Neural DTS: Integrating Hyperbolic Classifiers into Natural Language Inference Systems

Honoka Kobayashi and Hinari Daido * **and Daisuke Bekki**
Ochanomizu University
{kobayashi.honoka | daido.hinari | bekki}@is.ocha.ac.jp

Abstract

Dependent Type Semantics (DTS) provides a highly rigorous framework for natural language inference (NLI), yet its scalability is severely bottlenecked by the need for manually created world knowledge. To overcome this knowledge acquisition bottleneck, we present a novel neuro-symbolic NLI system that integrates Hyperbolic Entailment Cones for automated conceptual hierarchy discovery. By exploiting the geometric properties of hyperbolic space, our model efficiently learns lexical entailment relations and dynamically injects them as logical axioms during the DTS proof-search process. Evaluations on our constructed diagnostic dataset show that our hybrid approach broadens the coverage of complex lexical variations and paraphrases without manual engineering.

1 Introduction

Computational approaches to natural language semantics can be broadly categorized into two paradigms: logical methods based on formal linguistics and data-driven methods based on large-scale neural language models. The former offers the advantage of strictly formalizing the correspondence between sentence structure and meaning, thereby guaranteeing the validity of inference through formal proofs. However, these "symbolic" methods face significant practical constraints, as essential world knowledge for reasoning must be manually provided as axioms. Conversely, neural language models exhibit a highly flexible capacity for knowledge acquisition, yet their inference processes remain opaque, making it difficult to guarantee logical consistency or explainability. From the perspective of theoretical linguistics, it has long been posited that human language comprehension involves a distinct "language faculty" that cannot be reduced solely to "general cognitive processes". Simultaneously, it is undeniable that general cognitive processes—such as association, pattern recognition, and knowledge acquisition—contribute to language understanding. Therefore, a fundamental challenge in capturing the full scope of human language comprehension lies in formally defining the interface between these two domains: integrating the structural and principled theory of the language faculty (formal linguistics) with the flexible and continuous knowledge processing mechanisms (neural embedding models).

Neural DTS is a research project aimed at elucidating this interaction between the language faculty and general cognitive processes, both theoretically and computationally. It achieves this by centering on a proof-theoretic theory of meaning, Dependent Type Semantics (DTS) (Bekki and Mineshima, 2017), as a formal linguistic model, while incorporating externally acquired neural embeddings as a model of general cognition. Toward this goal, this paper proposes and implements a novel neuro-symbolic framework that dynamically supplements world knowledge regarding conceptual entailment—which is required during the DTS proof construction process—using a *hyperbolic* classifier. Specifically, we represent conceptual entailment, such as hyponymy-hypernymy relations, using Hyperbolic Entailment Cones (Ganea et al., 2018). We introduce a classifier as an "oracle" to determine the validity of these relations on-the-fly during the proof search. By injecting the outputs of this externally trained geometric embedding into the inference rules of DTS, we extend the system's reasoning capacity into a unified framework where the continuous nature of human knowledge acquisition and the discrete, logically explainable nature of formal reasoning are integrated into a coherent whole.

*This work was conducted independently and does not reflect the views or positions of Amazon.

Proceedings of the 6th Workshop on Natural Language Meets Logic and Machine Learning (NALOMA), pages 9–18

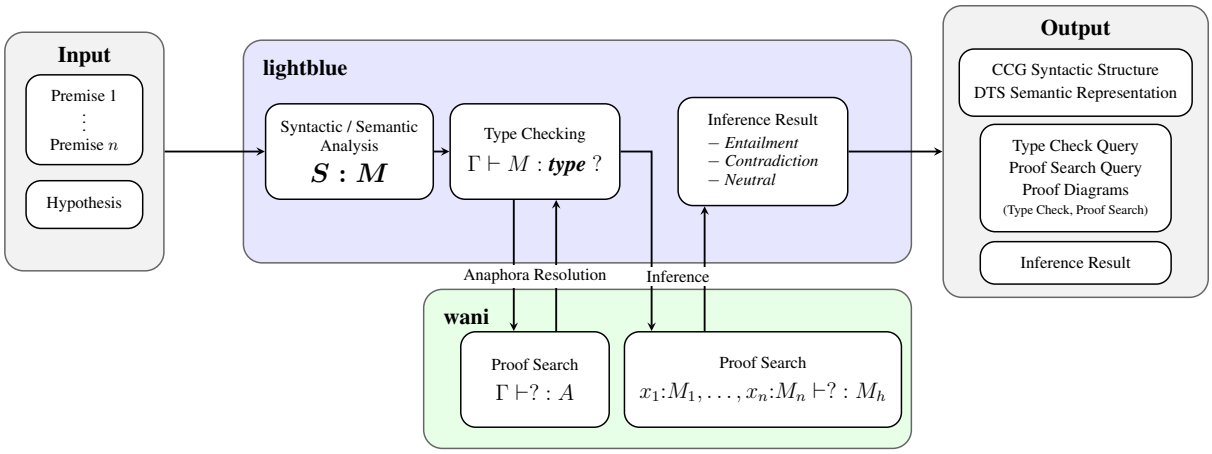

Figure 1: Natural Language Inference Pipeline

2 Previous Work

2.1 Natural Language Inference Pipeline

One of the latest advancements in linguistic-based natural language inference (NLI) is a system proposed by Tomita et al. (2025), illustrated in Figure 1. This system is composed of the CCG (Combinatory Categorial Grammar) (Steedman, 1996, 2000) syntactic parser `lightblue`[1] and the theorem prover `wani` (Daido and Bekki, 2020; Daido, 2022), which utilizes a sub-system of DTS.

