

A Single Vector Is Not Enough: Taxonomy Expansion via Box Embeddings (Extended Abstract)*

Song Jiang¹, Qiyue Yao¹, Qifan Wang² and Yizhou Sun¹

¹University of California, Los Angeles,

²Meta AI

{songjiang, yzsun}@cs.ucla.edu, qiyueyao@g.ucla.edu, wqfcr@meta.com

Abstract

Taxonomies support various practical web applications such as product navigation in online shopping and user profile tagging on social platforms. Most existing methods for expanding taxonomies encode entities into vector embeddings (i.e., single points). However, we argue that vectors are insufficient to model the “is-a” hierarchy in taxonomy (*asymmetrical* relation), because two points can only represent pairwise similarity (*symmetrical* relation). To this end, we propose to project taxonomy entities into *boxes* (i.e., hyperrectangles). Two boxes can be “contained”, “disjoint” and “intersecting”, thus naturally representing an asymmetrical taxonomic hierarchy. Upon box embeddings, we propose a novel model BOXTAXO for taxonomy expansion. The core of BOXTAXO is to learn boxes for entities to capture their child-parent hierarchies. Extensive experiments on two benchmarks demonstrate the effectiveness of BOXTAXO compared to vector based models.

1 Introduction

A taxonomy is a schema of hierarchical classification, which is used to organize conceptual entities into a tree-like structure according to their semantics. Taxonomies have been widely adopted to support various web services because of the effectiveness of indexing and organizing knowledge. For example, Amazon has a product taxonomy to facilitate online shopping [Mao *et al.*, 2020], and Pinterest uses taxonomy to enhance content understanding and recommendation [Gonçalves *et al.*, 2019; Manzoor *et al.*, 2020]. Many taxonomies were initially curated by domain experts, however, due to the constant and rapid growth of new concepts, automatically expanding existing taxonomies with these new entities is necessary to avoid their obsolescence. For consistency with existing literature, we follow [Shen *et al.*, 2020; Yu *et al.*, 2020] and refer to a child as *query*, and a parent as *anchor*. The terms are interchangeable throughout this paper.

Existing approaches for taxonomy expansion focus on capturing the child-parent hierarchies. Early efforts learn the hierarchies by exploiting the semantic relatedness between two entities. The semantics can be represented by lexical patterns [Snow *et al.*, 2004; Hearst, 1992] or later the more powerful distributional word embeddings [Mikolov *et al.*, 2013; Chang *et al.*, 2018; Fu *et al.*, 2014]. Beyond semantics, recent works have further explicitly modeled the tree structure of taxonomy. They use various structural summaries, including paths [Yu *et al.*, 2020; Liu *et al.*, 2021; Jiang *et al.*, 2022] and local graphs [Wang *et al.*, 2021; Shen *et al.*, 2020; Mao *et al.*, 2020], as additional signals to enhance the learning of child-parent hierarchies. Representing hierarchies is more in line with the geometric properties of hyperbolic space [Ganea *et al.*, 2018; Nickel and Kiela, 2017]. Several works [Aly *et al.*, 2019; Ma *et al.*, 2021] model the child-parent relations by learning hyperbolic representations.

The core methodology of most aforementioned approaches is to learn *vector* embeddings for entities in taxonomy. The child-parent relation is then inferred by computing the relatedness of a pair of entities upon their vector embeddings. However, the vector embeddings, i.e., points in geometric space, can only represent the pairwise similarity, which is a *symmetrical* relation (Similarity is usually measured by distance – either Euclidean or geodesic– of two points). The taxonomic child-parent hierarchies, on the contrary, are naturally *asymmetrical*. Therefore, the vector based embeddings are *not* sufficient to represent the hierarchies in taxonomy, limiting their effectiveness in taxonomy expansion.

To overcome this insufficiency, instead of vectors, we propose to use *boxes* to represent the entities in the taxonomy. A box is an axis-aligned hyperrectangle in geometric space, which can be characterized by two points. Unlike a single point, the benefit of a box is that box has a geometric region, which enables it to represent the more complicated asymmetrical pairwise relations such as “enclose”, “disjoint” and “intersect”. Specifically, a child box is entirely enclosed inside its parent box (e.g., “Graph Neural Network” and “Machine learning”). Two entities are fully separated if they are not in a child-parent hierarchy (e.g., “Programming Language” and “Machine learning”). The boxes of two entities overlap if they share common children in taxonomy (e.g., “Computer Vision” and “Machine learning”).

