MRNBOX: Ripple Diffusion Meets Geometric Reasoning for Taxonomy Completion

Anonymous authors
Paper under double-blind review

ABSTRACT

The Taxonomy Completion (TC) task aims to find optimal insertion positions for new concepts under hypernym-hyponym constraints, but faces three challenges: sparse descriptions and polysemy weaken semantic embeddings with no quantifiable parent-priority mechanism; traditional GNNs miss path-dependent decay in hierarchical correlations; and global traversal is inefficient with invalid candidate interference. To address these, we propose MRNBOX, which resolves semantic sparsity via adaptive embeddings fusing conceptual descriptions, average path lengths, and topological association features; models hierarchical structures via ripple diffusion and sparse Mixture-of-Experts (MoE) for dynamic neighbor screening; and performs two-stage geometric reasoning, where box embedding exclusion first prunes irrelevant branches before fine-grained matching refines positioning. This enables efficient semantic-structural collaborative modeling for dynamic taxonomy maintenance.

1 INTRODUCTION

Taxonomies are tree-structured hierarchies built on hypernym-hyponym ("is-a") relations and are critical across domains (Yu et al., 2020), e.g., e-commerce product taxonomies for search and recommendation (Zhang et al., 2014), the ACM Computing Taxonomy for academic topics (Rous, 2012), and the ICD-11 disease taxonomy for medical diagnosis (Reed, 2010). Their continuous evolution drives automatic TC (Zhang et al., 2021), which locates optimal positions for new concepts (query nodes): a non-leaf query is defined by a hypernym (parent) and a hyponym (child), while a leaf query lies in the last layer and is defined by its hypernym alone (Arous et al., 2023). As shown in Fig. 1, *Image Recognition* is inserted at the non-leaf position $\langle$*Computer Vision*, *Object Detection*$\rangle$, while *Facial Recognition* is inserted at the leaf position $\langle$*Computer Vision*$\rangle$.

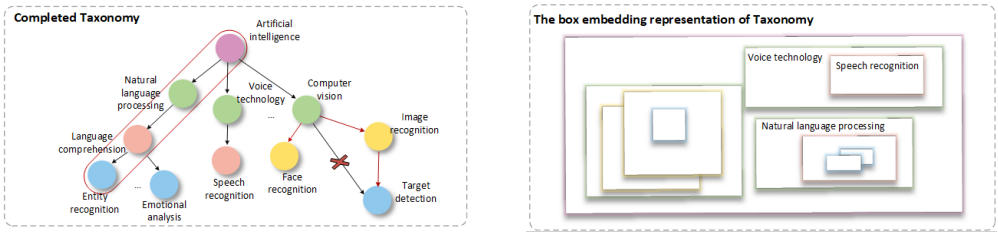

Figure 1: Example of Taxonomy Completion

Existing methods model taxonomies from semantic and structural views, yet face 3 core issues:

Issue 1: Limitations in semantic description for node embedding generation. Current approaches exploit semantic co-occurrence patterns in concept descriptions to identify inclusion relationships, proving effective for TC (Xu et al., 2025). However, they struggle with new concepts (e.g., inadequate descriptions) and semantically ambiguous cross-domain concepts (e.g., polysemous terms like *apple* in botany vs. consumer electronics). In fact, the topological arrangement of nodes inherently encodes their inclusion relationships. Moreover, insertion should prioritize the most specific parent (fine-grained priority principle), yet current methods lack quantitative evaluation of this criterion (D'Amico et al., 2025). In fact, node connections weaken with path depth, which measures hierarchical strength.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Issue 2: Structural modeling deficiencies. Mainstream methods primarily employ Graph Neural Networks (GNNs) (Jiang et al., 2022) to model neighbor and path information (Liu et al., 2021), yet treat all adjacent nodes as equally important during propagation, failing to differentiate node significance. Notably, node importance varies even within the same hierarchical level, where core and edge concepts coexist and importance is unevenly distributed. The Ripple spreading algorithm (Yu & Song, 2024) addresses this by modeling the influence of the central node (information source) over its neighbors, transmitting information outward layer by layer with decreasing intensity, but it cannot measure importance strength among nodes at the same level. We therefore apply Mixture-of-Experts (MoE) (Masoudnia & Ebrahimpour, 2014), using dynamic routing to selectively activate relevant experts for identifying mission-critical nodes.

Issue 3: Interference from irrelevant candidate positions. Existing methods traverse the entire taxonomy to compute similarity between query nodes and every candidate position (Shi et al., 2024), a global strategy that introduces invalid-candidate interference and significant computational overhead. In fact, parent and child nodes exhibit strict semantic inclusion, and node coverage narrows with path length: the closer a node is to the root, the wider its coverage. Consequently, no candidate search is needed within a root's direct child subtree if semantic containment is absent. Crucially, box embedding (Jiang et al., 2023) provides geometric exclusivity: a child box strictly contained in a parent box has near-zero intersection with boxes of other same-level parents (Gao et al., 2025). As shown in Fig. 1, the boxes for *Speech recognition* and *Natural language processing* are completely non-intersecting.

In summary, we propose MRNBOX, an efficient taxonomy completion framework integrating semantic and structural information. For issue 1, a multi-view node embedding adaptively fuses conceptual descriptions, average path lengths, and topological association features. For issue 2, hierarchical ripple sets organize multi-hop neighbors by hop distance, while an MoE system with sibling importance evaluators and descendant impact predictors dynamically filters critical nodes via layer-aware gating, mitigating local overfitting and global over-expansion. For issue 3, a box embedding-based exclusion mechanism first prunes irrelevant branches via containment, then performs fine-grained semantic-structural matching to improve accuracy and efficiency.

The specific contributions of this paper are summarized as follows:

- **Adaptive embedding fusion.** We fuse conceptual descriptions, average path lengths, and topological association features via adaptive weighting, addressing semantic sparsity and structural quantification gaps.
- **Hierarchical ripple diffusion with MoE.** We capture exponential decay in taxonomic correlations through hierarchical ripple diffusion and dynamically screen critical nodes via a MoEs module, overcoming the homogeneous aggregation of conventional GNNs.
- **Two-stage geometric reasoning.** We exploit the geometric exclusivity of box embeddings to first prune irrelevant branches via coverage evaluation, then perform fine-grained semantic-structural matching.

2 Related Work

2.1 Taxonomy Completion

Automatic taxonomy updating comprises two paradigms: Taxonomy Expansion (TE) and Taxonomy Completion (TC). TE attaches new concepts as leaf nodes by finding optimal parents, while TMN (Zhang et al., 2021) pioneered TC to jointly match parent and child nodes, overcoming leaf-only insertion limits. TC models then evolve along four lineages: Embedding-Optimized methods (TMN (Zhang et al., 2021), QEN (Wang et al., 2022)) improve structural expressiveness via multi-relational embedding optimization but degrade under pseudo-leaf noise; Contrastive & Representation Enhancement methods, e.g., TaxoEnrich (Jiang et al., 2022) (structural-semantic dual encodings), TaxoComplete (Arous et al., 2023) (position-enhanced matching with anchor sets), and CoSTC (Niu et al., 2024) (intra/inter-view contrastive learning), address single-signal bottlenecks; Generative & Multi-Stage approaches such as GenTaxo (Zeng et al., 2021) and TEF (Meng et al., 2025) handle complex hierarchical extension; and Prompt-Based methods like TacoPrompt (Xu et al., 2023) mitigate leaf-node overfitting via outcome contextualization.

2.2 BOX EMBEDDING

Box embeddings offer a geometric alternative for hierarchical modeling, representing concepts as hyperrectangles that capture containment, dependencies, and uncertainty. BoxTaxo (Jiang et al., 2023) encodes asymmetric hypernymy-hyponym constraints via hypercube containment, while TAXBOX (Xue et al., 2023) improves matching with scoring functions for leaf versus non-leaf insertion. Beyond taxonomy, they prove effective across domains: BoxCD (Gao et al., 2025) for cognitive diagnosis, InBox (Xu et al., 2024) for recommendation, Task2Box (Daroya et al., 2024) for asymmetric task relations, and BERE (Messner et al., 2022)/BoxTE (Hwang et al., 2022) for relation extraction and knowledge graph completion. A common theme is that boxes encode hierarchical inclusion through geometric enclosure, yielding an explicit set-theoretic interpretation hard to obtain in hyperbolic or cone-based spaces (Daroya et al., 2024). Unlike curvature-dependent distances, nested box regions represent parent–child structures (Jiang et al., 2023), while intersections and overlaps express sibling proximity, uncertainty, and multi-parent configurations (Xu et al., 2024). Independent lower/upper-bound parameterization further enables fine control over asymmetric relations and hierarchical tightness (Dasgupta et al., 2020). These properties establish box embeddings as a foundation for our structural reasoning components.

