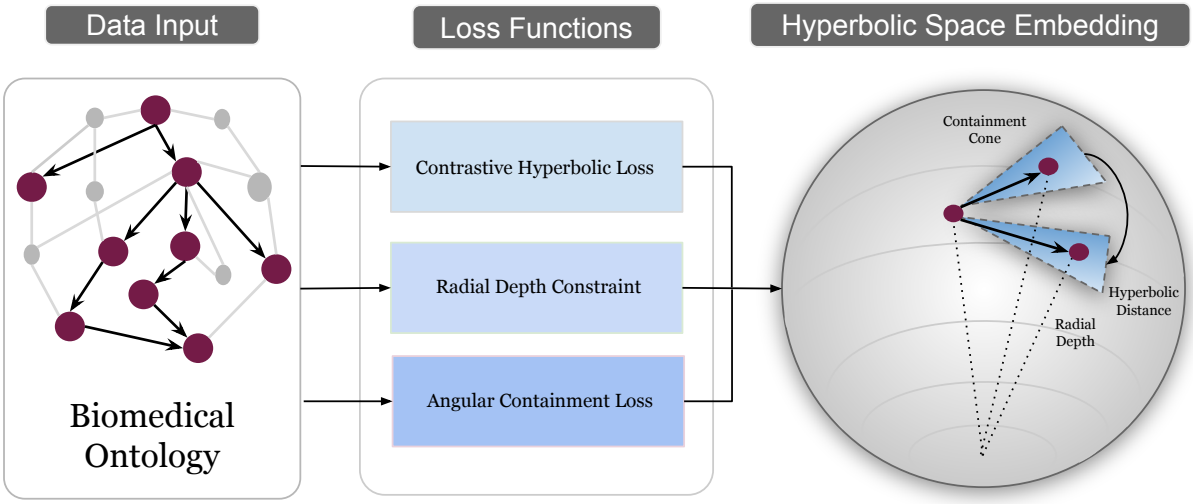

Figure 2: The proposed multi-component framework for Hyperbolic Embedding of Large Biomedical Ontologies.

(Riemannian) distance between any two points $u, v \in \mathbb{B}_c^d$ is given by:

$$d_{\mathbb{B}_c}(u,v) = \operatorname{acosh}\left(1 + 2\frac{\|u-v\|^2}{(1-\|u\|^2)(1-\|v\|^2)}\right), \tag{1}$$

where acosh is the inverse hyperbolic cosine function. This distance metric expands rapidly near the boundary, naturally separating deep hierarchical levels while keeping high-level nodes compact near the origin. For a positive parent–child pair (p, x) and N' negative samples $\{x_i'\}_{i=1}^{N'}$ (drawn via random or hard negative mining from the graph), we convert distances into unnormalized similarities via exponential kernels: $s = \exp\big(-d_{\mathbb{B}_c}(p,x)\big)$ and $s_i' = \exp\big(-d_{\mathbb{B}_c}(p,x_i')\big)$. The probability of selecting the true child among all candidates is given by the softmax function: $p(x) = s/\big(s + \sum_{i=1}^{N'} s_i'\big)$. The corresponding contrastive loss is given by the negative log-likelihood formulation [7]:

$$\mathscr{L}_{\mathrm{h}} = -\mathbb{E}_x\big[\log p(x)\big], \tag{2}$$

where $\mathbb{E}$ is the expectation function with respect to the probability measure on x. This objective encourages exponentially stronger similarity (i.e., markedly smaller hyperbolic distance) between each parent and its true child than to negatives. Because the exponential amplifies small distance differences and the underlying metric is already non-Euclidean, the loss produces smooth, stable gradients that are especially effective for optimization in curved hyperbolic space.

2.2. Radial Depth Regularization

A fundamental geometric property of the Poincare ball is that hierarchical depth is encoded along the *radial* direction: higher-level (more abstract) nodes naturally lie closer to the origin, while deeper (more specific) nodes approach the boundary $\|x\| \to 1$. To explicitly enforce this global ordering and prevent norm collapse, we introduce a radial ranking regularizer as follows. For each parent–child pair (p, x), let $r_p = \|p\|$ and $r_x = \|x\|$ denote the hyperbolic norms. We require $r_x \geq r_p + \delta$ (and set margin $\delta = 0.02$ in our experiments) and penalize violations using a hinge loss:

$$\mathscr{L}_{\mathrm{d}} = \mathbb{E}_{(p,x)}\Big[\max\big(0,\ r_p - r_x + \delta\big)\Big], \tag{3}$$

where max(.) is the maximum operator function. When the inequality holds, the loss is zero; otherwise, gradients push the child outward (or the parent inward) along approximately radial geodesics. This global structural prior complements the local relational objective $\mathscr{L}_{\mathrm{h}}$, prevents boarder collapse, and ensures consistent depth alignment across the entire hierarchy.