When a set of premises and a hypothesis are provided as input, the process begins with `lightblue`, which performs syntactic parsing using CCG and semantic composition based on DTS. Subsequently, a type-checking process is conducted to verify whether the semantic representations of the sentences possess the sort "type". Following this, a proof search is executed by `wani`. In this stage, `wani` searches for a proof term that takes the semantic representations of the premises as assumptions and the semantic representation of the hypothesis as the conclusion. Based on the outcome of the proof search, the system yields an inference result: *Yes* if a proof term exists, *No* if a proof term exists for the negation of the hypothesis, and *Unknown* otherwise.

2.2 Neural DTS

DTS is a framework for formal semantics based on Dependent Type Theory (DTT) (Martin-Löf, 1984), capable of handling complex linguistic phenomena—such as negation, conditionals, quantification, anaphora, and presupposition—in a unified and rigorous manner. Since inference in DTS is formulated as the construction of proof diagrams within the type theory, the validity of the inference results is guaranteed. However, the cost of manually describing world knowledge as axioms, including lexical entailment relations, has been identified as a practical challenge.

To address this issue, Neural DTS (Bekki et al., 2021, 2022) was proposed as a framework to integrate DTS with neural embedding models. Bekki et al. (2021, 2022) presented the mathematical theory and learning algorithms for Neural DTS. Furthermore, Iinuma et al. (2023) worked on the implementation of Neural DTS, utilizing neural architectures such as Multi-Layer Perceptrons (MLP) and Neural Tensor Networks (Socher et al., 2013).

While their work demonstrated the practical feasibility of Neural DTS, there remains significant room for development. In particular, existing implementations relying on standard neural embedding models often operate in Euclidean spaces, which are inherently limited in their capacity to embed complex, hierarchical conceptual structures (e.g., hypernymy and hyponymy). This limitation underscores the need for a more geometrically appropriate representation of lexical entailment to further improve classification accuracy and reasoning efficiency, paving the way for the hyperbolic approach proposed in this paper.

3 The Hyperbolic Classifier

In this paper, we propose a novel approach for Neural DTS that integrates a hyperbolic classifier utilizing Hyperbolic Entailment Cones into a natural language inference system based on DTS.

[1] https://github.com/DaisukeBekki/lightblue

3.1 Advantages of Hyperbolic Embeddings

Lexical hypernymy-hyponymy relations possess a hierarchical tree structure. In such structures, the number of nodes increases exponentially with depth; however, the volume of Euclidean space grows only polynomially relative to its radius. Consequently, low-dimensional Euclidean spaces fail to arrange nodes appropriately, leading to significant structural distortion. In contrast, hyperbolic space possesses the property that its volume expands exponentially with respect to its radius. This allows for the embedding of hierarchical structures with extremely low distortion, even in low-dimensional spaces. In particular, Hyperbolic Entailment Cones, which we employ in this study, represent concepts as cones to geometrically define the asymmetric relationship of entailment. As reported by Ganea et al. (2018), this method achieves high performance in entailment detection tasks using resources such as WordNet (Bond et al., 2012).

3.2 Mathematical Definition of the Embedding Model

In Hyperbolic Entailment Cones, each concept u is embedded as a point $\boldsymbol{u}$ in the Poincaré ball $\mathbb{D}^n$, the open unit ball that serves as a model of n-dimensional hyperbolic space (throughout, we write the non-bold u for a concept and the bold $\boldsymbol{u}$ for its embedding vector). Although $\mathbb{D}^n$ is bounded in the ordinary Euclidean sense, its boundary lies infinitely far from the center O under the hyperbolic metric; consequently, moving a point outward toward the boundary corresponds to descending the taxonomy into ever more specific concepts. To avoid a singularity at the center, where the aperture defined below is undefined, an open ball $\mathcal{B}^n(O, \varepsilon)$ ($\varepsilon > 0$) around O is excluded. Intuitively, each concept u then owns a cone that fans out from its point $\boldsymbol{u}$ toward the boundary; every concept falling inside this cone is treated as a more specific instance (a hyponym) of u, so that *dog is-a animal* holds exactly when the point for *dog* lies inside the cone of *animal*. A learnable $K > 0$ parameterizes the cone's half-aperture $\psi(\boldsymbol{u})$, that is, how wide this cone opens:

$$\psi(\boldsymbol{u}) = \sin^{-1}\left(K\frac{1-\|\boldsymbol{u}\|^2}{\|\boldsymbol{u}\|}\right) \tag{1}$$