3 METHODOLOGY

As illustrated in Fig. 2, MRNBOX comprises three core components. Module (a) generates initial node embeddings by fusing (1) CLIP-pretrained concept descriptions, (2) average path length, and (3) topological association features. Module (b) dynamically builds a hierarchical neighbor set via ripple diffusion, where the central node acts as the wave source and neighbors expand layer-wise by hop count, with a dynamic attenuation mechanism covering multi-hop structures. To emphasize same-level node importance, we design an MoE module with local experts (ChildExpert/SiblingExpert) for low-hop tree structures and global experts (CrossTreeExpert/HubExpert) for long-range dependencies, adaptively combined by a layer-aware gating mechanism. Module (c) has two stages: dynamic pruning of independent branches, then structure-aware scoring (balancing parent–child inclusion and sibling compatibility) to optimize candidate positions.

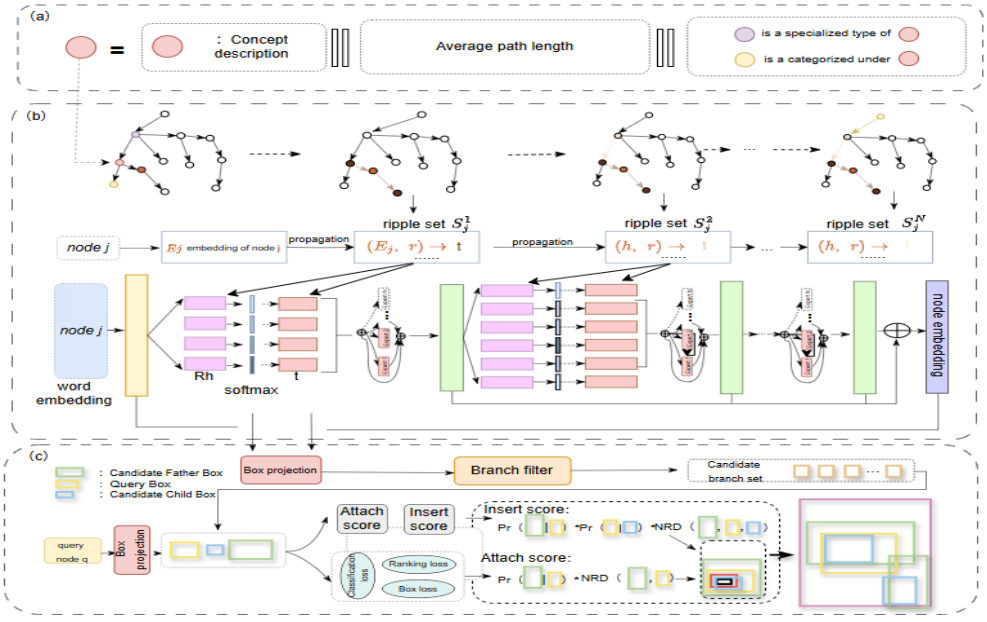

Figure 2: The Framework of MRNBOX

3.1 MULTI-VIEW FUSION NODE EMBEDDING REPRESENTATION

For $n \in N$, we fuse conceptual description d, topological association features s, and average path length p. The encoder projects d and s into the initial embedding $\mathbf{h}_n$. Here, $\boldsymbol{x}_n^{(d)}$, $\boldsymbol{x}_n^{(s)}$, and $\boldsymbol{x}_n^{(p)}$ de-

note the conceptual, topological association (ancestor, descendant, sibling), and average path length embeddings, respectively. The initial node embedding $\boldsymbol{h}_n$ is formed by Eq. (1).

$$\boldsymbol{h}_n = f_{\text{proj}}\Big(\big[\boldsymbol{x}_n^{(d)} \| \boldsymbol{x}_n^{(s)} \| \boldsymbol{x}_n^{(p)}\big]\Big) = W \cdot \big[\boldsymbol{x}_n^{(d)} \| \boldsymbol{x}_n^{(s)} \| \boldsymbol{x}_n^{(p)}\big] + \boldsymbol{b} \tag{1}$$

Here, $\| \cdot \|$ is vector concatenation, f_{proj} is linear projection, $W \in \mathbb{R}^{d \times (d_d + d_s + d_p)}$ and $\boldsymbol{b} \in \mathbb{R}^d$ are trainable parameters. $\boldsymbol{x}_n^{(d)}$ and $\boldsymbol{x}_n^{(s)}$ come from a pre-trained CLIP text encoder (Eq. (2)-(3)).

$$\boldsymbol{x}_n^{(d)} = CLIPTextEncoder(n_{\text{desc}}) \in \mathbb{R}^{d_t} \tag{2}$$

$$\boldsymbol{x}_n^{(s)} = \tfrac{1}{|T_n|} \sum\nolimits_{t \in T_n} CLIPTextEncoder(t), T_n = \big\{t_n^a, t_n^d, t_n^s | a \in A(n), d \in D(n)\big\} \tag{3}$$

Here, n_{desc} is the concept description of n; $A(n)$ and $B(n)$ are its ancestor and descendant sets; and $t_n^{(a)}, t_n^{(d)}, t_n^{(s)}$ are textual descriptions of its topological correlations with ancestor a, descendant d, and sibling s nodes. Specifically, for $\forall a \in A(n)$, we generate $t_n^a =$ "a is a specialized type of n''; for $\forall d \in D(n)$, $t_n^d =$ "d is categorized under n''; and for siblings, given parent p and child set $S(n) = \textsc{Succ}(p) \setminus \{n\}$, $t_n^s =$ "$n, s_1, s_2, \ldots,$ and s_k share the same parent category p''.

$x_n^{(p)}$ in Eq. (1) is given by Eq. (4).

$$x_n^{(p)} = \big[|A(n)|, |D(n)|, \bar{l}_n\big] \in \mathbb{R}^3 \tag{4}$$

Here, $A(n) = \{a \mid a \in \mathcal{N}, a \text{ is the ancestor of } n\}$, $D(n) = \{d \mid d \in N, d \text{ is a descendant of } n\}$.

$\bar{l}_n$ is the average distance from node n to all its ancestors, as formalized in Eq. (5).

$$\bar{l}_n = \sum\nolimits_{a \in A(n)} dist(n, a)/|A(n)| \tag{5}$$

3.2 Ripple sets-based graph representation learning

MRNBOX couples ripple diffusion with sparse MoE in a depth-aware manner: ripple diffusion builds hop-dependent multi-hop neighborhoods, while MoE selects hop-specific experts to score and filter candidates per stage, forming an "expand–select" pipeline where ripple expansion defines the structural scope and MoE identifies relevant structural roles. The final embedding f_n comes from ripple set-based aggregation with MoE-driven node selection.

Ripple Set Construction. Inspired by Ripple spreading armknecht2015ripple, we develop a hierarchical neighbor expansion with distance-proportional decay: the central node propagates influence through successive layers (1-hop $\rightarrow$ 2-hop $\rightarrow \cdots$), with signal strength attenuating per hop. Given a taxonomy $T = (N, E)$ with center $u \in N$, the s-th layer ripple set $R^s(u)$ is defined in Eq. (6).

$$R^s(u) = \begin{cases} \{u\} & , s = 0 \\ \Big\{v \in N_0 \mid \exists n \in R^{(s-1)}(u), \big((n,v) \in E \vee (v,n) \in E\big), \wedge v \notin \bigcup_{i=0}^{s-1} R^{(i)}(u)\Big\} & , \text{other} \end{cases} \tag{6}$$

Note that when $s = 1$, we include u' sibling nodes sharing the same parent in the 1-layer ripple set. $R^{(1)}(u)$, as defined in Eq. (7).

$$R^{(1)}(u) = \underbrace{\{v \in N \mid (u,v) \in E \vee (v,u) \in E\}}_{\text{(a)}} \cup \underbrace{\{v \in N \mid \exists p \in N,\ (p,u) \in E \wedge (p,v) \in E \wedge v \neq u\}}_{\text{(b)}} \tag{7}$$

Here, (a) denotes u' direct neighbors, while (b) denotes its sibling nodes sharing parent node p.