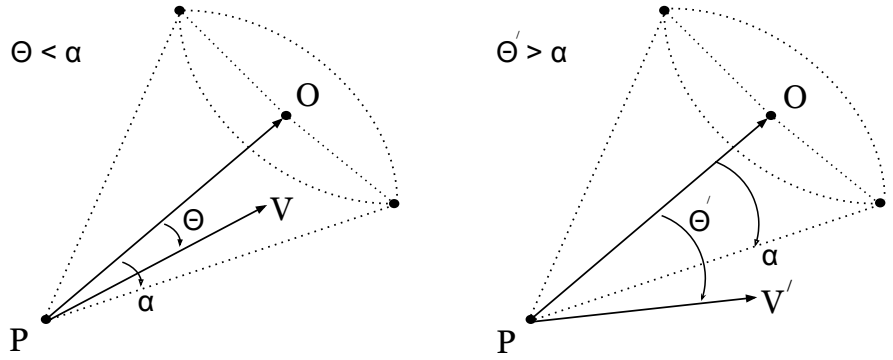

Figure 3: Containment Cone for a point P with a positive example V within its cone naturally supporting conjunctions ($\wedge$) and a negative sample V' outside its cone associated with negations ($\neg$).

2.3. Hyperbolic Cone Constraint

While distance and radial terms capture proximity and depth, they do not explicitly control *direction*. To enforce directional hierarchy (i.e., that children of the same parent lie in a coherent subtree while unrelated nodes are angularly separated), we impose a cone containment constraint in the tangent space at the origin. We first map the embeddings to the Euclidean tangent space at $\mathbf{0}$ using the logarithmic map: $\tilde{p} = \log_0(p)$, $\tilde{x} = \log_0(x)$, and $v = \tilde{x} - \tilde{p}$. The map $\log_0$ is an isometry, so angles computed here faithfully reflect hyperbolic directions.

As shown in Figure 3, the angle θ between the parent's radial vector $\tilde{p}$ and the relative direction v is calculated as:

$$\theta = \arccos\left(\frac{\langle \tilde{p},\, v\rangle}{\|\tilde{p}\|\,\|v\|}\right),$$

where this value is measured in radians. Positive pairs for a true descendant vector v of ascendant p require it lie inside the cone ($\theta_{pv} \leq \alpha$). This is enforced using a penalty loss given as: $\max\big(0,\ \theta_{pv} - \alpha\big)$, where $\alpha \in (0, \pi)$ is the aperture of the allowed cone. Note that this loss is zero for any true child within the cone of its parent. Conversely, negative pairs for each negative sample v' require the opposite ($\theta_{pv'} \geq \alpha$), which in turn gives rise to the penalty term: $\max\big(0,\ \alpha - \theta_{pv'}\big)$. Negative samples can be randomly generated. These however often provide weak signal given the hyperbolic distances since violations are rarely triggered. A better strategy is to utilize nodes without ascendant-descendant relation and from a different subsumption group in order to sharpen the cone structure and enforce partial order constraints. The final cone object loss is defined as [8]:

$$\mathscr{L}_{\text{cone}} = \mathbb{E}_{(p,v)}\big[\max\big(0,\ \theta_{pv} - \alpha\big)\big] + \mathbb{E}_{(p,v')}\big[\max\big(0,\ \alpha - \theta_{pv'}\big)\big], \tag{4}$$

where the expectation is with respect to all positive pairs in the first term and with respect to all negative pairs in the second term. These angular terms promote coherent subtree geometry: all descendants of a node cluster within its conical sector while branches remain well-separated, greatly improving interpretability and reducing overlap between unrelated parts of the hierarchy.