*Jiang, et al. "A single vector is not enough: Taxonomy expansion via box embeddings." 2023.

Despite the natural and intuitive representation of taxonomic hierarchies, the box embeddings for taxonomy expansion still face three main challenges. First, limited taxonomy annotation is available for new entities, making it difficult to learn accurate boxes and infer their positions in the taxonomy in a supervised manner. Second, most existing box embeddings approaches optimize boxes by capturing probabilistic properties, which have proven difficult to train in practice [Li *et al.*, 2018; Vilnis *et al.*, 2018]. The reason is box pairs that are supposed to “enclose” or “intersect”, but are wrongly disjoint during training, will never be corrected because the gradients from the probabilistic loss function are zero in this case. [Li *et al.*, 2018; Dasgupta *et al.*, 2020] mitigate this issue by representing the edges of boxes as probabilistic density distributions, i.e., making the box “soft”. However, such “soft” boxes lose the intuitive interpretability of normal “hard” boxes. Third, different from reasoning in the existing structure, taxonomy expansion requires learning boxes for new entities. Therefore, a desired model should be generalizable, which is able to generate box embeddings compatible with existing taxonomies for new entities.

In this paper, we propose BOXTAXO, a self-supervised model that expands taxonomy with box embeddings. With self-supervised learning, our model does not require annotated labels, but creates training samples from the existing taxonomy. Specifically, each (child, parent) pair in the existing taxonomy is treated as a positive sample. The entities that are not the ancestors of each child are collected as negative samples. To optimize the box embeddings, we propose a joint loss function that guides the boxes to capture the taxonomic hierarchies from both the geometric view and the probabilistic view. The joint view loss function can avoid the gradient missing issue mentioned above and still ensure the boxes are intuitive and interpretable to humans. The box embeddings are encoded via a pre-trained language model to ensure the generalizability to new entities. At inference time, box embeddings offer an easy and natural way to find an appropriate anchor for a query, by checking whether the box of a candidate parent fully contains the box of the query. We implement this from the probabilistic view in BOXTAXO.

Our **main contributions** are summarized as follows: 1) We propose to use box embeddings for taxonomy expansion, which can accurately represent the hierarchies in taxonomy. 2) We develop a self-supervised model that optimizes the box embeddings through joint learning of geometry and probability. 3) We conduct an extensive set of experiments on two real-world taxonomies. Experimental results demonstrate the effectiveness of BOXTAXO compared to vector based representations. We also provide various ablation studies and analyses to understand how BOXTAXO works.

2 Proposed Method: BOXTAXO

2.1 Box Projection

The nodes in taxonomy are conceptual entities. To represent each entity as a box in the latent space, BOXTAXO uses a two-stage projection process. In specific, an entity is first encoded as numeric embeddings from natural language via an *entity*

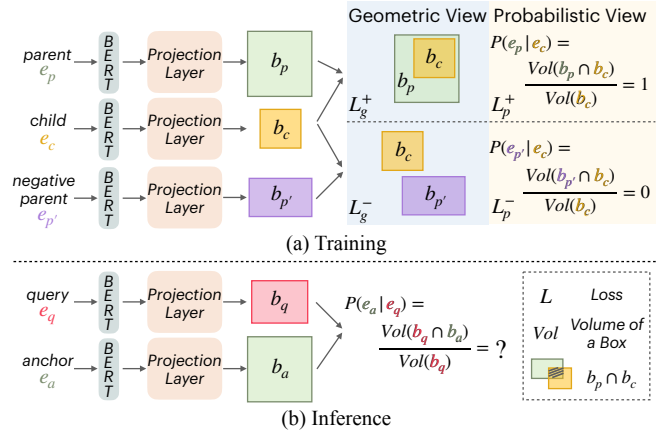


Figure 1: The overview of BOXTAXO. The entities in taxonomy are first projected to boxes based on Bert. (a) Training: the box embeddings are optimized from a joint view of geometry and probability, in order to accurately represent the taxonomic hierarchies. (b) Inference: check whether a query’s box is enclosed by the candidate anchor’s box in a probabilistic way. Note that the boxes shown in this figure are 2D, but they can be in higher dimension spaces, i.e., hyperrectangles.

encoder, and then converted into a rectangular lattice by a *box projector*. We now introduce these two operators in detail.