The ripple expansion propagates up to depth S, with node u' complete ripple set defined in Eq. (8).

$$G(u) = \bigcup\nolimits_{s=1}^{S} R^{(s)}(u) \tag{8}$$

Each node $v \in G(u)$ is labeled with its layer $s_v \in [1, S]$, indicating that v is u's s_v-th neighbor.

Each hop encodes a distinct regime: 1-hop captures parent–child–sibling patterns, 2-hop reveals cross-branch interactions, and deeper hops expose long-range dependencies, motivating a hierarchical expert design in MoE.

Dynamic Ripple Node Selection via MoE.

Since ripple diffusion yields depth-stratified neighborhoods, we assign an expert pool per hop range, with MoE gating producing hop-dependent weights: shallow hops emphasize Child/Parent/Sibling experts, deeper hops rely on CrossTree/Hub experts. MoE thus acts as a depth-aware selector.

Low-Hop Experts $s \leq 2$: Within 2-hop ripple sets, the local neighborhood maintains a clear tree-like hierarchied. We thus design specialized experts—ChildExpert, SiblingExpert, and ParentExpert—to capture distinct semantic relations and structural features at different levels.

- **ChildExpert**: It models child node features relative to a center node by combining attention-based structural modeling with MLP-based semantic matching, as defined in Eq. (9).

$$\text{Score}^{(s)}_{\text{child}}(u,v) = \underbrace{((W_Q^{(s)} h_u) \cdot (W_K^{(s)} h_v))/\sqrt{d}}_{(a)} + \underbrace{MLP(h_u \oplus h_v)}_{(b)} \tag{9}$$

Here, query $W_Q^{(s)} h_u$ captures the expected child pattern at layer s; key $\mathbf{k} = W_K^{(s)} h_v$ represents candidate v. Part (a) computes their scaled dot-product for topological relevance; part (b) models (u, v)'s semantic compatibility via a nonlinear transformation of concatenated features.

- **SiblingExpert:** It evaluates sibling nodes for complementary or exclusive relationships with a scoring function formalized in Eq. (10).

$$\text{Score}^{(s)}_{\text{sib}}(u,v) = sim_{base} - \lambda \cdot \max(0, sim_{base} - \ sim_{neg} + \gamma) \tag{10}$$

Here, sim_{base} and sim_{neg} denote the cosine similarities for positive pair (u, v) and negative pairs (randomly non-siblings), with λ controling the negative samples' influence, and γ as the threshold.

- **ParentExpert:** It evaluates parent candidates v for central node u, based on semantic relevance and topological reachability, with scoring formalized in Eq. (11).

$$\text{Score}^{(s)}_{\text{parent}}(u,v) = \underbrace{sim_{\text{base}}(h_u, h_v)}_{\text{(a)}} \cdot \underbrace{\prod_{(a,b)\in Path(v,u)} sim(h_a, h_b)}_{\text{(b)}} \tag{11}$$

Here, (a) measures parent–child semantic proximity; (b) measures path-wise node similarity. Similarity decays exponentially for weakly-connected edges, quantifying topological consistency.

High-hop Experts $s \geq 3$: As depth increases and local structures sparsely the model employs long-range experts, including Cross-Tree Experts and Hub Experts.

- **Cross-Tree Experts model.** It semantic relationships across subtrees by measuring inter-subtree semantic overlap with scoring functions defined in Eq. (12).

$$\text{Score}^{(s)}_{\text{cross}}(u,v) = \text{sim}(h_u, h_v) \cdot \underbrace{(|\text{Tree}_u \cap \text{Tree}_v|)/(|\text{Tree}_u \cup \text{Tree}_v|)}_{(*)} \tag{12}$$

Here, $(*)$ uses the Jaccard coefficient to measure subtree similarity, computing the ratio of shared nodes to unique nodes in both subtrees, where $Tree_u$ denotes all nodes in the subtree rooted at u.

- **Hub Experts.** It quantifies a node's topological hub importance by evaluating local connectivity and global betweenness centrality with the scoring function formalized in Eq. (13).

$$\text{Score}^{(s)}_{\text{hub}}(u,v) = \deg(v) \cdot \underbrace{\sum\nolimits_{s\neq v\neq t} (\sigma_{st}(v)/\sigma_{st})}_{(*)} \tag{13}$$

Here, $deg(\cdot)$ is the degree of node u. $(*)$ counts shortest paths between all pairs (s, t) through v: σ_{st} is the number of shortest paths between s and t, and $\sigma_{st}(v)$ those passing through v.

Our layer-aware gating dynamically weights expert scores for node u and its candidate set R_s across hierarchy levels and structural types. The score for expert m is given in Eq. (14).

$$\alpha_m^{(s)} = softmax_m\left(W_m \cdot \left[h_u \| \bar{h}_t \| \text{PE}^{(t)}\right]\right) \tag{14}$$

Here, h_u is the initial embedding of central node t, $\bar{h}_t$ the mean ripple set features at layer t, and $PE^{(t)}$ the layer-wise positional encoding; $[h_u \parallel \bar{h}_t \parallel PE^{(t)}]$ is the concatenated input.

Thus, the node's final fused score is given by Eq. (15).

$$Score^{(s)}(u,v) = \sum\nolimits_m \alpha_m^{(s)} \cdot Score_m^{(s)}(u,v), m \in \{\text{child}, \text{sib}, \text{parent}, \text{cross}, \text{hub}\} \tag{15}$$

3.3 BOX GEOMETRIC FILTERING AND RE-RANKING

In box-embedding taxonomy, each parent box contains its children's, with sizes decreasing from root to leaves. This enables efficient pruning: if a query fails containment with a root's direct child, we prune the entire subtree, since no downstream box can satisfy containment under the size hierarchy.
Filtering. For node n with embedding f_n, its box is given in Eq. (16).

$$b_n = \left[z_n^{\min}, z_n^{\max}\right] \in \mathbb{R}^{2d} \tag{16}$$

Here, $z_n^{\min} \leq z_n^{\max}$ are the box's minimum and maximum boundaries, as specified in Eq. (17)–(18).

$$z_n^{\min}, z_n^{\max} = o_n[:d], o_n[d:] \tag{17}$$

$$z_n^{\max} = \max\left(z_n^{\min} + \epsilon, z_n^{\max}\right) \tag{18}$$

Here, $o_n[:d]$ and $o_n[-d:]$ denote the first and last d elements of o_n. $z_n^{\max}$ enforces a minimum margin ϵ between upper and lower boundaries across all dimensions to prevent invalid boxes. o_n is the Highway Network output after nonlinear decoding, as in Eq. (19).

$$o_n = W_{\text{out}} y^{(L)} + b_{\text{out}} \tag{19}$$

Here, $y^{(L)} \in R^H$ denotes the output representation of v after L layers Highway Network. Following Highway Network principle, the l-th layer output $y^{(l)}$ is given by Eq. (20).

$$y^{(l)} = \begin{cases} \mathrm{f}_n & l = 0 \\ H^{(l)} \odot D^{(l)} + x^{(l-1)} \odot (1 - D^{(l)}) & l \geq 0 \end{cases} \tag{20}$$

Here, $D^{(l)}$ and $H^{(l)}$ are the transformation gate and nonlinear result at layer l, as in Eq. (21).

$$D^{(l)} = \sigma\left(W_T^{(l)} x^{(l)} + b_T^{(l)}\right), H^{(l)} = \phi\left(W_H^{(l)} x^{(l)} + b_H^{(l)}\right) \tag{21}$$

Here σ is the Sigmoid activation function, ϕ denotes the nonlinear activation function and $x^{(l)} = y^{(l-1)}$. When $l = 0$, $y^{(l)}$ becomes the initial input f_n.