The combined learning objective is a weighted sum of the three complementary losses:

$$\mathscr{L}_{total} = \mathscr{L}_{\text{h}} + \beta.\mathscr{L}_{\text{d}} + \mathscr{L}_{\text{cone}},$$

where $\beta > 0$ is a small balancing coefficient (radial ordering is partly encouraged by the geometry itself, so β is typically modest). Together, these components enforce local relational proximity, global depth ordering, and directional subtree alignment, yielding high-quality, low-distortion hierarchical representations in hyperbolic space. The process for optimizing the total objective function using the above components is summarized in Algorithm 1.

Algorithm 1 Hyperbolic Embedding with Containment Cone Constraints

Require: Edge index E (parent → child), depth array $\mathscr{D}$ (depth of each node)
Ensure: Trained Poincare embeddings $\mathbf{X} \in \mathbb{B}_c^d$, Containment Score (%)

Initialize Poincare embeddings $\mathbf{X}$ uniformly at random
for iteration $t = 1$ to T **do**
  $\mathbf{E}_{\text{ordered}} \leftarrow \textsc{DepthOrderEdges}(E, \mathscr{D})$ ▷ top-down edge ordering
  $\mathscr{L}_{\text{h}}, \mathscr{L}_{\text{cone}}, \mathscr{L}_{\text{d}} \leftarrow 0$
  for each batch $\mathbf{B} \in \mathbf{E}_{\text{ordered}}$ **do**
    Compute: $\mathscr{L}_{\text{h}}$ using Relation (2)
    Compute: $\mathscr{L}_{\text{d}}$ using Relation (3)
    Compute: $\mathscr{L}_{\text{cone}}$ using Relation (4)
    $\mathscr{L}_{\text{total}} \leftarrow \mathscr{L}_{\text{h}} + \beta.\mathscr{L}_{\text{d}} + \mathscr{L}_{\text{cone}}$ ▷ hierarchy distortion
    Update: Riemannian Gradient Descent step with $\mathbf{X}$
  end for
  Compute: containment score using Relation (5) ▷ angular distortion
end for
return X

3. Experimentation

In this section we conduct experiments to showcase the different aspects of the proposed algorithm using two different settings. In the first setting, synthetic graphs are generated in order to train the model with different characteristics including branching factor, depth, and cone angle. In the second set of experiments, we train and apply the model to subsumption groups from a real-world biomedical ontology, namely SNOMED CT, and compare its performance with existing methods.

3.1. Synthetic Hierarchical Dataset Generation

We generate five synthetic hierarchical trees G_1–G_5 with progressively increasing depth to assess hyperbolic embedding performance under growing complexity. A stochastic rooted tree generator starts from a single root (depth 0) and expands level-by-level, assigning each non-leaf node a random number of children sampled uniformly from 1 to the maximum branching factor, yielding irregular acyclic hierarchies. Specifically, G_1 uses depth 5 and branching factor 3, G_2 depth 10 and branching factor 3, G_3 depth 15 and branching factor 3, G_4 depth 15 and branching factor 2 (to keep node count manageable while emphasizing extreme depth), and G_5 depth 20 and branching factor 2 as shown in Table 2. These configurations probe the effects of depth, branching interactions, and embedding scalability. We construct the edge index E from parent–child relations as a tensor and compute ground-truth depths $\mathscr{D}$ via breath first search from the root for quantitative evaluation.

We train the model under three settings—optimizing $\mathscr{L}_h$, $\mathscr{L}_{cone}$, or $\mathscr{L}_{total}$ (combined)—for 1000 epochs in 2-dimensional Poincare space using the Riemannian Adam optimizer with learning rate 0.01 and batch size 256. Parent–child edges are processed in depth-ordered batches obtained via breath first traversal from the root. The first setting optimizes solely the Poincare distance softmax loss, the cone setting minimizes exclusively the hyperbolic cone loss with a 60° ($\pi/3$) margin, and the combined setting jointly optimizes the Poincare distance softmax loss, cone loss, and radial ranking loss (weighted by $\beta = 0.5$). We monitor two metrics: hierarchical distortion, defined as the average per-batch loss and the angle distortions, defined as the number of angle violations with respect to the 60-degree margin (computed as the fraction of positive parent-child pairs whose tangent-space angle exceeds $\pi/3$). The results are summarized in Table 1.