Entity Encoder. The pre-trained language models (PTMs) have shown promising achievements in many natural language tasks [Qiu *et al.*, 2020]. Encouraged by their success, we use PTMs to encode the entities into embeddings. Without loss of generality, we use Bert [Devlin *et al.*, 2019] as the entity encoder in this paper. Formally, for the i -th entity e_i , Bert converts it into k -dimensional representation $n_i \in \mathbb{R}^k$:

$$n_i = \text{Bert}(e_i). \quad (1)$$

The entities in taxonomy are usually curated and thus could have definition sentences. Therefore, for such a definition sentence s_i , we concatenate it with its entity e_i and build the input of Bert as “[CLS] e_i, s_i [SEP]”. We then use the output embeddings of “[CLS]” in the final Bert layer as the representation n_i of entity e_i . The representation n_i encodes the contextual semantics of the entity. Please note other pre-trained language models, such as Roberta [Liu *et al.*, 2019] and ELECTRA [Clark *et al.*, 2020], are flexible to be replaced as the encoder.

Box Projector. We then project the entity representation n_i into box embeddings. A box can be defined by two points (i.e., vectors). We therefore use the center point $c_i \in \mathbb{R}^d$ and the offset vector $o_i \in \mathbb{R}^d$ to represent a box $b_i \in \mathbb{R}^d$, i.e., $b_i = (c_i, o_i)$, where d is the dimension of box embeddings. Note that c_i and o_i are just two vector embeddings. To represent an entity e_i as box embeddings b_i , we project the entity representation n_i into the center c_i and offset o_i , separately. Specifically, because this projection is only a dimension transformation between two embeddings, we simply use two multilayer perceptrons (MLPs) as the projectors:

$$c_i = \text{MLP}_c(n_i), \quad o_i = \text{MLP}_o(n_i), \quad (2)$$

where MLP_c and MLP_o are the projection layers for center c_i and offset o_i , respectively. To ensure the learned box b_i is a valid rectangle, we further apply an exponential operator to the offset o_i , so that every dimension of o_i is guaranteed to be larger than 0.

2.2 Box Training

We now seek to optimize the box embeddings such that they can accurately represent the taxonomic hierarchies, i.e., the child-parent relations. Because each $\langle \text{child}, \text{parent} \rangle$ pair in the taxonomy is a natural “label”, we propose to fine-tune the entity encoder and box projector in a self-supervised manner. Specifically, we utilize all the immediate $\langle \text{child}, \text{parent} \rangle$ pairs in the taxonomy as positive samples. Negative samples have been demonstrated to be crucial in optimizing box embeddings [Lees *et al.*, 2020]. Therefore, for each child node in such a pair, we collect its “siblings”, “uncles” and “cousins” as the negative samples against the child-parent relations. Compared to vectors, box embeddings are more powerful in representing child-parent relations. We show how boxes achieve this advantage from two views: the *geometric* view and the *probabilistic* view. Accordingly, we design two training objective functions, the geometric loss and the probabilistic loss, to jointly optimize the box embeddings.

Geometric View. We first show how the child-parent relation can be represented with box embeddings in geometric language. A box with d -dimensional center and offset vectors is a d -dimensional hyperrectangle in Euclidean space. A $\langle \text{child}, \text{parent} \rangle$ pair can be semantically interpreted as “child *is-a* parent” or “child *is-one-of* parent” [Hearst, 1992]. Therefore, we let the child hyperrectangle be fully *enclosed* by the parent hyperrectangle, indicating the child entity is one kind of parent. Formally, for a d -dimensional child box $b_c = (c_c, o_c)$, since o_c is regularized to be positive, we denote by $l_c = c_c - o_c$ and $r_c = c_c + o_c$ the minimum and maximum corner points of the hyperrectangle, respectively. Similarly, for parent box $b_p = (c_p, o_p)$, denote by $l_p = c_p - o_p$ and $r_p = c_p + o_p$ the minimum and maximum corner points. Then the “enclose” relation has:

$$l_c^i \geq l_p^i, \quad r_c^i \leq r_p^i, \quad \forall i \in \{1, 2, \dots, d\}, \quad (3)$$

where i denotes the i -th dimension of the embeddings. We derive a loss function L_g^+ to ensure boxes satisfy this geometric enclose relation (Eq. (3)) for pair $\langle e_c, e_p \rangle$, formalized as:

$$L_g^+ = \frac{1}{d} \left[\sum_{i=1}^d \max(0, l_p^i - l_c^i + \delta) + \sum_{i=1}^d \max(0, r_c^i - r_p^i + \delta) \right], \quad (4)$$

where δ is a hyper-parameter across all d dimensions that controls the geometric margin between the child and parent boxes.

