

QuanTaxo: A Quantum Approach to Self-Supervised Taxonomy Expansion

Sahil Mishra¹, Avi Patni², Niladri Chatterjee², Tanmoy Chakraborty¹

¹Department of Electrical Engineering, Indian Institute of Technology Delhi

²Department of Mathematics, Indian Institute of Technology Delhi

sahil.mishra@ee.iitd.ac.in, patni.avi@gmail.com, {niladri, tanchak}@iitd.ac.in

Abstract

A taxonomy is a hierarchical graph of knowledge that provides valuable insights for various web applications. However, the manual construction of taxonomies requires significant human effort. As web content continues to expand at an unprecedented pace, existing taxonomies risk becoming outdated, struggling to incorporate new and emerging information effectively. As a consequence, there is a growing need for dynamic taxonomy expansion to keep them relevant and up-to-date. Existing taxonomy expansion methods often rely on classical word embeddings to represent entities. However, these embeddings fall short of capturing hierarchical polysemy, where an entity’s meaning can vary based on its position in the hierarchy and its surrounding context. To address this challenge, we introduce *QuanTaxo*, a quantum-inspired framework for taxonomy expansion that encodes entities in a Hilbert space and models interference effects between them, yielding richer, context-sensitive representations. Comprehensive experiments on five real-world benchmark datasets show that *QuanTaxo* significantly outperforms classical embedding models, achieving substantial improvements of 12.3% in accuracy, 11.2% in Mean Reciprocal Rank (MRR), and 6.9% in Wu & Palmer (Wu&P) metrics across nine classical embedding-based baselines.

Code: <https://github.com/sahilmishra0012/QuanTaxo>

1 Introduction

Taxonomy is a hierarchically structured knowledge graph designed to portray the hypernymy (“is-a”) relationship between concepts, showing how broad categories subsume more specific ones. Therefore, it serves as core infrastructure for many online applications by efficiently indexing and organizing knowledge. For example, Amazon (Mao et al. 2020) and Alibaba (Karamanolakis, Ma, and Dong 2020) use taxonomies to power e-commerce search and browsing, while Pinterest leverages them for content recommendation and advertisement targeting (Gonçalves et al. 2019; Manzoor et al. 2020).

Traditional taxonomies are typically constructed from scratch by domain experts, making the process slow and labor-intensive. Early automation efforts used unsupervised

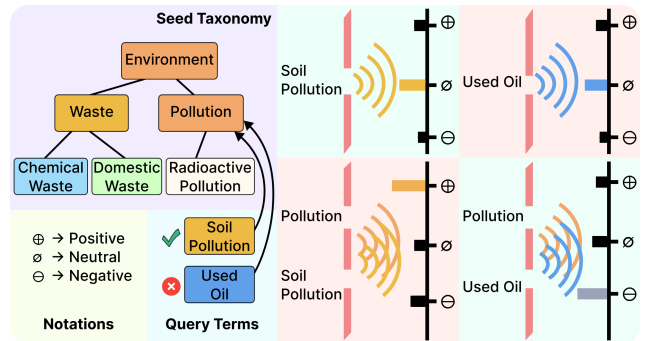


Figure 1: An illustration of taxonomy expansion using *QuanTaxo* on the *Environment* seed taxonomy, where the query terms *Soil Pollution* and *Used Oil* are to be inserted.

methods such as graph pruning (Velardi, Faralli, and Navigli 2013), hierarchical clustering (Zhang et al. 2018a), and topic modeling (Liu et al. 2012; Wang et al. 2013) to induce taxonomies from raw data, but they often fail to match the structure and coherence of expert-designed taxonomies. As data continuously evolves, there is a growing need to incorporate new concepts into existing taxonomies. To address this, we focus on *taxonomy expansion* — the task of integrating new concepts, or *query nodes*, into an existing *seed taxonomy* by placing them under suitable *anchor nodes*. For example, as shown in Fig. 1, *Soil Pollution* can be accurately placed under *Pollution* to maintain structural consistency.