Across the five synthetic hierarchical datasets G_1–G_5, spanning shallow trees with depth 5 to very

Table 1
Performance comparison on synthetic datasets G_1–G_5 using three different loss components, namely contrastive hyperbolic loss $\mathscr{L}_{\text{h}}$, cone loss $\mathscr{L}_{\text{cone}}$, and combined total loss $\mathscr{L}_{total}$. Results are reported as radial distance and angle distortions corresponding to values of each loss term (lower is better).

Objective Function	Distortion Metric	G_1	G_2	G_3	G_4	G_5
$\mathscr{L}_{\text{h}}$	Hierarchy	0.110	0.019	0.022	0.127	0.051
	Angle	0.220	0.130	0.800	0.343	0.812
$\mathscr{L}_{\text{cone}}$	Hierarchy	0.020	0.017	0.092	0.090	0.045
	Angle	0.110	0.010	0.000	0.191	0.062
$\mathscr{L}_{total}$	Hierarchy	0.010	0.015	0.070	0.063	0.034
	Angle	0.100	0.040	0.000	0.140	0.000

Table 2
Characteristics of the generated synthetic hierarchical datasets.

Dataset	Max Depth	Max Branch Factor	# Nodes
G_1	5	3	69
G_2	10	3	2,117
G_3	15	3	69,659
G_4	15	2	158
G_5	20	2	1,041

deep trees with depth 20, the combined objective consistently achieves the lowest hierarchical distance distortion and the fewest angle violations (relative to a 60° margin) compared to either the Poincare distance loss $\mathscr{L}_{\text{h}}$ or the cone loss $\mathscr{L}_{\text{cone}}$ alone. The pure Poincare approach exhibits the highest angle violations, particularly in deeper trees (e.g., 0.800 in G_3 and 0.812 in G_5), indicating limited directional alignment despite moderate distortion in some cases. Conversely, the cone-only objective effectively suppresses angle violations (reaching 0 in G_3 and minimal values elsewhere) but results in higher hierarchical distance distortion than the combined model for most datasets. Overall, the combined setting provides the best trade-off, achieving the lowest distortion across all graphs (e.g., 0.010 in G_1, 0.034 in G_5) while nearly eliminating angle violations (0 in G_3 and G_5, and substantially reduced in the remaining graphs), demonstrating that jointly optimizing hyperbolic proximity and directional cone constraints produces the most accurate and geometrically faithful hierarchical embeddings.

We visualize the embedded points trained with different angles of margin, including 15, 30, 60, and 90 degrees, as shown in Figure 4 within the 2-dimensional Poincare ball. It can be observed that with the increasing aperture angle, parent–child entailments experience more freedom to expand in the radial dimension as expected. With margin equal to 15 degrees for example, the cone structures are the narrowest and therefore enforce sharper containment constraints. Moreover, the distance regularization term helps prevent boarder collapse in all cases by keeping the distance between nodes at the top of the hierarchical structures smaller than the nodes near the boarder (e.g. leaf nodes).

3.2. Use Case: SNOMED CT

For a use case using a real-world biomedical ontology, in this section embeddings are learned based on various subsumptions groups from SNOMED CT. To construct a hierarchical graph from SNOMED CT for hyperbolic embedding with cone constraints, we wish to extract explicit subsumption (parent–child) relationships while incorporating logical operators and restriction constructs. SNOMED CT is represented as a directed acyclic graph, where hierarchical relations are expressed primarily through 'rdfs:subClassOf' axioms. However, some subclass axioms involve complex logical expressions such as 'owl:intersectionOf' and 'owl:Restriction', which result in representations that are not meaningful for