Oppositely, for a negative pair $\langle \text{child}, \text{parent}' \rangle$, denoted by $\langle e_c, e_{p'} \rangle$, the child hyperrectangle should be *disjoint* with the negative parent hyperrectangle. We implement this “disjoint” relationship by enforcing the intersection between the child box and the negative parent box to be empty. Formally, for such a box pair $\langle b_c, b_{p'} \rangle$, their intersection $b_z = b_c \cap b_{p'}$ is

formalized as:

$$l_z = \max(l_c, l_{p'}), \quad r_z = \min(r_c, r_{p'}). \quad (5)$$

An empty intersection, i.e., $b_z = \emptyset$, essentially means every dimension of the intersection b_z is less than or equal to 0. Based on this property, we derive a loss function L_g^- to minimize the offset o_z of the intersection, formalized as:

$$L_g^- = \frac{1}{d} \sum_{i=1}^d (o_z^i - \epsilon)^2, \quad (6)$$

where ϵ is a hyper-parameter to adjust the margin of intersection. If $\epsilon > 0$, we allow some intersection between two boxes, and when $\epsilon \leq 0$, we force the two boxes to be separated. Note that the offset can be derived by $o_z = (r_z - l_z)/2$.

Probabilistic View. We now introduce how the child-parent relation is represented by box embeddings from a probabilistic perspective. We first define taxonomic probability: 1) (Taxonomic Probability) Taxonomic probability $P(e_y | e_x)$ is the likelihood of event “from a given entity e_x , another entity e_y can be reached along a given 1-length edge” occurring. For a $\langle \text{child}, \text{parent} \rangle$ pair $\langle e_c, e_p \rangle$ in taxonomy, the taxonomic probability $P(e_p | e_c) = 1$, because given a child, its exact parent can always be retrieved along the edge connecting them. If a child has multiple parents, we define the taxonomic probability as 1 for all parents. Similarly, for a negative pair $\langle \text{child}, \text{parent}' \rangle$, denoted by $\langle e_c, e_{p'} \rangle$, since the negative parent can not be directly reached given the child, the taxonomic probability $P(e_{p'} | e_c) = 0$. Desired box embeddings should satisfy these conditions of taxonomic probability for both positive and negative pairs, so that they can accurately represent the child-parent hierarchies in taxonomy.

Similar to using diagrams of sets to describe probabilities (i.e., Venn diagram [Venn, 1880]), box embeddings provide a natural graphical way to calculate the taxonomic probability. Following [Vilnis *et al.*, 2018; Li *et al.*, 2018; Onoe *et al.*, 2021], we use the volume of the intersection between child box and parent box, divided by the volume of child box, to represent the taxonomic probability $P(e_p | e_c)$, formalized as:

$$P(e_p | e_c) = \frac{\text{Vol}(b_p \cap b_c)}{\text{Vol}(b_c)}, \quad (7)$$

where the $\text{Vol}(\cdot)$ is the volume of a box. On this basis, we propose a probability loss function for each positive child-parent pair $\langle e_c, e_p \rangle$, denoted by L_p^+ , which is formalized as:

$$L_p^+ = (P(e_p | e_c) - 1)^2, \quad (8)$$

and also a probability loss function for each negative pair $\langle e_c, e_{p'} \rangle$, denoted by L_p^- , which is formalized as:

$$L_p^- = (P(e_{p'} | e_c) - 0)^2. \quad (9)$$

Box Regularization. In both geometric and probabilistic views, we design loss functions that minimize the intersection of two negative box embeddings, i.e., the negative geometric loss L_g^- and the negative probabilistic loss L_p^- . Actually, if a box is near zero in all its embedding dimensions, or its volume is close to zero, these two losses are also able to be minimized. In this case, however, the learned box embeddings are

meaningless and can hardly represent the taxonomic hierarchies. To avoid this “cheating” during training, we regularize that box embeddings can not be too small in all dimensions. For box embeddings b_e of entity e , we implement this constraint by regularizing the offset o_e with regularization loss L_r :

$$L_r = \frac{1}{d} \sum_{i=1}^d \min(0, o_e^i - \phi)^2, \quad \forall i \in \{1, 2, \dots, d\}. \quad (10)$$

ϕ controls the minimum length of boxes in each dimension.