Early work on taxonomy expansion used self-supervision, relying on lexical patterns or distributional embeddings to learn parent-child relationships from a small seed taxonomy (Jurgens and Pilehvar 2015; Snow, Jurafsky, and Ng 2004). However, these methods struggled with limited data and underused structural information. More recent approaches improve this by using paths (Liu et al. 2021; Jiang et al. 2022) or local graphs (Wang et al. 2021; Berant et al. 2015) to better model hierarchy. Others leverage hyperbolic geometry (Ganea, Becigneul, and Hofmann 2018; Nickel and Kiela 2017) or box embeddings (Abboud et al. 2020; Jiang et al. 2023) for more expressive hierarchical representations.

Most existing methods represent taxonomy entities using classical word embeddings and infer parent-child relations by measuring similarity between their vectors. How-

ever, such embeddings struggle to capture nuanced, compositional meaning. For instance, “fish” and “drown” may appear similar because they co-occur in sentences like “*Fish can swim in water where others would drown.*” Yet, these embeddings fail to capture the negative connotation - “*Fish cannot drown.*” These limitations call for more expressive representations that model hierarchical and contextual subtleties. This motivates quantum-inspired embeddings, which use principles such as superposition and entanglement to better encode such complex semantics.

To show the importance of quantum embeddings, we model the parent-child hierarchy in the taxonomy by treating individual entities as having limited standalone significance, while their superposition reveals the degree of relatedness between them. Specifically, as illustrated in Fig. 1, we draw inspiration from Fraunhofer’s double-slit experiment in Quantum Physics (Born and Wolf 2013). In this analogy, the two slits represent the parent-child relationship between taxonomy entities. When only one of the slits is opened, the wave corresponding to the individual word passes through, registering on the detection screen as a neutral entity, as seen in cases like “Soil Pollution” and “Used Oil.” However, when both slits are opened, the superposition of the waves from both words reveals the nature of their relationship, which is positive between “Pollution” and “Soil Pollution”, but negative between “Pollution” and “Used Oil.”

We introduce **QuantTaxo**, the **Quantum Taxonomy Expansion Framework** that leverages the superposition principle from quantum physics to represent hierarchical polysemy in a taxonomy. Our key contributions are as follows:

We first construct training data using a self-supervised framework based on the seed taxonomy. Each ⟨parent, child⟩ pair from the taxonomy serves as a positive example, while negative samples are created by pairing the child with non-ancestral nodes. As illustrated in Fig. 1, ⟨Environment, Waste⟩ and ⟨Pollution, Radioactive Pollution⟩ are positive samples, whereas ⟨Pollution, Chemical Waste⟩ and ⟨Chemical Waste, Radioactive Pollution⟩ are negatives.

Secondly, we propose a quantum modeling framework to represent ⟨parent, child⟩ pairs in a complex probabilistic Hilbert space. Inspired by Fraunhofer’s double-slit experiment, we adopt an entanglement-based approach to superimpose parent and child embeddings, yielding quantum entity representations via density matrices. The framework is based on two core hypotheses: (i) a word is a linear combination of latent concepts with complex weights, and (ii) multiple words form a complex superposition of their states. Accordingly, we introduce two variants: (i) **Quant-Sup**, which models entities as linear combinations of latent concepts, and (ii) **Quant-Mix**, which models them as weighted mixtures of word states.

Thirdly, we propose a joint representation framework to quantify the relatedness between parent and child entities in the taxonomy. This involves superimposing their quantum representations to form a composite representation that captures their degree of “entanglement” or interconnectedness. From this, we extract specialized “entangled features” reflecting relational coherence. We assess relatedness using

mathematical properties such as the trace and diagonal elements of the joint matrix—the trace captures cumulative shared attributes, while the diagonal highlights individual contributions and structural alignment within the hierarchy. Ablation studies comparing complex and real embeddings reveal why the complex space is essential, and comprehensive experiments on five benchmarks against nine strong baselines show that **QuantTaxo** lifts performance by an average of 12.3% in accuracy, 11.2% in MRR, and 6.9% in Wu&P metrics, confirming its effectiveness for taxonomy expansion.