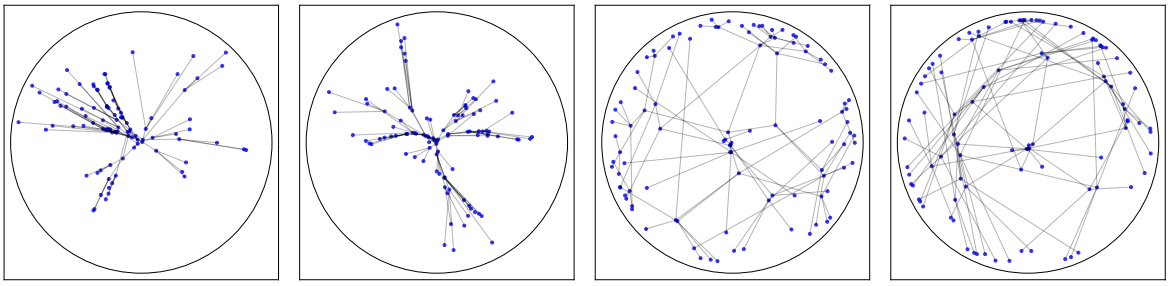

Figure 4: 2D Visualization of Poincare embeddings optimized with objective function $\mathscr{L}_{total}$ on different runs with increasing margin angle α, namely 15, 30, 60, and 90 degrees.

graph-based learning tasks if not handled correctly. We describe this process in the following.

In the OWL representation of SNOMED CT, subsumption can appear in simple or logically defined forms that must be normalized when constructing a pure taxonomy graph. A straightforward case is a direct assertion such as Ester $\sqsubseteq$ ChemicalSubstance, which can be directly converted into a parent–child edge. More complex cases involve intersection expressions, for example Pesticide $\sqsubseteq$ (Agrochemical biocide $\sqcap$ Industrial agent), meaning that *Pesticide* is a subclass of a conjunction; for taxonomy extraction this can be decomposed into the two edges Pesticide $\sqsubseteq$ Agrochemical biocide and Pesticide $\sqsubseteq$ Industrial agent. In logically defined concepts expressed via equivalence axioms, such as $C \equiv (X \sqcap \exists r.D)$, the class C is defined as the intersection of a named superclass X and a role restriction which can be realized using symbolic reasoning tools first. Thus, across all cases—direct subsumption, conjunction-based subsumption, and equivalence-based logical definitions—the transformation reduces OWL axioms to explicit subclass relations between named classes, producing a clean directed acyclic hierarchy suitable for graph representation. To account for these constructions, we consider three sets of input graphs after processing the raw SNOMED CT ontology:

G^1_{snomed}: The simplest and safest approach is to extract only direct rdfs:subClassOf relations where both subject and object are named SNOMED concept URIs. Any subclass relation whose parent is a blank node is ignored. This yields a clean asserted hierarchy consisting exclusively of explicit SNOMED entities and direct parent–child edges, which can be oriented top-down (parent $\longrightarrow$ child) for graph construction. This method avoids logical definition structures and ensures that the resulting graph contains no artificial nodes introduced by OWL encoding.

G^2_{snomed}: A slightly richer approach involves parsing owl:intersectionOf expressions and extracting any named SNOMED classes that appear within them, while still ignoring restriction nodes. In cases where owl:equivalentClass axioms define a concept as an intersection of a parent concept and additional role restrictions, only the named superclass can be retained as a taxonomic edge. This preserves meaningful subsumption information while discarding non-taxonomic logical definitions.

G^3_{snomed}: For maximal correctness, especially when working with SNOMED's fully defined concepts, our approach is to first classify the ontology using an OWL reasoner, namely HermiT[3], and then extract only inferred SubClassOf relations between named classes. This produces a fully resolved taxonomy without blank nodes or restriction structures and ensures that all implicit hierarchical relationships are captured. The resulting graph can then be safely converted into a representation consisting solely of explicit SNOMED concept nodes connected by top-down subsumption edges.

For each of these complexity levels, we then train hierarchical embeddings on three subsumption

[3]http://www.hermit-reasoner.com/

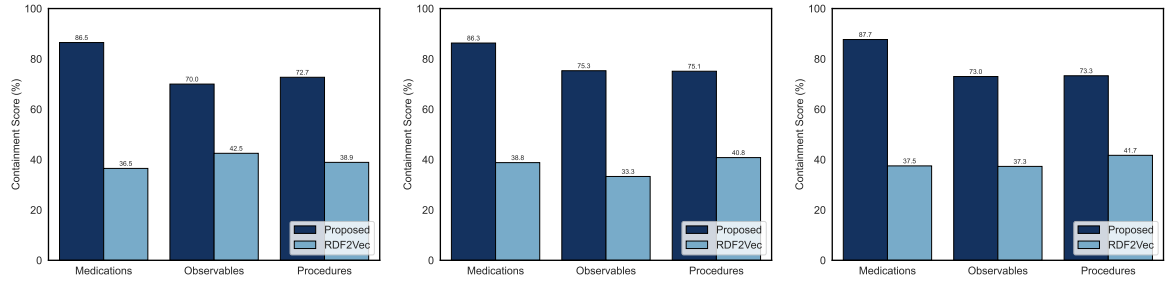

Figure 5: Comparison of Containment Scores (%) between the proposed embedding model and RDF2Vec on three subsumption groups from SNOMED CT: Medications, Observables, and Procedures over three graph settings: G^1_{snomed} with subsumption relations only (left), G^2_{snomed} with subsumption plus intersection relations (middle), G^3_{snomed} and with subsumption, intersection, and restriction properties (right).