Geometrically, this aperture shrinks as u moves away from the origin, so that deep, specific concepts near the boundary own narrow cones, whereas general concepts near the center own wide cones that subsume many descendants. To ensure a valid arcsine domain, we impose the constraint:

$$K \le \frac{\varepsilon}{1-\varepsilon^2} \tag{2}$$

The geometric condition for a concept v (hyponym) to be entailed by a concept u (hypernym) is that the geodesic angle $\Xi(\boldsymbol{u}, \boldsymbol{v})$ between the two vectors must fall within the aperture of the cone formed by the hypernym u. Specifically, this is expressed by the following condition:

$$\Xi(\boldsymbol{u}, \boldsymbol{v}) \le \psi(\boldsymbol{u}) \tag{3}$$

Here the geodesic angle $\Xi(\boldsymbol{u}, \boldsymbol{v})$ measures the direction of v as seen from u relative to the cone's axis, which points radially outward from the center O: it is small when v lies straight ahead inside the cone and large when it points away. Concretely, writing $\angle Ouv$ for the angle at vertex u in the triangle Ouv, it is defined as:

$$\Xi(\boldsymbol{u}, \boldsymbol{v}) := \pi - \angle Ouv = \cos^{-1}\left(\frac{N}{D}\right) \tag{4}$$

where N and D are defined as:

$$N = \langle \boldsymbol{u}, \boldsymbol{v}\rangle(1+\|\boldsymbol{u}\|^2) - \|\boldsymbol{u}\|^2(1+\|\boldsymbol{v}\|^2)$$
$$D = \|\boldsymbol{u}\|\|\boldsymbol{u}-\boldsymbol{v}\|\sqrt{1+\|\boldsymbol{u}\|^2\|\boldsymbol{v}\|^2 - 2\langle \boldsymbol{u}, \boldsymbol{v}\rangle}$$

Here, $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ denote the Euclidean norm and the standard inner product, respectively. We reproduce this closed form from Ganea et al. (2018) for completeness; for the intuition that follows, it suffices to read $\Xi(\boldsymbol{u}, \boldsymbol{v})$ simply as the angular position of v relative to u's cone axis.

Furthermore, the energy function $E(\boldsymbol{u}, \boldsymbol{v})$ is defined as follows:

$$E(\boldsymbol{u}, \boldsymbol{v}) = \max(0, \Xi(\boldsymbol{u}, \boldsymbol{v}) - \psi(\boldsymbol{u})) \tag{5}$$

Intuitively, E measures how far v lies outside u's cone: it is zero whenever v is correctly contained, and grows with the size of the angular violation otherwise.

During model training, the following loss function $\mathcal{L}$ is minimized to reduce the energy for the positive sample set $\mathbb{P}$ while ensuring that the energy of the negative sample set $\mathbb{N}$ is at least a margin γ:

$$\mathcal{L} = \sum_{(\boldsymbol{u},\boldsymbol{v})\in\mathbb{P}} E(\boldsymbol{u},\boldsymbol{v}) + \sum_{(\boldsymbol{u}',\boldsymbol{v}')\in\mathbb{N}} \max(0, \gamma - E(\boldsymbol{u}',\boldsymbol{v}')) \quad (6)$$

In other words, training pulls genuine hyponym pairs inside their hypernym's cone, driving their energy toward zero, while pushing unrelated pairs at least a margin γ outside it.

Through this optimization, a geometrically consistent hierarchical structure is constructed within the hyperbolic space.

3.3 Data Preparation and Preprocessing

To develop a classifier capable of effectively evaluating and reasoning over world knowledge in Japanese, we construct our training dataset using the procedure outlined below, following the methodology of the previous study (Ganea et al., 2018). Our implementation is based on the official repositories provided by the authors.[2][3]

1. **Relation Extraction**: We extract hypernym-hyponym relations between nouns based on SynsetIDs from WordNet (Bond et al., 2012).

2. **Relation Classification**: By computing the transitive closure and transitive reduction of the extracted relations, we classify edges into "Basic edges," which represent direct parent-child relationships, and "Non-Basic edges," which represent other transitive relationships.

3. **Data Splitting**: The entire dataset is divided into training (Train), validation (Valid), and evaluation (Eval) sets. All Basic edges, which are essential for maintaining the hierarchical structure, are included in the training data. We then evaluate and compare model performance by varying the proportion of additional Non-Basic edges.

4. **Cross-lingual Alignment and Filtering**: To apply this learned hierarchical structure to our target Japanese vocabulary, we map the embeddings of the English SynsetIDs to their corresponding Japanese lemmas using the Open Multilingual WordNet (OMW) (Bond and Foster, 2013). By taking the intersection of the trained concepts and the available Japanese translations, we simultaneously assign the learned representations to the Japanese vocabulary and filter out any concepts that lack a Japanese translation. Importantly, this filtering is applied strictly as a post-processing step for the Eval set; the Train set retains all original English concepts to preserve the continuity and density of the underlying graph structure.