2 Preliminaries

2.1 Hilbert Space

In quantum probability theory (Nielsen and Chuang 2010; Von Neumann 2018), quantum systems are modeled within a Hilbert space \mathbb{H}^n , a complex vector space where states are represented as vectors (or density operators), and probabilities are computed from their inner products.

We follow the Dirac notation commonly used in quantum theory, where a state vector $\vec{\psi} \in \mathbb{C}^n$ is denoted as a ket $|\psi\rangle$, and its transpose as a bra $\langle\psi|$. The inner and outer products of unit vectors \vec{u} and \vec{v} are written as $\langle u|v\rangle$ and $|u\rangle\langle v|$, respectively. A vector $|\psi\rangle$ can be expressed as a superposition of basis vectors,

$$|\psi\rangle = \sum_{i=1}^n a_i e^{i\phi_i} |e_i\rangle, \quad (1)$$

where $a_j e^{i\phi_j}$ is the complex-valued probability amplitude for the j^{th} basis vector $|e_j\rangle$. Here, $a_j \geq 0$ are real amplitudes satisfying the normalization $\sum_j a_j^2 = 1$, and $\phi_j \in [-\pi, \pi]$ are the phase angles. Each amplitude can also be expressed in Euler form as $a_j \cos \phi_j + ia_j \sin \phi_j$, and is computed via the inner product: $a_j e^{i\phi_j} = \langle e_j | \psi \rangle$.

Further, the projection measurement is computed as $p(e_j | \psi) = a_j^2 = |\langle e_j | \psi \rangle|^2$, where $p(e_j | \psi)$ represents the probability of the quantum event $|e_1\rangle$, given the quantum state $|\psi\rangle$. The vector $|\psi\rangle$ in Eq. 1 represents a word as a combination of sememes (Goddard 1994)¹, which are the fundamental, indivisible semantic components of word meanings in a language. For instance, the word “robot” can be composed of sememes like “machine”, “automation”, “technology” and “artificial.” The complex phases $\{\phi_j\}_{j=0}^n$ capture *quantum interference* between words. For example, given two words, w_k and w_p with complex amplitudes $a_j^{(k)} e^{i\phi_j^{(k)}}$ and $a_j^{(p)} e^{i\phi_j^{(p)}}$ respectively for the sememe e_j , their combination affects the probability of being in state e_j as follows $\left| a_j^{(k)} e^{i\phi_j^{(k)}} + a_j^{(p)} e^{i\phi_j^{(p)}} \right|^2 = \left| a_j^{(k)} \right|^2 + \left| a_j^{(p)} \right|^2 + 2a_j^{(k)} a_j^{(p)} \cos\left(\phi_j^{(k)} - \phi_j^{(p)}\right)$, where the term $2a_j^{(k)} a_j^{(p)} \cos\left(\phi_j^{(k)} - \phi_j^{(p)}\right)$ represents the interference between w_k and w_p .

¹Sememes are also referred to as latent concepts. Each word is a combination of n latent concepts.

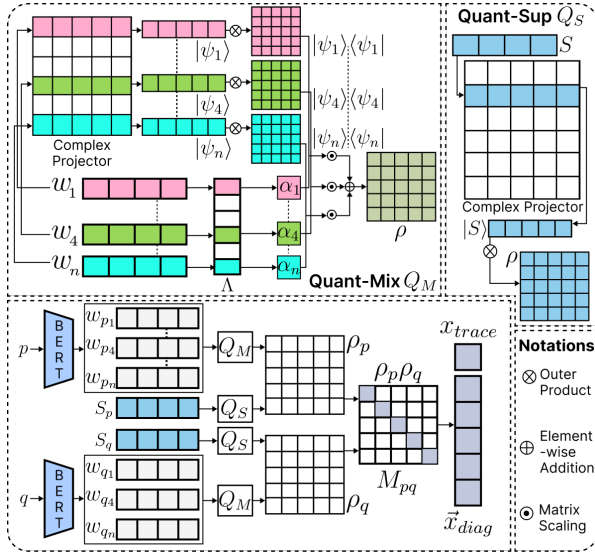


Figure 2: An illustration of the QuanTaxo framework.