groups from SNOMED CT using the proposed method with a batch size of 1024 and a learning rate of 0.01 with the Riemannian Adam optimizer. The model incorporates a cone-based loss with a margin angle of $\pi/6$ to enforce angular separation among sibling embeddings, alongside the standard Poincare distance loss. The network was trained for 500 epochs over the full set of edges from G^1_{snomed}, G^2_{snomed}, and G^3_{snomed}, optimizing a combination of objective functions to produce embeddings that respect both hyperbolic distances and hierarchical angular constraints.

In order to measure the integrity of the learned embeddings with respect to subsumption graphs in each case and to quantify how consistently cone constraints are satisfied we use the following metric. Let $\{\theta_i\}_{i=1}^N$ denote the set of angles formed between embedding vectors of antecedent–descendant pairs. Given a margin angle $\alpha > 0$ specifying the minimum separation required for meaningful cone containment, the **angle distortion** (AD) is defined as

$$\text{Angle Distortion } (\%) = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}\big[\theta_i \geq \alpha\big] \times 100, \tag{5}$$

where $\mathbb{1}[\cdot]$ is the indicator function and N is the total number of subsumption relationships. This metric is used to find the percentage of nodes that violate the containment constraints, i.e. $\theta_i \geq \alpha$ (see Figure 3). Equivalently, the **containment score** (CS) can be defined as the percentage of nodes that satisfy the containment constraints, i.e. $\theta_i \leq \alpha$, which we use in our experiments to report with. Values range from 0% (no sibling pair meets the threshold) to 100% (all pairs satisfy the angular separation). High values indicate that children are well-separated within their parent's cone, preserving the hierarchical structure, whereas low values reveal frequent angular violations and potential overlap among subtrees. Angle distortion thus provides an interpretable, local measure of geometric fidelity to the ontology's subsumption hierarchy. In terms of subsumption logic, high angle distortion corresponds to embeddings that faithfully preserve the hierarchical relations extracted from axioms. Conversely, low angle distortion signals that the geometric embedding may violate some of these hierarchical constraints, potentially obscuring the intended subsumption relations among child concepts.

To provide a Euclidean baseline for comparison with hyperbolic embeddings, we trained a RDF2Vec [9] model on random walks generated from each graph. For each concept, random walks were sampled by traversing outgoing subclass relations up to depth 2, producing sequences of concept identifiers analogous to sentences in natural language. To assess hierarchical consistency, we computed the containment angle between each parent–child pair using the standard cosine-based angular distance between their embedding vectors. An angle violation is recorded whenever the measured angle exceeds $\alpha = \pi/6$ similar to above, indicating insufficient geometric separation. The percentage angles satisfying this constraint is reported as the containment satisfaction score.

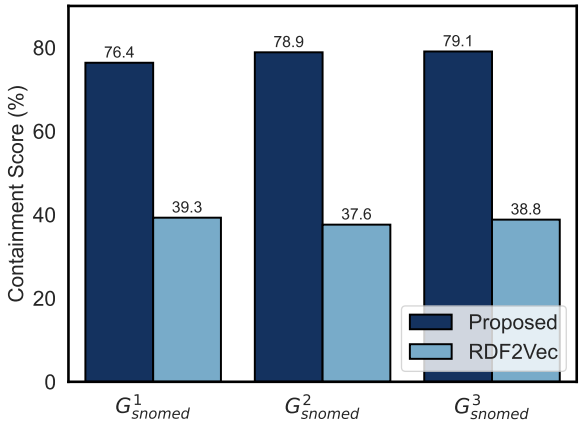

Figure 6: Comparison of Reasoner Effect on Containment Scores (%) between the proposed embedding model and RDF2Vec, overall subsumption groups. Graph settings are G^1_{snomed} with subsumption relations only, G^2_{snomed} with subsumption plus intersection relations, and G^3_{snomed} with subsumption, intersection, and restriction properties from SNOMED CT.