[2] https://github.com/dalab/hyperbolic_cones

[3] https://github.com/facebookresearch/poincare-embeddings

3.4 Model Training and Results

We trained models using three distinct methods—Order Embeddings (Vendrov et al., 2016), Poincaré Embeddings (Nickel and Kiela, 2017), and Hyperbolic Entailment Cones (Ganea et al., 2018)—and compared their performance. The parameters configured during training are detailed in Table 1.

Table 1: Parameter settings used during training

Item	OrderEmb	Poincaré	Hyp Cones
Dimension	5 / 10	5 / 10	5 / 10
LR	1×10^{-1}	1×10^{-4}	3×10^{-4}
Epochs	500	300	300
Optimizer	SGD	ExpMap	RSGD
Neg Samples	10	10	10
Margin (γ)	1.0	—	0.01
Init Model	None	Poincaré	Poincaré
Neg Sampling	parent	child	parent

The performance of these models, evaluated by their F1 scores on the binary entailment classification task, is summarized in Table 2 (5 dimensions) and Table 3 (10 dimensions). We assessed the performance by varying the proportion of Non-Basic edges appended to the training data.

Table 2: Experimental results (5 dimensions)

	Embedding Dimension = 5				
Model	0%	10%	25%	50%	90%
Order Emb	**38.0%**	72.7%	**78.3%**	83.3%	87.4%
Poincaré	28.6%	55.2%	73.8%	79.4%	82.4%
Hyp Cones	29.9%	**73.2%**	75.1%	**92.0%**	**93.3%**

Table 3: Experimental results (10 dimensions)

	Embedding Dimension = 10				
Model	0%	10%	25%	50%	90%
Order Emb	**46.0%**	72.4%	81.4%	87.0%	87.0%
Poincaré	30.2%	68.0%	82.3%	84.5%	86.3%
Hyp Cones	30.9%	**76.9%**	**87.9%**	**92.8%**	**93.9%**

From the results shown in Table 2 and Table 3, we observed a trend across all methods: the F1 score improved as the proportion of Non-Basic edges added to the training data increased. In particular, the high accuracy achieved by Hyperbolic Entailment Cones even in low-dimensional spaces (10 dimensions or fewer) indicates that the exponentially growing hierarchical structure of WordNet is successfully embedded with minimal distortion. These results are highly consistent with the findings of the previous study by Ganea et al. (2018), confirming that our training implementation accurately reproduces the geometric characteristics of each embedding method.

3.5 Standalone Oracle Evaluation

Building upon the models trained in the previous section, we implemented the hyperbolic classifier *oracle* in Haskell, the same language as the `wani` theorem prover, allowing it to be integrated natively into the prover for NLI. To evaluate its standalone performance, we exported the embeddings trained with 10 dimensions and 90% of the edges.

The Haskell-based *oracle* yielded an F1 score of 92.85% on the test set. Crucially, this result is directly comparable to the performance achieved during the Python-based training phase, demonstrating that the exported geometric parameters maintain their numerical stability and strict discriminative capability when loaded into a purely functional programming environment. This confirms that our Haskell implementation acts as a reliable and rigorous *oracle* for the formal proof search in DTS.

4 Integration into Natural Language Inference Systems

Next, we propose a framework for integrating the trained hyperbolic classifier into our NLI system.

4.1 Framework for Dynamic Axiom Addition

When the `wani` theorem prover receives the trained classifier as part of its input, it determines dynamically whether to invoke the hyperbolic *oracle* during its backward inference process. In `wani`, backward inference refers to the process of advancing the proof search by applying a subset of DTT rules to generate subgoals from the original premises and hypothesis. During this process, if the conclusion of a generated subgoal takes the form of a predicate (represented as a constant in DTT) applied to one or more terms—for example, $\mathbf{dog}(s, x)$—`wani` invokes the classifier. In this section, we assume that predicates represent properties of the given terms. Specifically, it searches the current premises for an entry that shares the same term but applies a different predicate, such as $\mathbf{puppy}(s, x)$. It then passes the premise's predicate **puppy** (as a hyponym candidate) and the subgoal's predicate **dog** (as a hypernym candidate) to the classifier. If the probability calculated by the classifier exceeds a threshold of 0.5, `wani` treats the semantic entailment $\mathbf{puppy} \sqsubseteq \mathbf{dog}$ as a valid logical axiom, successfully resolving that specific subgoal.

4.2 Entailment Determination by the Oracle Classifier

The *oracle* determines the entailment probability between surface-level words w_1 and w_2 by mapping them to their corresponding synset embeddings. To handle polysemy and the convergence of multiple concepts into a single Japanese lemma, we first define the entailment probability between two vectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ as:

$$\delta(\boldsymbol{v}_1, \boldsymbol{v}_2) = \frac{1}{1 + e^{\Xi(\boldsymbol{v}_1, \boldsymbol{v}_2) - \psi(\boldsymbol{v}_1)}} \tag{7}$$

Let $S(w)$ be the set of vectors associated with word w. The *oracle*'s final decision logic is defined as the maximum score among all possible combinations:

$$oracle(w_1, w_2) = \max_{\boldsymbol{v}_1 \in S(w_1), \boldsymbol{v}_2 \in S(w_2)} \delta(\boldsymbol{v}_1, \boldsymbol{v}_2) \tag{8}$$

The system treats the relation as a valid logical axiom if $oracle(w_1, w_2) > 0.5$.