2.2 Sentence Representation

A sentence is formed as a combination of words, with each existing in a superposition of underlying sememes. Therefore, the sentence S is a non-classical combination of these sememes. Mathematically, it is represented by an $n \times n$ density matrix ρ , which is positive semi-definite ($\rho \geq 0$) and with a unit trace ($\text{Tr}(\rho) = 1$). The diagonal elements of ρ indicate the contribution of individual concepts, while off-diagonal elements capture their quantum-like correlations between them.

In quantum probabilistic space, a sentence can be represented in two ways: *superposition* and *mixture*. In the *superposition* representation, the sentence exists simultaneously in multiple potential states as a combination of latent concepts, capturing the inherent uncertainty and overlap of interpretations, akin to a quantum particle that exists in multiple states. The density matrix for the sentence S is,

$$\rho = |S\rangle \langle S|, \quad (2)$$

where sentence S is represented (using Eq. 1) as $|S\rangle = \sum_{i=1}^n a_j e^{i\phi_j} |e_j\rangle$.

While in the *mixture* representation, a sentence is a combination of different word states, where each interpretation is considered as a distinct possibility, weighted by its corresponding probability. Unlike superposition, where states coexist as latent concepts, the mixture approach assigns a classical probability distribution over possible word states. The density matrix is computed as,

$$\rho = \sum_{i=1}^n \lambda_i |\psi_i\rangle \langle \psi_i|, \quad (3)$$

where λ_i represents the weight of the word ψ_i with $\sum_{i=1}^n \lambda_i = 1$.

2.3 Taxonomy Expansion

Definition 1. Taxonomy: A taxonomy $\mathcal{T}^o = (\mathcal{N}^o, \mathcal{E}^o)$ is a tree-like directed acyclic graph, where each node $n \in \mathcal{N}^o$

represents a concept and each edge $\langle n_p, n_c \rangle \in \mathcal{E}^o$ denotes a "parent-child" relationship between nodes n_p and n_c .

Definition 2. Taxonomy Expansion: Given a seed taxonomy $\mathcal{T}^o = (\mathcal{N}^o, \mathcal{E}^o)$ and a set of emerging concepts \mathcal{C} , the task is to update the seed taxonomy to $\mathcal{T} = (\mathcal{N}^o \cup \mathcal{C}, \mathcal{E}^o \cup \mathcal{R})$, where \mathcal{R} is the set of newly created relationships linking existing entities \mathcal{E}^o with emerging entities \mathcal{C} . Since surface names of entities alone lack true semantics, entity descriptions D are used to augment representations. Moreover, during inference, query node $q \in \mathcal{C}$ identifies its best-suited parent node $n_p \in \mathcal{N}^o$ by maximizing the matching score ($n_p = \arg \max_{a \in \mathcal{N}^o} f(a, q)$).

3 The QuanTaxo Framework

This section presents QuanTaxo, an end-to-end neural network implemented in the Hilbert space (c.f Figure 2).

3.1 Complex-valued Entity Representation

We start with computing the complex-valued representation of the entities. Given a candidate term $n_p \in \mathcal{N}^o$ and a query term n_q ($n_q \in \mathcal{N}^o$ during training or $n_q \in \mathcal{C}$ while inference), we compute their real-valued vector representations from their surface names and descriptions and then project them into complex space.