Ranking. For new node $q \in C$, its embedding f_q is encoded by CLIP and decoded into box b_q via a Highway Network. We first compute containment scores between b_q and the root's direct children $Root_1$ for initial filtering, as in Eq. (22).

$$Score_i = sim_{\text{box}}(b_{r_i}, b_q), \quad r_i \in \text{Root}_1, Root_1 = \{r_1 \mid r_1 \text{ is the direct child of } R\}_{l=1}^{n} \tag{22}$$

Here, $sim_{box}(\cdot,\cdot)$ computes box similarity via containment volume and centroid distance(Eq. (23)).

$$Sim(b_r, b_q) = \alpha \cdot IoU(b_r, b_q) + (1 - \alpha) \cdot \exp\left(-\beta \cdot \|c_q - c_r\|_2\right) \tag{23}$$

Here, $\alpha \in [0,1]$ balances volume and centroid similarity, while $\beta > 0$ penalizes centroid distance. c_n is the box centroid (Eq. (24)), and $IoU(b_r, b_q)$ the box overlap (Eq. (25)).

$$c_n = \left(z_n^{\min} + z_n^{\max}\right)/2 \tag{24}$$

$$IoU(b_r, b_q) = Vol(b_q \cap b_r)/Vol(b_q) \tag{25}$$

Here, $Vol(b_n)$ is the soft box volume. Top-k scoring branches form $Root_1^{top}$ candidates (Eq. (26)).

$$\text{Root}_1^{\text{top}} = Top - K\left(\{Score_i\}_{i=1}^{n}, k\right) \tag{26}$$

We prune unselected the root child nodes and their descendant subtrees $Desc(c)$ from the initial graph T_0, generating the trimmed tree $T_q \subseteq T_0$, as formalized in Eq. (27).

$$T_q = T_0 \setminus \left\{v \mid v \in Desc(c), c \in C_1 \setminus C_1^{\text{top}}\right\} \tag{27}$$

In the pruned tree T_q, we compute the combined geometric score $Score(q,p,c)$ between each candidate position (p,c) and the query box, as formulated in Eq. (28).

$$Score(q,p,c) = \underbrace{Sim_{\text{box}}(b_p, b_q)}_{s_1} \cdot \underbrace{Sim_{\text{box}}(b_q, b_c)}_{s_2} \cdot Trans(s_1, s_2) \tag{28}$$

Here, s_1 and s_2 represent box similarity scores, while $Trans(s_1, s_2)$ dynamically balances insertion vs append operations importance for structural matching, as defined in Eq. (29).

$$Trans(s_1, s_2) = \begin{cases} s_2 \cdot SoftMax\left(\frac{1}{\|c_q - c_c\|}\right), & \text{leaf} \\ 1, & \text{non-leaf} \end{cases} \tag{29}$$

The scoring adaptation prioritizes $Sim_{box}(b_q, b_c)$ for leaf insertions (child-box containment sensitive), while reducing adjustments for non-leaf cases (parent-dependent structural transitions).

4 EXPERIMENT

4.1 EVALUATION SETUP

We used four public datasets (Table 1): two Microsoft Academic graph subgraphs (**MAG_CS(MC)**, **MAG_PSY(MP)**) and two word web subgraphs (**WordNet-Verb(WV)** jurgens2016semeval, **SemEval-Food(SF)** bordea2015semeval).

Table 1: Statistics of Dataset Characteristics

Dataset	$\|N\|$	$\|\varepsilon\|$	$\|C\|$	Dataset	$\|N\|$	$\|\varepsilon\|$	$\|C\|$
MAG-CS	24754	42329	153726	Wordnet-Verb	13936	13408	51159
MAG-PSY	23187	30041	101077	SemEval-Food	1486	1533	6122

We select the latest and most representative TC models as baselines: TMN zhang2021taxonomy, QEN wang2022qen, TaxoEnrich jiang2022taxoenrich, TaxoComplete arous2023taxocomplete, TaxBox xue2023insert, and CoSTC niu2024contrastive. Following the protocol of TaxBox xue2024insert, we report Mean Rank (MR), Reciprocal Rank per query (RR), Mean Reciprocal Rank (MRR), Hit@k (micro recall), and P@k (macro precision).

4.2 COMPARATIVE ANALYSIS.

We compare MRNBOX with seven state-of-the-art models on 4 datasets (Table 2).

Table 2: Overall results on the four taxonomy completion datasets

Dataset	Method	MR	MRR	Hit@1	Hit@5	Hit@10	P@1	P@5	P@10	R@1	R@5	R@10
SF	TMN	559	0.211	0.037	0.113	0.160	0.074	0.046	0.032	0.107	0.187	0.220
	QEN	353	0.313	0.070	0.176	0.234	0.146	0.074	0.049	0.219	0.309	0.330
	TaxoEnrich	305	0.348	0.113	0.247	0.290	0.230	0.100	0.063	0.117	0.267	0.317
	TaxBox	261	0.344	0.115	0.244	0.283	0.277	0.117	0.068	0.211	0.328	0.343
	CoSTC	**91**	0.634	0.398	0.688	0.756	0.277	0.162	**0.105**	0.146	0.428	0.557
	MRNBOX	109	**0.754**	**0.525**	**0.711**	**0.761**	**0.728**	**0.197**	0.101	**0.728**	**0.871**	**0.937**
	Improv.	-18	12.0%	12.7%	2.3%	0.5%	45.1%	3.5%	-0.4%	58.2%	44.3%	38.0%
WV	TMN	1445	0.304	0.072	0.163	0.215	0.108	0.049	0.032	0.054	0.147	0.207
	QEN	2095	0.331	0.074	0.178	0.233	0.113	0.054	0.036	0.091	0.233	0.310
	TaxoEnrich	2873	0.320	0.069	0.168	0.229	0.106	0.052	0.035	0.039	0.090	0.123
	TaxBox	1363	0.311	0.091	0.191	0.241	0.155	0.065	0.041	0.092	0.264	0.402
	CoSTC	241	0.505	0.146	0.392	0.531	0.225	0.086	0.057	0.095	0.278	0.391
	MRNBOX	**205**	**0.890**	**0.571**	**0.857**	**0.857**	**0.667**	**0.208**	**0.112**	**0.717**	**0.878**	**0.978**
	Improv.	36	38.5%	42.5%	46.5%	32.6%	44.2%	12.2%	5.5%	62.2%	60%	58.7%
MC	TMN	639	0.204	0.036	0.099	0.139	0.156	0.086	0.060	0.056	0.145	0.189
	QEN	3960	0.147	0.017	0.062	0.097	0.076	0.054	0.042	0.083	0.229	0.395
	TaxoEnrich	5545	0.184	0.043	0.107	0.158	0.142	0.075	0.055	0.162	0.434	0.574
	TaxBox	596	0.240	0.051	0.139	0.184	0.238	0.131	0.087	0.638	0.705	0.791
	CoSTC	228	0.544	0.216	0.438	0.499	0.491	0.179	0.088	0.511	0.832	0.893
	MRNBOX	**165**	**0.648**	**0.341**	**0.567**	**0.601**	**0.631**	**0.209**	**0.111**	**0.631**	**0.907**	**0.943**
	Improv.	63	10.4%	12.5%	12.9%	10.2%	14%	3%	2.3%	12%	7.5%	5%
MP	TMN	212	0.471	0.141	0.305	0.377	0.287	0.124	0.077	0.175	0.369	0.446
	QEN	1778	0.293	0.103	0.150	0.206	0.103	0.059	0.042	0.199	0.403	0.477
	TaxoEnrich	2201	0.357	0.082	0.219	0.293	0.167	0.089	0.036	0.234	0.424	0.510
	TaxBox	256	0.444	0.127	0.286	0.359	0.288	0.129	0.081	0.288	0.592	0.756
	CoSTC	244	0.580	0.217	0.488	0.593	0.307	0.186	0.093	0.312	0.670	0.795
	MRNBOX	**212**	**0.733**	**0.312**	**0.554**	**0.666**	**0.667**	**0.237**	**0.115**	**0.667**	**0.741**	**0.864**
	Improv.	32	15.3%	9.5%	6.6%	7.3%	36%	5.1%	2.2%	35.5%	7.1%	6.9%

MRNBOX outperforms all existing methods in MRR and Hit@k metrics across all datasets, proving the significant advantage of its multi-view fusion node embedding representation in collaborative modeling of semantic and structural information, which achieves an MRR of 0.890 on Wordnet-Verb, 38% higher than CoSTC to effectively capture complex semantic relationships and topological features; it significantly enhances recall capability in large-scale datasets like MAG-CS, with

notably improved R@5 over CoSTC. The ripple set-based graph representation learning module dynamically filters important nodes via mixed experts to avoid multi-hop neighborhood noise accumulation and address the global structure perception bottleneck of traditional methods. Further, it demonstrates outstanding concept prediction precision, especially in SemEval-Food (Hit@1 surpasses TaxoBox), attributed to the box geometric filtering mechanism that uses hypercube geometric repulsion to filter irrelevant branches during re-ranking, enabling efficient hierarchical completion.