Joint Loss. Finally, we combine the geometric losses, the probabilistic losses and the regularization loss to jointly train the model. Formally, the final loss function is:

$$L = \alpha(L_g^+ + L_g^-) + \beta(L_p^+ + L_p^-) + \gamma L_r, \quad (11)$$

where the α , β and γ are hyper-parameters to control the contributions of each single loss function.

2.3 Inference with Box

During inference, our goal is to find an appropriate parent entity, i.e., an anchor, from the taxonomy for a given new query. In contrast to vector embeddings that measure the distance between two points, box embeddings are more intuitive and natural to determine whether a candidate parent is suitable: by checking to what extent the box of anchor encloses the box of the query. We implement this idea in a probabilistic way as shown in Fig. 1 (b). Specifically, for query e_q , we first project it into a box b_q and then compare it with each candidate anchor e_a ’s box b_a ¹. Formally, we rank the candidates by their taxonomic probabilities $P(e_a|e_q)$. A higher $P(e_a|e_q)$ indicates that anchor e_a is more likely to be an appropriate parent for the query e_q . In some cases, the taxonomic probability values of many anchors could be the same. For example, if the query box is enclosed by the box of a leaf anchor node, then it is also enclosed by all ancestors (until the root) of this leaf anchor, i.e., $P(e_a|e_q) = 1$ for the leaf anchor and all its ancestors. In this case, we return this leaf anchor as the predicted parent, since it is the finest-grained and more precisely describes the query. We note this leaf anchor should have the smallest volume because it is enclosed by all ancestors. Therefore, for candidate anchors with the same taxonomic probability, we perform a second ranking according to the volume of their boxes, so that finer-grained anchors can be placed higher.

3 Experiments

We compare BOXTAXO with vector based embeddings baselines for taxonomy expansion and report the results in Table. 1. We include two lines of baselines: 1) Because BOXTAXO only models the simple (child, parent) pairs during training, we first compare BOXTAXO with vector based counterparts that also focus on such pairs, i.e., TAXI, HypeNet and Bert+MLP. BOXTAXO outperforms them with significant gains, indicating the effectiveness of box embeddings against vectors for taxonomy expansion. 2) We also compare

BOXTAXO with vector based baselines that use advanced structural summaries, including local graphs (TaxoExpan) and paths (STEAM). Despite not explicitly modeling such structural signals, BOXTAXO still achieves a clear improvement over TaxoExpan and shows comparable results with STEAM. We are encouraged by these results as it shows the potential to facilitate box embeddings with advanced structures to further boost taxonomy expansion.

Dataset	Environment			Science		
Metric	ACC	MRR	Wu&P	ACC	MRR	Wu&P
TAXI	16.7	N/A	44.7	13.0	N/A	32.9
HypeNet	16.7	23.7	55.8	15.4	22.6	50.7
Bert+MLP	11.1	21.5	47.9	11.5	15.7	43.6
TaxoExpan	11.1	32.3	54.8	27.8	44.8	57.6
STEAM	<u>36.1</u>	<u>46.9</u>	<u>69.6</u>	36.5	48.3	68.2
BOXTAXO	38.1	47.1	75.4	<u>31.8</u>	<u>45.3</u>	<u>64.7</u>

Table 1: Results of BOXTAXO on taxonomy expansion compared to vector based methods. We use the same experimental setting as [Yu *et al.*, 2020] and the baseline results are from [Yu *et al.*, 2020]. We report the averages of ten runs of BOXTAXO. The best results are in boldface, and the second-best results are underlined. The “N/A” indicates that MRR is not applicable to TAXI.

Acknowledgments

The authors express their gratitude to Jie Tang, Qingyang Zhong, Jifan Yu, and Kewei Cheng for their insightful discussions. This work was partially supported by NSF 2211557, NSF 1937599, NSF 2119643, NASA, SRC, Okawa Foundation Grant, Amazon Research Awards, Cisco research grant, Picsart Gifts, and Snapchat Gifts.