5 System Evaluation

We integrated *oracle* into our NLI system and evaluated its overall impact. It is important to note that the primary objective of this evaluation is not to benchmark quantitative performance on a large-scale dataset, but rather to provide a proof-of-concept (PoC) for the proposed hybrid architecture. Specifically, we aim to demonstrate that dynamic knowledge injection by *oracle* successfully bridges the "knowledge bottleneck" without compromising the logical rigor and formal provability inherent in DTS.

To this end, we first present a case study to illustrate the internal mechanism of the proof

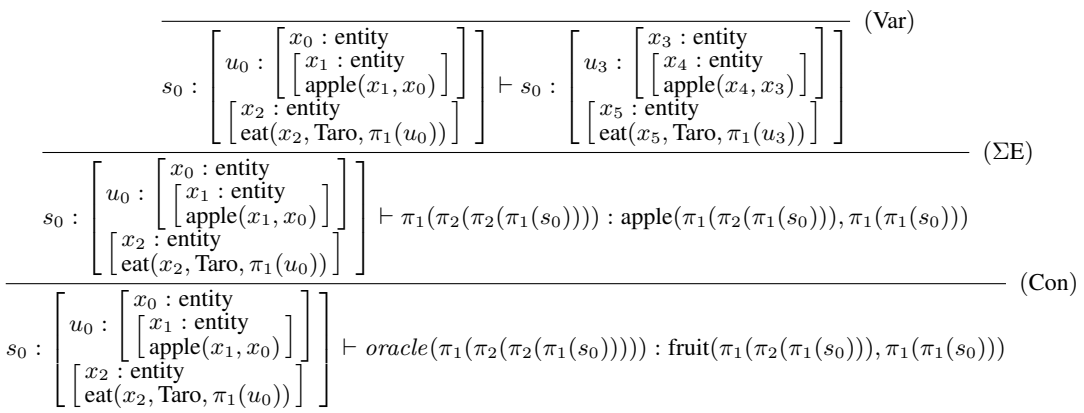

Figure 2: Part of proof search result

search, followed by a rigorous pipeline evaluation using a carefully curated set of monotonicity reasoning tasks.

5.1 Case Study: Lexical Entailment with Oracle

To verify the functional integrity of the integrated *oracle*, we consider a representative upward monotonicity task involving a hyponymy-hypernymy relation:

> **Premise:** Taro eats an apple.
> **Hypothesis:** Taro eats a fruit.

In conventional logic-based NLI systems, this inference would typically result in *Unknown* unless the specific lexical knowledge—*every apple is a fruit*—is manually encoded as a logical axiom. In contrast, our proposed framework successfully yields a *Yes* result without requiring additional explicit premises.

Figure 2 illustrates a fragment of the proof diagram generated for this case. The diagram follows the standard proof-theoretic notation of DTS: a judgment $\Gamma \vdash M : A$ asserts that the proof term M inhabits type A under context Γ; (ΣE) is the elimination rule for Σ-types (dependent pairs); and the projection operators π_1 and π_2 retrieve the first and second components of such a pair, respectively. We omit a full presentation of the DTS inference rules here and refer the reader to the DTS literature (Bekki and Mineshima, 2017; Bekki et al., 2023) for their formal definitions. As shown in the figure, when the entailment relationship between *apple* and *fruit* becomes necessary during the proof search, `wani` internally invokes the *oracle*. The hyperbolic classifier determines the validity of this relation based on its embedding space (e.g., computing an entailment probability that exceeds the 0.5 threshold) and dynamically injects it as a logical axiom into the reasoning process. This confirms that the system can bridge the knowledge gap through a hybrid approach combining formal logic and continuous geometric embeddings.

5.2 Dataset Construction

Based on the successful verification in the case study, we proceeded to a pipeline evaluation. Rather than relying on a large-scale, noisy dataset, we constructed a highly controlled, diagnostic Japanese test suite by manually translating and adapting 20 specific test cases from the Monotonicity Entailment Dataset (MED) (Yanaka et al., 2019).

The rationale for selecting monotonicity reasoning as our benchmark is that it rigorously tests a system's ability to integrate formal logical structure with continuous lexical knowledge. While the direction of entailment is governed by the logical context (e.g., upward or downward monotonicity), the actual proof often requires implicit world knowledge that is absent from the premise.

Specifically, we focused on 20 scenarios of upward monotonicity, where an entailment validly holds from a specific concept to a more general one in an upward-entailing context. Because our *oracle* is designed to dynamically supplement this missing hierarchical knowledge, these curated cases provide an ideal environment to evaluate the precise synergy between DTS-based logical inference and the hyperbolic classifier, strictly isolating the reasoning process from irrelevant

syntactic parsing noise.