4.3 Ablation Study

Table 3 shows refined ablation results on four datasets. After re-training and re-tuning each variant, degradation patterns remain stable and conventional, confirming MRNBOX does not rely on any single module. Removing L0 (multi-view semantic–structural embeddings) causes the largest MRR/Hit@1 drops (e.g., -20% on SemEval-Food, -52% on SemEval-Verb, -12% on MAG-CS, -31% on MAG-PSY). Removing L2 (ripple-set hierarchical refinement) mainly harms Hit@5/10, though variants stay functional and sometimes slightly improve early-rank precision on deeper taxonomies (e.g., MAG-CS). Removing L1 (geometric box filtering) selectively reduces precision (e.g., Prec@1 drops by 14–35%), matching its late-stage refinement role. Combined removals (e.g., L0&L2) degrade most due to lost semantic and structural signals, yet stay within expected ranges. Overall, MRNBOX's gains come from the complementary contributions of L0, L2, and L1, not from any dominant or irreplaceable module.

Table 3: Ablation results on four datasets

Datasets	Method	MR	MRR	Hit@1	Hit@5	Hit@10	Prec@1	Prec@5	Prec@10	R@1	R@5	R@10
SF	MRNBOX	109	0.754	0.525	0.711	0.761	0.728	0.197	0.101	0.728	0.871	0.937
	w/o L0	100	0.553	0.200	0.480	0.480	0.312	0.153	0.075	0.312	0.718	0.718
	w/o L1	132	0.568	0.241	0.435	0.471	0.581	0.209	0.113	0.581	0.885	0.932
	w/o L2	247	0.631	0.394	0.552	0.570	0.750	0.210	0.108	0.750	0.950	0.950
	w/o L2&L1	277	0.584	0.311	0.471	0.505	0.756	0.227	0.121	0.311	0.471	0.505
	w/o L0&L2	292	0.442	0.386	0.401	0.463	0.218	0.123	0.074	0.218	0.600	0.709
SV	MRNBOX	205	0.890	0.571	0.857	0.857	0.667	0.200	0.100	0.717	0.978	0.978
	w/o L0	286	0.370	0.114	0.242	0.319	0.244	0.104	0.068	0.244	0.479	0.642
	w/o L1	366	0.441	0.141	0.302	0.373	0.239	0.103	0.063	0.239	0.514	0.636
	w/o L2	279	0.510	0.222	0.469	0.414	0.444	0.165	0.093	0.444	0.818	0.909
	w/o L2&L1	235	0.548	0.263	0.438	0.497	0.453	0.1508	0.0854	0.442	0.820	0.746
	w/o L0&L2	547	0.504	0.162	0.378	0.441	0.295	0.137	0.080	0.295	0.655	0.770
MC	MRNBOX	165	0.648	0.341	0.567	0.601	0.631	0.209	0.111	0.631	0.907	0.943
	w/o L0	214	0.570	0.168	0.397	0.480	0.214	0.113	0.071	0.214	0.549	0.686
	w/o L1	397	0.344	0.086	0.207	0.269	0.276	0.133	0.086	0.276	0.553	0.663
	w/o L2	209	0.691	0.475	0.608	0.658	0.760	0.194	0.105	0.760	0.933	0.963
	w/o L2&L1	304	0.312	0.105	0.214	0.257	0.498	0.204	0.121	0.498	0.618	0.709
	w/o L0&L2	534	0.303	0.056	0.167	0.243	0.272	0.102	0.074	0.363	0.458	0.672
MP	MRNBOX	199	0.733	0.312	0.554	0.666	0.667	0.237	0.115	0.667	0.741	0.864
	w/o L0	671	0.424	0.134	0.274	0.340	0.318	0.130	0.086	0.318	0.612	0.729
	w/o L1	327	0.376	0.114	0.241	0.316	0.293	0.126	0.083	0.293	0.572	0.694
	w/o L2	212	0.551	0.262	0.412	0.482	0.572	0.180	0.105	0.572	0.827	0.907
	w/o L2&L1	200	0.514	0.240	0.397	0.455	0.544	0.188	0.103	0.544	0.592	0.745
	w/o L0&L2	204	0.422	0.130	0.270	0.303	0.289	0.126	0.086	0.290	0.533	0.760

4.4 Parameter Sensitivity Analysis

Table 4 evaluates MRNBOX on leaf and non-leaf insertions. MRNBOX achieves superior performance on both. While all models perform better on leaf nodes due to concentrated semantics, MRNBOX leads all metrics on the SemEval datasets. On the more challenging non-leaf tasks, it outperforms baselines on SemEval-Food and surpasses CoSTC on SemEval-Verb, demonstrating robust multi-level modeling under ambiguity and consistent advantages across task complexities.

Table 4: Performance Comparison on Leaf and Non-leaf Insertion Positions

Method	MRR	Hit@5	R@10	MRR	Hit@5	R@10	MRR	Hit@5	R@10	MRR	Hit@5	R@10
	SemEval-Food (Leaf)			SemEval-Food (Non-leaf)			SemEval-Verb (Leaf)			SemEval-Verb (Non-leaf)		
TMN	0.538	0.415	0.415	0.164	0.120	0.061	0.425	0.238	0.328	0.136	0.107	0.068
QEN	0.732	0.642	0.689	0.209	0.320	0.091	0.568	0.386	0.484	0.224	0.153	0.118
TaxoEnrich	0.732	0.585	0.667	0.149	0.040	0.030	0.313	0.168	0.226	0.025	0.005	0.004
TaxBox	0.649	0.460	0.512	0.120	0.081	0.560	0.454	0.298	0.365	0.160	0.082	0.346
CoSTC	0.825	0.745	0.780	**0.529**	**0.680**	0.360	0.651	0.410	0.547	0.344	0.316	0.218
MRNBOX	**0.923**	**0.820**	**0.918**	0.506	0.640	**0.840**	**0.761**	**0.545**	**0.649**	**0.370**	**0.399**	**0.469**
Improv.	9.8%	7.5%	13.8%	-2.3%	-4%	2%	11%	13.5%	10.2%	2.6%	8.3%	12.3%

We accessed box embeddings via (top-k) on SemEval-Food and WordNet-Verb datasets, the results shown in Fig. 3. From Fig. 3, MRNBOX achieves optimal performance at k=1 for SemEval-Food (demonstrating precise box-based filtering), but requires k=2 for WordNet-Verb to balance recall and accuracy. Performance degrades when $k > 2$, especially in Hit@5 for SemEval-Food, indicating noise from excessive retention. This reveals a key trade-off: clear semantics favor aggressive pruning (k=1), while complex domains need conservative retention (k=2).

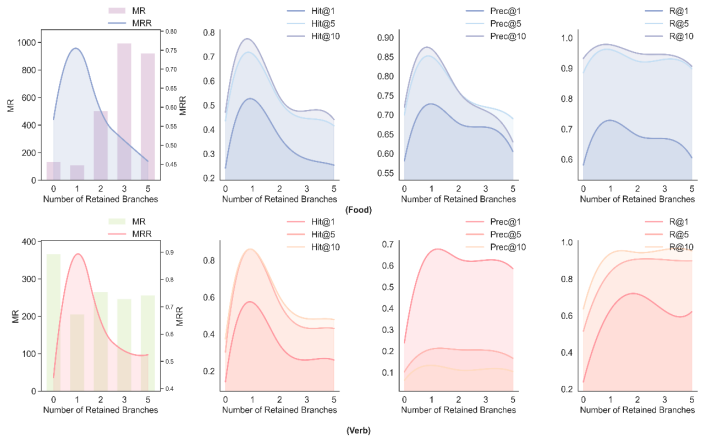

Figure 3: Top-k optimization in geometric box-based rearrangement and filtering

We evaluated MRNBOX across nine box-embedding dimensions (2–128) on the SemEval datasets using eleven metrics (MRR, Hit@k, Prec@k, R@k). Fig. 4 shows dimension-dependent variations with an optimal performance–efficiency tradeoff at moderate dimensions. Key patterns: (1) SemEval-Food peaks at 64d (4.2% Hit@1 degradation beyond this); (2) SemEval-Verb requires 128d/256d for optimal performance; (3) both achieve maximum recall (R@5/R@10) at their optimal dimensions. Declines beyond these points indicate overfitting, suggesting that mid-range dimensions (64d–128d) best balance capacity and generalization.