References

- [Aly *et al.*, 2019] Rami Aly, Shantanu Acharya, Alexander Ossa, Arne Köhn, Chris Biemann, and Alexander Panchenko. Every child should have parents: A taxonomy refinement algorithm based on hyperbolic term embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4811–4817, Florence, Italy, July 2019. Association for Computational Linguistics.
- [Chang *et al.*, 2018] Haw-Shiuan Chang, Ziyun Wang, Luke Vilnis, and Andrew McCallum. Distributional inclusion vector embedding for unsupervised hypernymy detection. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 485–495, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
- [Clark *et al.*, 2020] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

¹Here we use all entities in the existing taxonomy as the candidate anchor set.

- [Dasgupta *et al.*, 2020] 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, 2020.
- [Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [Fu *et al.*, 2014] Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1199–1209, 2014.
- [Ganea *et al.*, 2018] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic entailment cones for learning hierarchical embeddings. In *International Conference on Machine Learning*, pages 1646–1655. PMLR, 2018.
- [Gonçalves *et al.*, 2019] Rafael S Gonçalves, Matthew Horridge, Rui Li, Yu Liu, Mark A Musen, Csongor I Nyulas, Evelyn Obamos, Dhananjay Shrouthy, and David Temple. Use of owl and semantic web technologies at pinterest. In *International Semantic Web Conference*, pages 418–435. Springer, 2019.
- [Hearst, 1992] Marti A Hearst. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 14th International Conference on Computational Linguistics*, 1992.
- [Jiang *et al.*, 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*, pages 925–934, 2022.
- [Lees *et al.*, 2020] Alyssa Lees, Chris Welty, Shubin Zhao, Jacek Korycki, and Sara Mc Carthy. Embedding semantic taxonomies. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1279–1291, 2020.
- [Li *et al.*, 2018] Xiang Li, Luke Vilnis, Dongxu Zhang, Michael Boratko, and Andrew McCallum. Smoothing the geometry of probabilistic box embeddings. In *International Conference on Learning Representations*, 2018.
- [Liu *et al.*, 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv:1907.11692*, 2019.
- [Liu *et al.*, 2021] 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*, pages 3854–3863, 2021.
- [Ma *et al.*, 2021] Mingyu Derek Ma, Muhao Chen, Te-Lin Wu, and Nanyun Peng. HyperExpan: Taxonomy expansion with hyperbolic representation learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.
- [Manzoor *et al.*, 2020] Emaad Manzoor, Rui Li, Dhananjay Shrouthy, and Jure Leskovec. Expanding taxonomies with implicit edge semantics. In *Proceedings of The Web Conference 2020*, pages 2044–2054, 2020.
- [Mao *et al.*, 2020] Yuning Mao, Tong Zhao, Andrey Kan, Chenwei Zhang, Xin Luna Dong, Christos Faloutsos, and Jiawei Han. Octet: Online catalog taxonomy enrichment with self-supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2247–2257, 2020.
- [Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.
- [Nickel and Kiela, 2017] Maximillian Nickel and Douwe Kiela. Poincaré embeddings for learning hierarchical representations. *Advances in neural information processing systems*, 30, 2017.
- [Onoe *et al.*, 2021] Yasumasa Onoe, Michael Boratko, Andrew McCallum, and Greg Durrett. Modeling fine-grained entity types with box embeddings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Online, August 2021. Association for Computational Linguistics.
- [Qiu *et al.*, 2020] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897, 2020.
- [Shen *et al.*, 2020] Jiaming Shen, Zhihong Shen, Chenyan Xiong, Chi Wang, Kuansan Wang, and Jiawei Han. Taxoexpan: Self-supervised taxonomy expansion with position-enhanced graph neural network. In *Proceedings of The Web Conference 2020*, 2020.
- [Snow *et al.*, 2004] Rion Snow, Daniel Jurafsky, and Andrew Ng. Learning syntactic patterns for automatic hypernym discovery. *Advances in neural information processing systems*, 17, 2004.
- [Venn, 1880] John Venn. I. on the diagrammatic and mechanical representation of propositions and reasonings. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 10(59):1–18, 1880.
- [Vilnis *et al.*, 2018] Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. Probabilistic embedding of

knowledge graphs with box lattice measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[Wang *et al.*, 2021] Suyuchen Wang, Ruihui Zhao, Xi Chen, Yefeng Zheng, and Bang Liu. Enquire one’s parent and child before decision: Fully exploit hierarchical structure for self-supervised taxonomy expansion. In *Proceedings of the Web Conference 2021*, pages 3291–3304, 2021.

[Yu *et al.*, 2020] 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*, pages 1026–1035, 2020.