5.3 Experimental Results

Tables 4 and 5 present the confusion matrices of the inference results before and after the integration of the *oracle*, respectively, where rows correspond to system predictions and columns to gold labels.

Table 4: Before integration

	Yes	No	Unk	Other
Yes	0	0	0	0
No	0	0	0	0
Unk	10	0	10	0
Other	0	0	0	0

Table 5: After integration

	Yes	No	Unk	Other
Yes	8	0	1	0
No	0	0	0	0
Unk	2	0	9	0
Other	0	0	0	0

Table 6: Detailed performance metrics per class (After integration)

Class	**Precision**	**Recall**	**F1-score**
Yes	0.889	0.800	0.842
Unknown	0.818	0.900	0.857
Macro Average	0.854	0.850	0.850

Table 6 demonstrates the substantial impact of the *oracle* on the NLI system. Most notably, recall for the *Yes* class rises from 0.0% to 80.0%. This indicates that the vast majority of upward entailment cases, which previously caused the system to default to an *Unknown* state, are now resolvable. This confirms that the hyperbolic classifier functions effectively as a dynamic knowledge bridge, enabling the DTS-based inference engine to complete proofs by identifying hierarchical relations in a robust, data-driven manner.

Furthermore, the precision of 88.9% and F1-score of 84.2% for the *Yes* class demonstrate that the *oracle* does not merely provide random guesses but rather offers highly reliable lexical information. The high precision is particularly noteworthy in a logic-based framework like DTS, where even a single incorrect axiom can lead to a logically sound but factually false conclusion. The fact that the system maintains high precision while substantially increasing recall suggests that the geometric properties of the hyperbolic embedding space—specifically its ability to model asymmetric entailment—align well with the requirements of formal monotonicity reasoning.

Additionally, the macro-averaged F1-score of 85.0% confirms the overall robustness of the hybrid system, ensuring that the integration of the neural components does not degrade the performance on neutral cases. In summary, these results provide empirical evidence that the hybrid architecture of Neural DTS effectively mitigates the "knowledge bottleneck" problem. By decoupling logical structure from continuous lexical knowledge, the system achieves a degree of flexibility and coverage that was previously unattainable for purely logic-based NLI systems, all while retaining the formal provability of DTS.

6 Error Analysis

During inference, we observed several cases in which the probability assigned by the *oracle* to a hypernym-hyponym relation fell below the decision threshold of 0.5, resulting in incorrect entailment predictions at the sentence level. A representative example is shown below:

> **Premise:** Hanako gave Taro a rose.
> **Hypothesis:** Hanako gave Taro a flower.

In this case, successful inference requires the *oracle* to capture the taxonomic relation between *rose* and *flower*. However, the model produced a score of $oracle(\text{flower}, \text{rose}) = 0.0844$, which is well below the threshold, leading to a misclassification.

To determine whether such errors arise from the threshold setting or from more fundamental limitations of the embedding model, we conducted additional analyses focusing on word-substitution pairs contained in the MED dataset.

6.1 Methodology

To systematically analyze the probability distributions output by *oracle* for the substituted word pairs, we extracted and preprocessed evaluation pairs using the following procedure:

1. **Data Extraction**: We extracted 1,403 instances containing "lexical knowledge" in the genre field from the MED dataset.

2. **Word Pair Identification and Noise Removal**: We compared the premise and hypothesis sentences to extract only the substituted word pairs. During the process, functional words such as prepositions and personal pronouns, which introduce noise during translation or evaluation, were excluded.

3. **Translation and Filtering**: The extracted English word pairs were manually translated

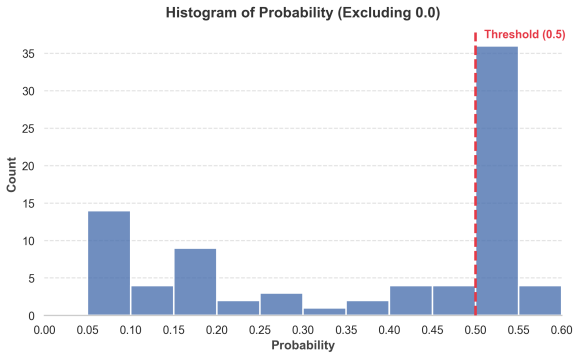

Figure 3: Probabilities for Entailment pairs ($N = 100$).

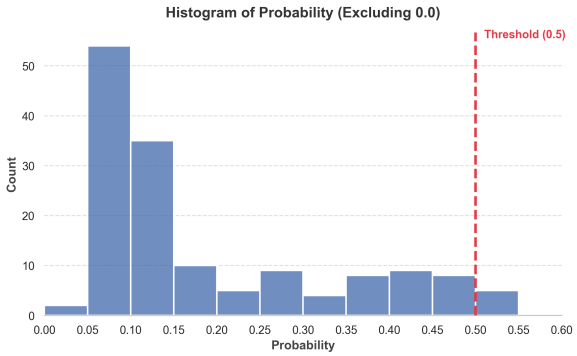

Figure 4: Probabilities for Neutral pairs ($N = 180$).

into Japanese. Pairs lacking an appropriate Japanese translation or those resulting in identical translated words were excluded.