Real-valued Entity Projection: Taxonomy nodes, defined by surface names and descriptions D , are encoded into real-valued vectors using pretrained models like BERT (Kenton and Toutanova 2019), in preparation for projection into complex space \mathbb{C}^n . Each entity $n_e \in \{n_p, n_q\}$ is formatted as $d_e = [\text{CLS}] \text{sur}(n_e) [\text{SEP}] D(n_e) [\text{SEP}]$, where $\text{sur}(n_e)$ and $D(n_e)$ denote the entity's surface name and description, respectively, and $[\text{CLS}]$, $[\text{SEP}]$ are BERT-specific tokens.

The textual input d_e is encoded by BERT as $z_e = \text{BERT}(d_e)$, where z_e contains the final-layer embeddings of all tokens. We extract the sentence embedding $S = z_{e[0]}$ from the $[\text{CLS}]$ token for the Quant-Sup and token embeddings $\{\psi\}_{i=0}^n = z_{e[1:m]}$ for the Quant-Mix.

Complex-valued Entity Projection: We project real-valued embeddings into the Hilbert space $\mathbb{H}^n = \mathbb{C}^n$ using a complex phase projector parameterized by amplitude A , phase Φ , and a learnable weight table Λ . Given a real-valued vector x , the projection is defined as $\text{ComplexProjector}(x) = A(x) \odot e^{i\Phi(x)}$, where $A(x) = \|x\|_2$ computes the amplitude, and $\Phi(x) = f_\phi(x)$ applies a linear transformation to generate the phase. To maintain a probabilistic interpretation, amplitudes are normalized such that $\sum_{j=0}^n a_j = 1$, $\forall a_j \in A$. The complex word embedding is then $|\psi_i\rangle = \text{ComplexProjector}(\psi_i)$, combining amplitude a and phase ϕ as per Eq. 1. The weights $\Lambda = \{\alpha_i\}_{i=0}^n$ modulate semantic importance in the Quant-Mix network (Section 3.2). For the Quant-Sup network, the complex embedding is $|S\rangle = \text{ComplexProjector}(S)$.

3.2 Quantum Representation

Quantum embeddings of entities are modeled as density matrices (Section 2.2). The Quant-Sup and Quant-Mix

Dataset	$ \mathcal{N}^o $	$ \mathcal{E}^o $	$ D $
SemEval-Env	261	261	6
SemEval-Sci	429	452	8
MAG-CS-WIKI	25,170	40,314	6
MAG-PSY-WIKI	10,671	14,080	6
WordNet	20.5	19.5	3

Table 1: Statistics of benchmark datasets. $|\mathcal{N}^o|$ and $|\mathcal{E}^o|$ denote the number of nodes and edges in the seed taxonomy, while $|D|$ is the taxonomy depth. For WordNet, values are averaged across 114 sub-taxonomies.

networks compute superposition and mixture-based representations, respectively. In the mixture representation (Eq. 3), a weighted sum of outer products of complex word embeddings is computed. To ensure the unit trace condition $\text{Tr}(\rho) = 1$, weights are normalized into probabilities: $\lambda_i = \frac{\alpha_i}{\sum_{j=0}^{\alpha_i} \alpha_j}$, where λ_i is the probability of the i -th word ψ_i . For comparison, we also consider a uniform weighting with $\lambda_i = 1/n$. In contrast, the superposition-based sentence embedding $|S\rangle$ directly produces a normalized density matrix (Eq. 2), inherently satisfying the trace condition.

3.3 Joint Representation

The query and parent entities are represented as density matrices ρ_q and ρ_p . Instead of using distance-based scoring or concatenation, we model their interaction via a joint representation defined as $M_{pq} = \rho_q \rho_p$.