The main results of the experiment are shown in Figure 5, considering three subsumption groups similar to [4], namely Medications, Observables, and Procedures, with 57261, 221708, and 61229 nodes as well as 82522, 399982, and 111731 edges respectively. The detailed evaluation across the three SNOMED-derived graphs (G^1_{snomed}, G^2_{snomed}, and G^3_{snomed}) and the three semantic categories (Medications, Observables, and Procedures) demonstrates a clear and consistent advantage of the proposed hierarchical embedding method over RDF2Vec. In every graph variant and category, the proposed method achieves substantially higher containment scores, confirming its superior ability to encode the underlying subsumption hierarchy.

For the minimal asserted hierarchy G^1_{snomed}, which contains only direct named `rdfs:subClassOf` relations, the proposed method already attains strong performance: 86.5% (Medications), 70.0% (Observables), and 72.7% (Procedures). In contrast, RDF2Vec yields markedly lower scores of 36.5%, 42.5%, and 38.9%, respectively. Enriching the taxonomy in G^2_{snomed} by incorporating named classes extracted from `owl:intersectionOf` expressions leads to further modest gains for the proposed method: 86.3% (Medications), 75.3% (Observables), and 75.1% (Procedures). RDF2Vec shows only marginal variation (38.8%, 33.3%, and 40.8%). Finally, the fully reasoned taxonomy G^3_{snomed}, which includes all inferred `SubClassOf` relations after OWL reasoning, yields the highest scores for the proposed method: 87.7% (Medications), 73.0% (Observables), and 76.6% (Procedures). RDF2Vec remains largely unchanged, with scores of 37.5%, 37.3%, and 41.7%.

These results reflect the differing inductive biases of the two approaches. The proposed method explicitly models the directed, top-down hierarchical structure via cone-based geometric constraints in the Poincare ball, and thus directly benefits from each additional source of taxonomic information. RDF2Vec, relying on random walks and co-occurrence statistics, does not explicitly capture information related to containment of parent–child relationships. The proposed hierarchical embedding method in this case benefits from the explicit angular constraints imposed by the parent–child relationships, which guide the placement of child embeddings within well-separated cones and preserve the intended hierarchical structure in the embedding space.

In order to analyze the effect of reasoner and axiomatic complexities, we plot the containment scores in Figure 6 overall subsumption groups separated as G^1_{snomed}, G^2_{snomed}, and G^3_{snomed}. This trend reflects differences in graph construction. G^1_{snomed} contains only direct `rdfs:subClassOf` relations between named concepts, forming a minimal but clean asserted hierarchy. G^2_{snomed} adds named

classes extracted from `owl:intersectionOf` expressions in `subClassOf` or `equivalentClass` axioms, enriching the taxonomy. G^3_{snomed} incorporates all inferred `SubClassOf` relations obtained via full OWL reasoning (i.e. HermiT), producing the most complete and semantically accurate hierarchy. The results indicate that the proposed hierarchical embedding method consistently outperforms RDF2Vec on all three SNOMED-derived graphs, achieving higher average scores in every case. Notably, the mean performance of the proposed method increases from G^1_{snomed} to G^3_{snomed}, whereas RDF2Vec shows relatively stagnant across graphs.

As the graph becomes progressively richer, the proposed method can exploit the additional structural cues, particularly the refined parent–child relations, resulting in embeddings that better reflect the underlying subsumption hierarchy. RDF2Vec, relying on random walks to capture local neighborhood co-occurrences, does not explicitly model hierarchical directionality and therefore benefits little from the added edges and inferred relations. Consequently, its performance remains largely unchanged across the three graph variants. The proposed method effectively leverages increasingly complete hierarchical information, whereas RDF2Vec's walk-based representations are insensitive to the deeper subsumption structure present in G^2_{snomed} and G^3_{snomed}.