4. **Normalization of Inference Direction**: We aligned the directional relationship between the premise word w_1 and the hypothesis word w_2 based on sentence monotonicity. For **upward-entailing** contexts, we evaluated the pair as (w_2, w_1) to test the hyponym-hypernym relation ($w_1 \sqsubseteq w_2$). For **downward-entailing** contexts, where the entailment direction is reversed, we aligned the pair as (w_1, w_2) to evaluate $w_2 \sqsubseteq w_1$. This ensures the *oracle* consistently measures the taxonomic subordination required for sentence-level entailment.

5. **Selection of Evaluation Pairs**: From the processed data, we constructed a final set of unique evaluation pairs categorized as Entailment and Neutral, ensuring no duplication.

6.2 Evaluation Results

Through the procedure described above, we obtained 100 Entailment pairs and 180 Neutral pairs. We calculated the entailment probabilities for these pairs using the *oracle* and classified them with a threshold of 0.5. Note that pairs containing out-of-vocabulary (OOV) words were excluded from the accuracy calculations.

6.3 Discussion

Mismatch Between Strict Hypernymy and Meronymy: As shown in Figure 3, the relatively low accuracy for Entailment pairs (62.12%) is primarily due to the structural divergence between the strict hypernymy modeled by the *oracle* and the broader semantic relations in the MED dataset. Specifically, many failure cases involve meronymy (part-whole relations). As shown in Table 7, pairs such as "Mt. Fuji–Japan" represent valid contextual entailments; however, their relative positions in the embedding space do not satisfy the geometric inclusion criteria of the hypernymic cones. Since the *oracle* is mathematically optimized to capture hypernymy, it assigns low probabilities to these relations. This result demonstrates that the model is behaving exactly as specified—preserving high specificity by excluding non-hypernymic pairs—even if it leads to lower recall in the presence of semantic diversity.

Table 7: Examples of Entailment pairs in MED misclassified by *oracle* due to non-taxonomic relations.

Hyper	Hypo	Score
Song	Lyrics	0.2617
Europe	Spain	0.1084
Skin	Epidermis	0.1696

High Specificity in Neutral Pairs: Conversely, as shown in Figure 4, the accuracy for the Neutral pairs demonstrated an extremely high value of 96.64%. This indicates that false positives—where the model erroneously outputs a high probability for word pairs inherently lacking a hierarchical relationship—rarely occur. This high specificity confirms that the *oracle* is highly reliable, consistently avoiding the hallucination of invalid logical axioms during the proof search.

The Challenge of Out-of-Vocabulary Words: For word pairs containing OOV words, the *oracle* defaults to a probability of 0.00. An analysis of these excluded pairs revealed that the majority consist of specific geographic names, personal names, or highly specific concepts (e.g., "leafy vegetables" or "hummingbirds"). This highlights an issue with the vocabulary coverage of the

pre-trained embeddings rather than a limitation of the geometric model or the threshold setting. Expanding the training dataset to improve vocabulary coverage remains a future challenge.

Summary and Threshold Justification: The above analysis demonstrates that the observed inference failures are not primarily caused by an inappropriate threshold. Instead, they stem from a fundamental mismatch between the *oracle*'s strict geometric design (optimized for IS-A relations) and the semantic diversity present in the evaluation dataset. Lowering the threshold indiscriminately to capture meronymy would compromise the high specificity (96.64%) observed for Neutral pairs. Therefore, retaining the current threshold of 0.5 is mathematically and practically appropriate for rigorous hypernym-hyponym classification.

7 Conclusion

In this study, we proposed and implemented a novel framework for integrating a hyperbolic classifier, based on Hyperbolic Entailment Cones, into a natural language inference system grounded in DTS. We demonstrated that the proposed hybrid system can automatically derive dynamic logical axioms to correctly resolve lexical entailments, which were previously impossible to prove without manual axiom creation. These results highlight the effectiveness of combining a broad coverage of continuous geometric embeddings with the formal reliability and explainability inherent in linguistically-oriented NLI systems.

Furthermore, our error analysis confirmed the high "geometric rigor" of the proposed *oracle*. Although the evaluation dataset contained positive entailment examples based on non-IS-A relations such as meronymy, our model successfully rejected them. This confirms that the approach safely injects external lexical knowledge while preserving strict geometric classification accuracy for taxonomic relationships.

To address current limitations, specifically misclassifications caused by out-of-vocabulary words, reconsidering and expanding the training dataset is an immediate next step. For future work, we plan to extend the model's inferential scope by mapping a broader set of semantic relationships, such as meronymy, into distinct geometric representations within the hyperbolic space, ultimately aiming to establish the proposed framework as the foundation for designing a highly versatile Neuro-Symbolic system.