We decompose the query and parent density matrices as $\rho_q = \sum_i \lambda_i |v_i\rangle \langle v_i|$ and $\rho_p = \sum_j \lambda_j |v_j\rangle \langle v_j|$, where $\lambda_i, |v_i\rangle$ and $\lambda_j, |v_j\rangle$ are the eigenvalues and eigenvectors of the query and parent respectively. These eigenvectors represent latent concepts (or sememes), weighted by their corresponding eigenvalues. The joint representation is given by $\rho_q \rho_p = \sum_{i,j} \lambda_i \lambda_j |v_i\rangle \langle v_j| |v_i\rangle \langle v_j|$, where the inner product $\langle v_i | v_j \rangle$ captures the alignment between basis vectors. Since $\langle v_i | v_j \rangle = \text{Tr}(|v_i\rangle \langle v_j|)$, the trace inner product becomes $\text{Tr}(\rho_q \rho_p) = \sum_{i,j} \lambda_i \lambda_j \langle v_i | v_j \rangle^2$, representing a cosine-similarity-based measure between latent spaces. This quantum-inspired similarity, denoted as M_{pq} , encodes the coherence between query and parent (Balkır 2014).

3.4 Entangled Features for Scoring

In quantum natural language processing, entity similarity is often measured using the negative von Neumann (VN) divergence, $-\Delta_{VN}(\rho_p || \rho_q) = \text{Tr}(\rho_p \log \rho_q)$. However, the matrix logarithm makes it difficult to use in end-to-end learning. To overcome this, we adopt the trace inner product, previously used for word and sentence similarity (Blacoe, Kashefi, and Lapata 2013), and shown to approximate the negative VN divergence effectively (Sordani, Bengio, and Nie 2014; Zhang et al. 2018b). Formally, it is defined as $x_{\text{trace}} = \text{Tr}(\rho_q \rho_p) = \sum_{i,j} \lambda_i \lambda_j \langle r_i | r_j \rangle^2$. This expression captures the semantic overlap driving the similarity between the density matrices of the parent and query entities. To enrich their joint representation, we include the diagonal ele-

ments of the similarity matrix M_{pq} , denoted as \vec{x}_{diag} , which reflect varying importance scores. The final feature vector is thus defined as $\vec{x}_{\text{feat}} = [x_{\text{trace}}; \vec{x}_{\text{diag}}]$. Leveraging these entangled features, we learn a scoring function to effectively rank anchor nodes $n_p \in \mathcal{N}^o$ for a query node q . We define the scoring function as $f(\cdot) : \mathbb{R}^{D_2} \times \mathbb{R}^{D_1} \rightarrow \mathbb{R}$ where $f^{(i)} = \gamma(\mathbf{W}_i f^{(i-1)} + \mathbf{b}_i) \forall i \geq 1$, $f^{(0)} = \vec{x}_{\text{feat}}$ and $f(n_p, n_q) = f^{(N)} = \sigma(f^{(N-1)})$, where γ and σ are the ReLU and sigmoid activations respectively.

3.5 Model Training and Inference

Self-supervised Data Generation. We use the seed taxonomy $\mathcal{T}^o = (\mathcal{N}^o, \mathcal{E}^o)$ to construct training data in a self-supervised manner. For each edge $\langle n_p, n_c \rangle \in \mathcal{E}^o$, where n_p is the parent and n_c the query term, we create a positive sample $\langle n_p, n_c \rangle$. To generate negatives, we fix n_c and randomly sample N non-descendant nodes $\{n_{p'}^i\}_{i=1}^N$ from \mathcal{N}^o —typically siblings, cousins, or other relatives of n_c . This yields a training instance $\mathbf{X} = \{\langle n_p, n_c \rangle, \langle n_{p'}^1, n_c \rangle, \dots, \langle n_{p'}^N, n_c \rangle\}$. Repeating this for each edge in \mathcal{T}^o forms the self-supervised dataset $\mathbb{X} = \{\mathbf{X}_1, \dots, \mathbf{X}_{|\mathcal{E}^o|}\}$.