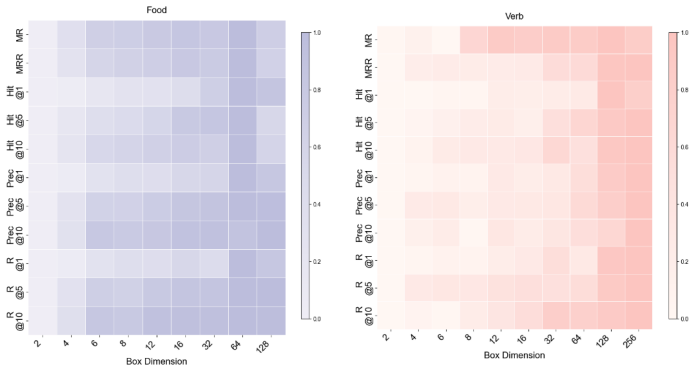

Figure 4: Optimal box dimension study under SemEval-Food and SemEval-Verb datasets

5 CONCLUSION

In this paper, we proposed MRNBOX and validated it through seven experiments over eleven metrics. Results show that MRNBOX effectively fuses heterogeneous semantic–structural features, differentiates node importance in graph representation learning, and alleviates the inefficiency of exhaustive candidate traversal caused by irrelevant or misleading positions. However, non-leaf node prediction remains limited: negative sampling bias concentrates candidates around leaf nodes, causing insufficient hard negatives and blurred decision boundaries. Other limitations include dependence on box embedding quality, limited APL benefits in shallow taxonomies, substantial overhead from hierarchical MoE, and challenges from concept noise and scalability on large or deep taxonomies. Future work will explore hierarchy-aware hard negative sampling with contrastive learning, improve noise robustness, optimize expert efficiency and routing, and extend MRNBOX to broader structural reasoning tasks such as knowledge graph completion and schema induction.

REFERENCES

Ines Arous, Ljiljana Dolamic, and Philippe Cudré-Mauroux. Taxocomplete: Self-supervised taxonomy completion leveraging position-enhanced semantic matching. In *Proceedings of the ACM Web Conference 2023*, pp. 2509–2518, 2023.

Simone D'Amico, Alessia De Santo, Mario Mezzanzanica, and Fabio Mercorio. Taxonomy expansion through collaborative llm mapping. In *Proceedings of the 40th ACM/SIGAPP Symposium on Applied Computing*, pp. 1961–1968, 2025.

Rangel Daroya, Aaron Sun, and Subhransu Maji. Task2box: Box embeddings for modeling asymmetric task relationships. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28827–28837, 2024.

Shib Dasgupta, Michael Boratko, Dongxu Zhang, Luke Vilnis, Xiang Li, and Andrew McCallum. Improving local identifiability in probabilistic box embeddings. *Advances in Neural Information Processing Systems*, 33:182–192, 2020.

Weibo Gao, Qi Liu, Linan Yue, Fangzhou Yao, Zhenya Huang, Zheng Zhang, and Rui Lv. Boxcd: Leveraging contrastive probabilistic box embedding for effective and efficient learner modeling. In *Proceedings of the ACM on Web Conference 2025*, pp. 3660–3671, 2025.

EunJeong Hwang, Jay Yoon Lee, Tianyi Yang, Dhruvesh Patel, Dongxu Zhang, and Andrew McCallum. Event-event relation extraction using probabilistic box embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 235–244, 2022.

Minhao Jiang, Xiangchen Song, Jieyu Zhang, and Jiawei Han. Taxoenrich: Self-supervised taxonomy completion via structure-semantic representations. In *Proceedings of the ACM Web Conference 2022*, pp. 925–934, 2022.

Song Jiang, Qiyue Yao, Qifan Wang, and Yizhou Sun. A single vector is not enough: Taxonomy expansion via box embeddings. In *Proceedings of the ACM Web Conference 2023*, pp. 2467–2476, 2023.

Zichen Liu, Hongyuan Xu, Yanlong Wen, Ning Jiang, Haiying Wu, and Xiaojie Yuan. Temp: taxonomy expansion with dynamic margin loss through taxonomy-paths. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pp. 3854–3863, 2021.

Saeed Masoudnia and Reza Ebrahimpour. Mixture of experts: a literature survey. *Artificial Intelligence Review*, 42:275–293, 2014.

Yuan Meng, Songlin Zhai, Yuxin Zhang, Zhongjian Hu, and Guilin Qi. Tef: Causality-aware taxonomy expansion via front-door criterion. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 8285–8294, 2025.

Johannes Messner, Ralph Abboud, and Ismail Ilkan Ceylan. Temporal knowledge graph completion using box embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7779–7787, 2022.

Yuhang Niu, Hongyuan Xu, Ciyi Liu, Yanlong Wen, and Xiaojie Yuan. Contrastive representation learning for self-supervised taxonomy completion. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pp. 6442–6450, 2024.

Geoffrey M Reed. Toward icd-11: Improving the clinical utility of who's international classification of mental disorders. *Professional Psychology: Research and Practice*, 41(6):457, 2010.

Bernard Rous. Major update to acm's computing classification system. *Communications of the ACM*, 55(11):12–12, 2012.

Jingchuan Shi, Hang Dong, Jiaoyan Chen, Zhe Wu, and Ian Horrocks. Taxonomy completion via implicit concept insertion. In *Proceedings of the ACM Web Conference 2024*, pp. 2159–2169, 2024.

Suyuchen Wang, Ruihui Zhao, Yefeng Zheng, and Bang Liu. Qen: Applicable taxonomy completion via evaluating full taxonomic relations. In *Proceedings of the ACM Web Conference 2022*, pp. 1008–1017, 2022.

Hongyuan Xu, Ciyi Liu, Yuhang Niu, Yunong Chen, Xiangrui Cai, Yanlong Wen, and Xiaojie Yuan. Tacoprompt: A collaborative multi-task prompt learning method for self-supervised taxonomy completion. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 15804–15817, 2023.

Hongyuan Xu, Yuhang Niu, Yanlong Wen, and Xiaojie Yuan. Compress and mix: Advancing efficient taxonomy completion with large language models. In *Proceedings of the ACM on Web Conference 2025*, pp. 4239–4249, 2025.

Zezhong Xu, Yincen Qu, Wen Zhang, Lei Liang, and Huajun Chen. Inbox: Recommendation with knowledge graph using interest box embedding. *arXiv preprint arXiv:2403.12649*, 2024.

Wei Xue, Yongliang Shen, Wenqi Ren, Jietian Guo, Shiliang Pu, and Weiming Lu. Insert or attach: Taxonomy completion via box embedding. *arXiv preprint arXiv:2305.11004*, 2023.

Shilin Yu and Yuantao Song. Ripple spreading algorithm: a new method for solving multi-objective shortest path problems with mixed time windows. *Complex & Intelligent Systems*, 10(2):2299–2325, 2024.

Yue Yu, Yinghao Li, Jiaming Shen, Hao Feng, Jimeng Sun, and Chao Zhang. Steam: Self-supervised taxonomy expansion with mini-paths. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1026–1035, 2020.

Qingkai Zeng, Jinfeng Lin, Wenhao Yu, Jane Cleland-Huang, and Meng Jiang. Enhancing taxonomy completion with concept generation via fusing relational representations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 2104–2113, 2021.

Jieyu Zhang, Xiangchen Song, Ying Zeng, Jiaze Chen, Jiaming Shen, Yuning Mao, and Lei Li. Taxonomy completion via triplet matching network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 4662–4670, 2021.

Yuchen Zhang, Amr Ahmed, Vanja Josifovski, and Alexander Smola. Taxonomy discovery for personalized recommendation. In *Proceedings of the 7th ACM international conference on Web search and data mining*, pp. 243–252, 2014.