Limitations

While our proposed framework successfully bridges the knowledge bottleneck for hypernymy in logic-based NLI, several limitations remain.

First, the current *oracle* is strictly optimized for taxonomic IS-A relations. As discussed in the Error Analysis, it struggles with other contextually valid semantic relations, such as meronymy (part-whole relations). Expanding the hyperbolic embedding space to accommodate multiple distinct relational structures is necessary for broader NLI coverage.

Second, the system's reasoning capacity is strictly bounded by the vocabulary coverage of the pre-trained WordNet embeddings. Out-of-vocabulary (OOV) words, particularly highly specific concepts and proper nouns, currently default to a zero entailment probability, leading to inference failures.

Finally, our pipeline evaluation was conducted on a small, carefully curated diagnostic set of 20 upward-monotonicity cases in Japanese, designed as a proof-of-concept rather than a large-scale benchmark. Scaling this evaluation to a substantially larger and more diverse test suite is a natural next step. We also restricted the present study to upward-entailing contexts. Downward-entailing contexts are logically more involved, primarily because they typically embed negation and other polarity-reversing operators. However, DTS already represents negation and such operators natively within its proof system, and the *oracle* supplies only lexical IS-A axioms whose validity is independent of the surrounding logical polarity. Because the proof search consumes these axioms in the same way regardless of monotonicity direction, we expect the extension to downward-entailing contexts to be conceptually straightforward; the principal remaining work is empirical validation at scale, which lies beyond the scope of this proof-of-concept and is left for future work.

Acknowledgments

This work was supported in part by JST CREST Grant Number JPMJCR2565 and JSPS KAKENHI Grant Number JP23H03452.

References

Daisuke Bekki and Koji Mineshima. 2017. Context-passing and underspecification in dependent type semantics. In *Studies in Linguistics and Philosophy*, pages 11–41.

Daisuke Bekki, Koji Mineshima, and Elin McCready, editors. 2023. *Logic and Engineering of Natural Language Semantics: 19th International Conference, LENLS19, Tokyo, Japan, November 19–21, 2022, Revised Selected Papers*, volume 14213 of *Lecture Notes in Computer Science*. Springer.

Daisuke Bekki, Ribeka Tanaka, and Yuta Takahashi. 2021. Integrating deep neural networks with dependent type semantics. In *Proceedings of the Symposium Logic and Algorithms in Computational Linguistics 2021 (LACompLing2021)*.

Daisuke Bekki, Ribeka Tanaka, and Yuta Takahashi. 2022. Learning knowledge with Neural DTS. In *Proceedings of the 3rd Natural Logic Meets Machine Learning Workshop (NALOMA III)*, pages 17–25.

Francis Bond, Timothy Baldwin, Richard Fothergill, and Kiyotaka Uchimoto. 2012. Japanese SemCor: A sense-tagged corpus of Japanese. In *Proceedings of the 6th International Conference of the Global WordNet Association (GWC-2012)*, pages 56–63.

Francis Bond and Ryan Foster. 2013. Linking and extending an open multilingual Wordnet. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1352–1362, Sofia, Bulgaria. Association for Computational Linguistics.

Hinari Daido. 2022. Development of an automated theorem prover for the fragment of dts. Master thesis, Ochanomizu University.

Hinari Daido and Daisuke Bekki. 2020. Development of an automated theorem prover for the fragment of dts. In *In the 17th International Workshop on Logic and Engineering of Natural Language Semantics (LENLS17)*.

Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic entailment cones for learning hierarchical embeddings. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1646–1655.

Mizuki Iinuma, Yuta Takahashi, Sora Tagami, and Daisuke Bekki. 2023. Neural DTS: A hybrid NLI system combining two procedural approaches. In *Proceedings of Procedural and computational models of semantic and pragmatic processes, ESSLLI2023 workshop, Ljubljana, Slovenia.*

Per Martin-Löf. 1984. *Intuitionistic Type Theory, Vol. 9*. Bibliopolis Naples.

Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems (NIPS 2017)*, pages 6338–6347.

Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. 2013. Reasoning with neural tensor networks for knowledge base completion. In *Advances in Neural Information Processing Systems (NIPS 2013)*, pages 926–934.

Mark Steedman. 1996. *Surface Structure and Interpretation*. The MIT Press, Cambridge.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press.

Asa Tomita, Mai Matsubara, Hinari Daido, and Daisuke Bekki. 2025. Natural language inference with CCG parser and automated theorem prover for DTS. In *Proceedings of the Second Workshop on the Bridges and Gaps between Formal and Computational Linguistics (BriGap-2)*, pages 1–7.

Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2016. Order-embeddings of images and language. In *Proceedings of the 4th International Conference on Learning Representations (ICLR 2016)*.

Hitomi Yanaka, Koji Mineshima, Daisuke Bekki, Kentaro Inui, Satoshi Sekine, Lasha Abzianidze, and Johan Bos. 2019. Can neural networks understand monotonicity reasoning? In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 31–40.