Model Training. We train the scoring function $f(\cdot)$ on the dataset \mathbb{X} using the binary cross-entropy loss $\mathcal{L}(\Theta) = -\sum_{\mathbf{X}_k} \sum_{i=1}^{N+1} \left[y^i \log f(\vec{x}_{\text{feat}}^i) + (1 - y^i) \log (1 - f(\vec{x}_{\text{feat}}^i)) \right]$, where each sample $(\vec{x}_{\text{feat}}^i, y^i)$ corresponds to a candidate pair $\langle n_p^i, n_c^i \rangle$ in data point \mathbf{X}_k , with $y^i = 1$ for positives and $y^i = 0$ for negatives.

Inference. Given a query node $c \in \mathcal{C}$, the goal during inference is to predict its parent node $n_p \in \mathcal{N}^o$ from the seed taxonomy. For each candidate n_p , we compute a matching score $f(n_p, n_c)$ and select the parent that maximizes this score: $n_p := \arg \max_{n_p \in \mathcal{N}^o} f(n_p, n_c)$. Candidates are ranked by their scores, and the top-ranked node is chosen as the predicted parent. This approach can be extended to return the top- k candidates, if needed.

Computational Complexity Analysis. During the training phase, the model processes $|\mathcal{E}^o| \times (N + 1)$ training instances per epoch, resulting in a computational cost that scales linearly with the number of edges in the seed taxonomy \mathcal{T}^o . During inference, for each query node $n_c \in \mathcal{N}^o$, the model computes $|\mathcal{N}^o|$ matching scores, one for every node in \mathcal{N}^o . Although $O(|\mathcal{N}^o|)$ computations per query can be expensive, in practice, this is sped up via batched scoring and GPU-accelerated matrix multiplications.

4 Experimental Setup

We first outline the experimental setup to evaluate the performance of `QuanTaxo`, covering benchmark datasets, baseline methods, evaluation metrics and implementation details.

4.1 Benchmark Datasets

We evaluate `QuanTaxo` on five benchmarks (c.f. Table 1) – two SemEval-2016 Task 13 datasets, **SemEval-Env** (Env) and **SemEval-Sci** (Sci) (Bordea, Lefever, and