4. Conclusion

We introduce a Poincare embedding method for biomedical ontologies, capturing parent–child containment via distance, cone, and ranking losses. A depth-ordered batching strategy preserves hierarchical structure by processing shallower concepts first. Evaluated on three SNOMED CT subsumption groups, our method consistently outperforms RDF2Vec across Medications, Observables, and Procedures, achieving higher containment scores . The results demonstrate the advantage of explicitly modeling hierarchical geometry and exploiting both asserted and inferred subsumptions, producing embeddings that can better represent ontological structure and support hierarchical reasoning tasks.

Declaration on Generative AI

During the preparation of this work, the author(s) used Grok 4.1 as research and coding assistant in order to optimize code execution and to improve text readability (e.g. figure captions, gammer check) of the manuscript. Main ideas, experiment runs, and conclusions drawn are the work of the author(s) alone.

References

[1] S. Kotitsas, D. Pappas, I. Androutsopoulos, R. McDonald, M. Apidianaki, Embedding Biomedical Ontologies by Jointly Encoding Network Structure and Textual Node Descriptors, in: D. Demner-Fushman, K. B. Cohen, S. Ananiadou, J. Tsujii (Eds.), Proceedings of the 18th BioNLP Workshop and Shared Task, Association for Computational Linguistics, Florence, Italy, 2019, pp. 298–308. URL: https://aclanthology.org/W19-5032/. doi:10.18653/v1/W19-5032.

[2] S. Mehryar, Resolution-Alignment-Completion of Tabular Electronic Health Records via Meta-Path Generative Sampling, in: S. Chang, M. Hulsebos, Q. Liu, W. Chen, H. Sun (Eds.), Proceedings of the 4th Table Representation Learning Workshop, Association for Computational Linguistics, Vienna, Austria, 2025, pp. 200–207. URL: https://aclanthology.org/2025.trl-1.17/. doi:10.18653/v1/2025.trl-1.17.

[3] T. Xia, L. Ding, G. Wan, Y. Zhan, B. Du, D. Tao, Improving Complex Reasoning over Knowledge Graph with Logic-Aware Curriculum Tuning, Proceedings of the AAAI Conference on Artificial Intelligence 39 (2025) 12881–12889. URL: https://ojs.aaai.org/index.php/AAAI/article/view/33405. doi:10.1609/aaai.v39i12.33405.

[4] K. Agarwal, T. Eftimov, R. Addanki, S. Choudhury, S. Tamang, R. Rallo, Snomed2Vec: Random Walk and Poincaré Embeddings of a Clinical Knowledge Base for Healthcare Analytics, 2019. URL: http://arxiv.org/abs/1907.08650. doi:10.48550/arXiv.1907.08650, arXiv:1907.08650 [cs].

[5] I. Chami, Z. Ying, C. Ré, J. Leskovec, Hyperbolic Graph Convolutional Neural Networks, in: Advances in Neural Information Processing Systems, volume 32, Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/hash/0415740eaa4d9decbc8da001d3fd805f-Abstract.html.

[6] O.-E. Ganea, G. Bécigneul, T. Hofmann, Hyperbolic Entailment Cones for Learning Hierarchical Embeddings, 2018. URL: http://arxiv.org/abs/1804.01882. doi:10.48550/arXiv.1804.01882, arXiv:1804.01882 [cs].

[7] M. Nickel, D. Kiela, Poincaré Embeddings for Learning Hierarchical Representations, in: Advances in Neural Information Processing Systems, volume 30, Curran Associates, Inc., 2017. URL: https://papers.nips.cc/paper_files/paper/2017/hash/59dfa2df42d9e3d41f5b02bfc32229dd-Abstract.html.

[8] Y. Bai, Z. Ying, H. Ren, J. Leskovec, Modeling Heterogeneous Hierarchies with Relation-specific Hyperbolic Cones, in: Advances in Neural Information Processing Systems, volume 34, Curran Associates, Inc., 2021, pp. 12316–12327. URL: https://proceedings.neurips.cc/paper/2021/hash/662a2e96162905620397b19c9d249781-Abstract.html.

[9] P. Ristoski, H. Paulheim, RDF2Vec: RDF Graph Embeddings for Data Mining, in: P. Groth, E. Simperl, A. Gray, M. Sabou, M. Krötzsch, F. Lecue, F. Flöck, Y. Gil (Eds.), The Semantic Web – ISWC 2016, Springer International Publishing, Cham, 2016, pp. 498–514. doi:10.1007/978-3-319-46523-4_30.