Shaolin Zhu, Leiyu Pan, Dong Jian, and Deyi Xiong. Overcoming language barriers via machine translation with sparse mixture-of-experts fusion of large language models. *Information Processing & Management*, 62(3):104078, 2025.

A Preliminary

A.1 Problem Formulation

Definition 1. Taxonomy. A taxonomy is a hierarchical structure $T = (N, E)$, where N is a set of nodes representing terms and each directed edge $\langle n_p, n_c \rangle \in E$ denotes a hypernym-hyponym relationship such that n_p (parent) is a hypernym of n_c (child) and n_c is a hyponym of n_p.

Definition 2. Multi-view Fusion Node Embedding Representation $\mathbf{h}_n$.

Definition 3. Embedding Representation of Node Concept Description $\boldsymbol{x}_n^{(d)}$. It is a vector representation generated by pre-trained text encoding (such as CLIP) from the concept description text n_{desc} of a node.

Definition 4. Taxonomy Path P. Given a set of nodes $P = \{n_0, n_1, \ldots, n_k\}$, for each $i \in \{0, 1, \ldots, k-1\}$, there exists an edge $\langle n_i, n_{i+1} \rangle \in E$. k is the length of the path, n_0 is the start node, n_k is the end node, and P is called the path from n_0 to n_k.

Definition 5. Average Path Length of Node n, $\bar{l}_n$. Given a node n, let its ancestor nodes be a_i $(i = 1, 2, \ldots, k)$. Then, the average path length $\bar{l}_n$ is the average value of the path lengths from node n to all its ancestor nodes.

Definition 6. Topological association features embedding representation of node n in taxonomy, $x_n^{(p)}$. The topological correlation embedding representation $x_n^{(p)}$ of node n in the taxonomy is composed of the average embeddings after encoding the explicit topological association features. These descriptions are formed respectively by the current node n, its set of ancestor nodes $A(n)$, set of descendant nodes $D(n)$, and set of sibling nodes $S(n)$.

Definition 7. Ripple Set $G(u)$. Given a taxonomy $T = (N, E)$, the ripple set $G(u)$ captures the neighborhood structure of nodes hierarchically around a central node $u \in N_0$, with a maximum expansion to S layers. The 0-th layer ripple set is denoted as $R^{(0)}(u) = \{u\}$. The first-layer ripple set $R^{(1)}(u)$ includes one-hop neighbors and cousin nodes (nodes with the same parent). The s - th layer ripple set ($s \geq 2$) consists of the neighborhood nodes of the s - th layer. The final complete ripple set $G(u) = \bigcup_{s=1}^{S} R^{(s)}(u)$.

A.2 Mixture-of-Expert

MoE, built upon the Transformer architecture, primarily consists of two components: **Sparse MoE Layer** replaces the standard Feed-Forward Network (FFN) layer in conventional Transformer models (Zhu et al., 2025). It comprises multiple "expert" models, where each expert is an independent neural network. In practice, these experts are typically FFNs, though more complex network architectures are also possible.

Gating Network / Router This component determines which tokens are routed to which expert(s), allowing a single token to be sent to multiple experts. The token routing mechanism is a critical design aspect of MoE: the router is composed of learned parameters and is jointly pretrained with the rest of the network.

B Schematic Illustration of the Proposed Method

B.1 Multi-view Fusion Node Embedding Representation

Fig.5 shows our multi-view node embedding for $n \in N$, which fuses conceptual description d, topological association features s, and average path length p. The encoder extracts semantic vectors from d and s linearly projected to form the initial embedding $\mathbf{h}_n$.

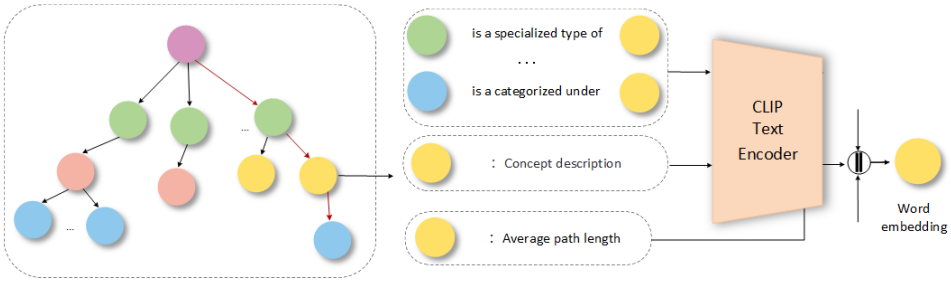

Figure 5: An example of multi-view fusion node embedding representation

B.2 Ripple Sets-based Graph Representation Learning

Ripple Set Construction. Ripple Set Construction resembles ripple spreading: the central node propagates influence layer by layer (1-hop → 2-hop → distant nodes), with signal strength attenuating at each hop (Fig. 6). This diffusion captures both local and global structures, as detailed in Algorithm 1.

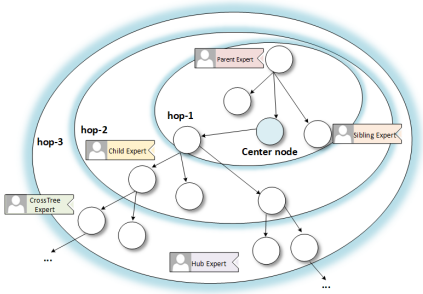

Figure 6: Ripple Diffusion in the Taxonomy

Algorithm 1 Ripple Sets-based Representation Learning

Input: Taxonomy graph $T = (N, E)$; central node u; max depth S; node features $\{h_v\}$
Output: Final embedding f_u
Initialize ripple sets: $\mathcal{R}^{(0)}(u) \leftarrow \{u\}$
for $s = 1$ to S **do**
    if $s = 1$ **then**
        $\mathcal{R}^{(1)}(u) \leftarrow$ direct neighbors and siblings of u
    else
        $\mathcal{R}^{(s)}(u) \leftarrow$ neighbors of $\mathcal{R}^{(s-1)}(u)$ excluding visited
    end if
end for
Aggregate ripple nodes: $G(u) \leftarrow \bigcup_{s=1}^{S} \mathcal{R}^{(s)}(u)$
for $s = 1$ to S **do**
    for each $v \in \mathcal{R}^{(s)}(u)$ **do**
        Compute expert scores $\{score_m^{(s)}(u, v)\}$ based on structural roles
        Compute expert weights $\alpha_m^{(s)}$ using gating network
        $final_score[u][v] \leftarrow \sum_m \alpha_m^{(s)} \cdot score_m^{(s)}(u, v)$
    end for
end for
$f_u \leftarrow$ AGGREGATE$(\{final_score[u][v] \cdot h_v \mid v \in G(u)\})$
return f_u

B.3 Box geometric filtering and re-ranking

In our box embedding taxonomy, parent nodes' boxes must fully contain all their children's boxes, with box sizes monotonically decreasing from the root (largest) to leaves. This geometric containment property enables an efficient pruning strategy (Algorithm 2): when a query fails the containment test with a root's direct child, we recursively prune the entire corresponding subtree, as the size hierarchy guarantees no downstream boxes could satisfy the containment relationship.

Algorithm 2 Box Geometry TC Filtering and Re-ranking

Input: Query box b_q; taxonomy tree T_0; root children set $\text{Root}_1 = \{r_i\}$; parameters α, β; top-k filter size k
Output: Pruned tree T_q; ranked candidate positions
for each $r_i \in \text{Root}_1$ **do**
    Compute similarity score $Score_i = Sim_{box}(b_{r_i}, b_q)$
end for
Select top-k root children: $\text{Root}_1^{top} \leftarrow TopK(\{Score_i\}, k)$
Prune tree: $T_q \leftarrow T_0 \setminus Descendants(\text{Root}_1 \setminus \text{Root}_1^{\text{top}})$
for each candidate position (p, c) in T_q **do**
    Compute $s_1 = Sim_{box}(b_p, b_q)$, $s_2 = Sim_{box}(b_q, b_c)$
    if c is a leaf node **then**
        $Trans \leftarrow$ structure-adjusted weight based on centroid distance
    else
        $Trans \leftarrow 1$
    end if
    Compute final score: $Score(q, p, c) \leftarrow s_1 \cdot s_2 \cdot Trans$
end for
return T_q and ranked positions based on $Score(q, p, c)$

C OTHER EXPERIMENTS

C.1 CASE STUDY

As shown in Table 5, MRNBOX exhibits three key strengths in the food taxonomy. First, it generalizes new concepts well, accurately placing *caster sugar* under the *powdered sugar* subtree, reflecting semantic awareness of their variant relationship. Second, it models node importance effectively, as seen in the prediction of *cheddar cheese*, where *cheese* ranks first, *hard cheese* second, and the broader category *dairy products* appears as a reasonable hypernym. Third, its pruning capability efficiently narrows the space for *almond milk* to the plant-based milk subtree, where *non-dairy beverages* and *nut milks* provide semantically valid alternatives. Even under loose supervision, the model captures functional associations, e.g., linking *sweetening* to *(flavorer, honey)*. However, random negative sampling at the non-leaf level reveals structural limitations. In *western omelet*, the lack of boundary constraints leads to confusion with *(candy, praline)*. For *green tea*, missing sibling contrast causes *fruit juices* to wrongly outrank *tea*. In *whole wheat bread*, predictions are confined to *bakery products*, but *sweet breads* is incorrectly ranked top due to blurred boundaries with healthy staples. These cases highlight the need for more structured sampling to preserve sibling exclusivity and hierarchical coherence.

Table 5: Case Study Highlighting Model Capabilities and Limitations (SemEval-Food)

Query and Position	TOP3 Predictions	Correct?
Q: cator sugar **P:** powered sugar	powered sugar; pinole; granulated sugar soup; scotch	✓
Q: sweetening **P:** flavorer, honey	flavorer; honey; flavore foodstuff; flavorer; sugar	✓
Q: western omelet **P:** firm omelet	candy; praline; soupscotch casserole; soupscotch	×
Q: almond milk **P:** plant-based milk	plant-based milk; non-dairy beverages; nut milks non-dairy beverages; nut milks	✓
Q: cheddar cheese **P:** cheese	cheese; hard cheese; dairy products hard cheese; dairy products	✓
Q: whole wheat bread **P:** whole wheat bread rolls	bakery products; sweet breads; cinnamon rolls bakery products; sticky buns	×
Q: green tea **P:** tea, green tea varieties	beverages; fruit juices; orange juice varieties fruit juices; apple juice varieties	×

C.2 Parameter Sensitivity Analysis

Fig. 7 shows MRNBOX's ripple layers (1-3 hops) performance datasets via polar visualization, revealing optimal propagation depths: (1) Simpler datasets SemEval (Food/Verb) peak at 2-hop, while complex Wab datasets (Psy/CS) achieve best performance at 3-hop (notably 0.963 R@10 for Wab-CS). The results demonstrate a clear trade-off, with 1-hop preserving local accuracy while deeper hops (2-3) significantly improving global coverage metrics, indicating MRNBOX's adaptive balance of local precision and global structural awareness.

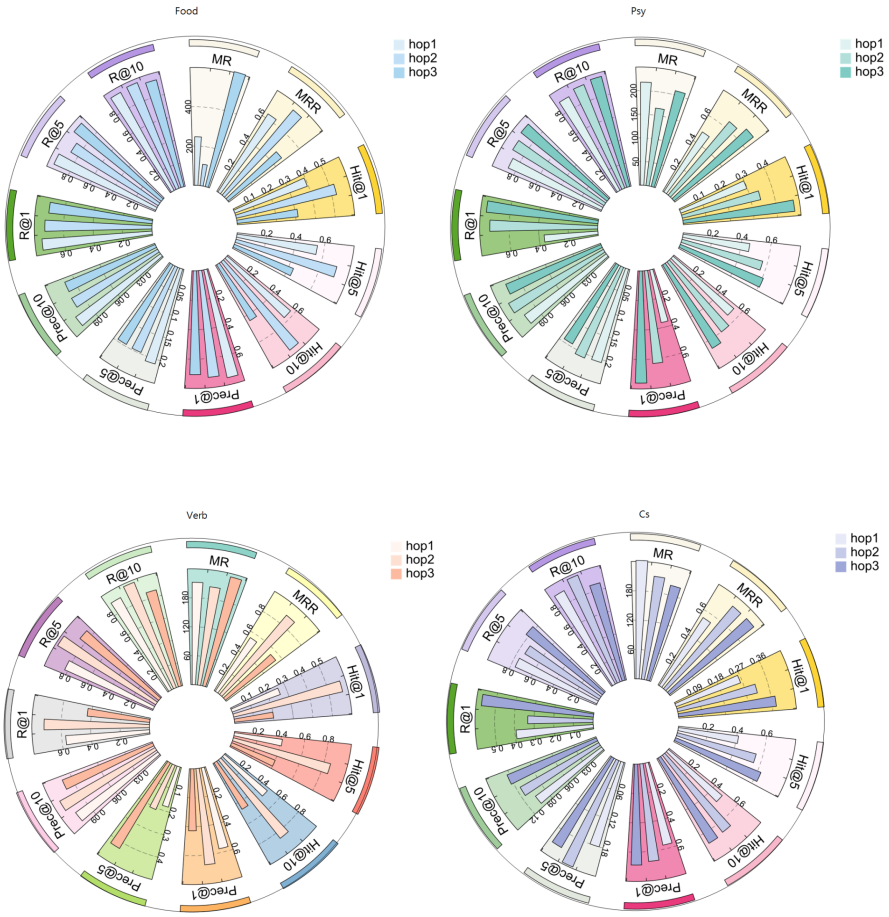

Figure 7: Four data sets ripple layers affect the performance of model

D Structural Analysis of Dates

The depth distribution shows that MAG-PSY exhibits a relatively shallow hierarchy (maximum depth of 5 and an average depth of only 3.14). In contrast, the sibling count distribution reveals an extremely heavy-tailed pattern, with some nodes having over 1700 siblings. This indicates that the primary challenge of MAG-PSY is not hierarchy depth but the severe "candidate explosion" caused by a large number of same-level alternatives. We further clarify in the main text how these structural characteristics relate to our model design: ripple diffusion effectively captures local subtree structures in shallow hierarchies, while the MoE sibling expert and the box-based geometric exclusion mechanism help mitigate noise and confusion introduced by large sibling sets.

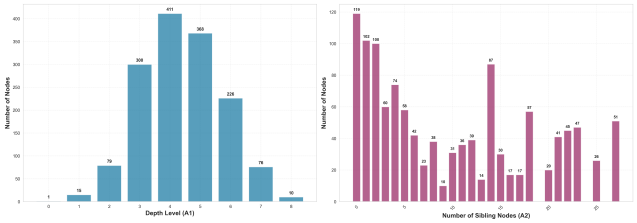

Figure 8: Depth and Breadth Distributions in WN-FOOD

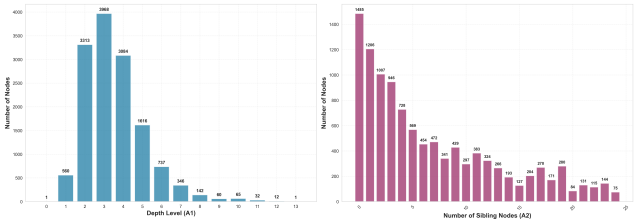

Figure 9: Depth and Breadth Distributions in WN-VERB.

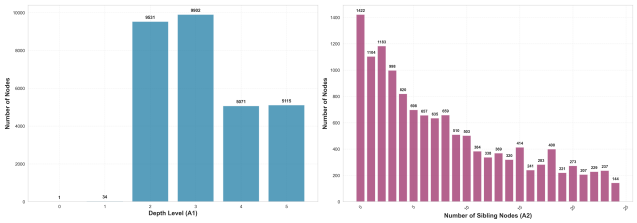

Figure 10: Depth and Breadth Distributions in MAG-CS.

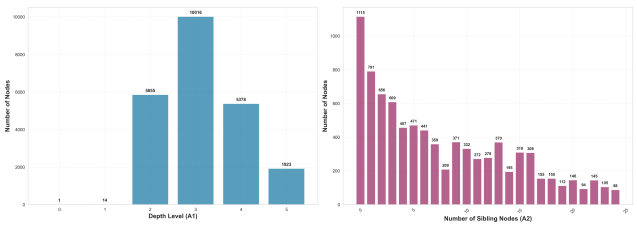

Figure 11: Depth and Breadth Distributions in MAG-PSY.