Dataset	SemEval16-Env			SemEval16-Sci			MAG-WIKI-CS			WordNet			MAG-WIKI-PSY		
Metric	Acc	MRR	Wu&P	Acc	MRR	Wu&P	Acc	MRR	Wu&P	Acc	MRR	Wu&P	Acc	MRR	Wu&P
BERT+MLP	12.6 ^{1.1}	23.9 ^{1.6}	48.3 ^{0.8}	12.2 ^{1.7}	19.7 ^{1.4}	45.1 ^{1.1}	8.3 ^{2.4}	15.7 ^{1.5}	36.2 ^{1.3}	9.2 ^{1.2}	17.4 ^{1.3}	43.5 ^{0.4}	7.8 ^{1.4}	15.5 ^{1.1}	37.1 ^{0.6}
TAXI	18.5 ^{1.3}	N/A	47.7 ^{0.4}	13.8 ^{1.4}	N/A	33.1 ^{0.7}	17.3 ^{1.6}	N/A	42.3 ^{0.5}	11.5 ^{1.8}	N/A	38.7 ^{0.7}	12.9 ^{1.4}	N/A	39.7 ^{0.6}
Arborist	9.9 ^{3.9}	27.2 ^{3.5}	43.4 ^{1.6}	19.1 ^{5.0}	32.7 ^{3.2}	47.8 ^{1.8}	15.6 ^{3.7}	34.4 ^{2.9}	46.3 ^{1.8}	12.5 ^{3.3}	27.7 ^{2.1}	52.0 ^{1.2}	15.1 ^{2.8}	28.1 ^{1.9}	44.9 ^{1.5}
TaxoExpan	10.7 ^{4.1}	28.7 ^{3.8}	48.5 ^{1.7}	24.2 ^{5.4}	40.3 ^{3.3}	55.6 ^{1.9}	16.8 ^{4.1}	34.9 ^{3.2}	47.7 ^{1.3}	17.3 ^{3.5}	31.1 ^{2.3}	57.6 ^{1.8}	15.9 ^{2.2}	30.4 ^{1.8}	51.2 ^{1.1}
TMN	34.6 ^{3.0}	41.7 ^{4.4}	53.6 ^{3.5}	32.6 ^{2.4}	46.1 ^{1.9}	65.3 ^{1.5}	24.4 ^{3.2}	39.7 ^{2.3}	53.1 ^{1.6}	20.3 ^{1.9}	35.9 ^{1.5}	54.7 ^{1.3}	19.7 ^{2.0}	34.8 ^{1.7}	51.9 ^{0.3}
STEAM	34.1 ^{3.4}	44.3 ^{2.1}	65.2 ^{1.4}	34.8 ^{4.5}	50.7 ^{2.5}	72.1 ^{1.7}	25.9 ^{4.0}	41.3 ^{2.5}	53.7 ^{0.4}	21.4 ^{2.8}	38.2 ^{1.4}	59.8 ^{1.3}	23.5 ^{3.1}	36.3 ^{1.6}	53.3 ^{1.2}
BoxTaxo	32.3 ^{5.8}	45.7 ^{3.2}	73.1 ^{1.2}	26.3 ^{4.5}	41.1 ^{3.1}	61.6 ^{1.4}	23.5 ^{5.4}	35.7 ^{4.1}	49.2 ^{1.7}	22.3 ^{3.1}	35.7 ^{2.7}	58.6 ^{1.2}	21.1 ^{3.4}	34.9 ^{2.9}	54.0 ^{1.3}
Musubu	42.3 ^{3.2}	57.1 ^{1.4}	64.4 ^{0.7}	44.5 ^{2.3}	59.7 ^{1.6}	75.2 ^{1.2}	28.5 ^{2.9}	37.1 ^{2.0}	51.9 ^{0.7}	25.3 ^{4.9}	36.1 ^{2.9}	61.2 ^{0.9}	25.7 ^{4.4}	35.0 ^{2.3}	57.1 ^{1.0}
TEMP	45.5 ^{8.6}	59.1 ^{6.3}	77.3 ^{2.8}	43.5 ^{7.8}	57.5 ^{5.6}	76.3 ^{1.5}	31.8 ^{1.4}	45.3 ^{2.1}	57.8 ^{1.2}	24.6 ^{5.1}	37.5 ^{4.6}	61.2 ^{2.3}	25.9 ^{2.1}	38.9 ^{1.9}	59.7 ^{0.3}
Quant-Sup	49.2^{2.1}	59.5^{1.2}	79.1^{0.3}	57.3^{2.8}	67.4^{1.2}	82.1^{0.4}	35.7^{2.2}	51.8^{1.6}	61.4^{0.6}	25.8^{1.1}	41.8^{0.6}	71.0^{0.3}	27.3^{1.4}	43.2^{0.7}	61.0^{0.4}
Quant-Mix	<u>46.6^{2.1}</u>	<u>57.9^{1.2}</u>	<u>76.1^{0.1}</u>	<u>52.4^{1.7}</u>	<u>64.1^{1.1}</u>	<u>77.6^{0.6}</u>	<u>33.2^{1.9}</u>	<u>47.7^{1.1}</u>	<u>59.2^{0.7}</u>	<u>22.1^{1.3}</u>	<u>39.6^{0.8}</u>	<u>69.8^{0.4}</u>	<u>23.5^{1.5}</u>	<u>38.4^{1.0}</u>	<u>59.4^{0.5}</u>

Table 2: Performance comparison between `QuantTaxo` and baseline methods. Results for each method are presented as **mean^{std-dev}** in percentage across three runs with three random seeds. The best performance is marked in bold, while the best baseline is underlined. As TAXI (Panchenko et al. 2016) outputs the taxonomy as a whole, it cannot produce MRR values.

Buitelaar 2016), **WordNet**, which contains 114 depth-3 sub-taxonomies with 10–50 nodes each (Bansal et al. 2014), and two large-scale taxonomies, **MAG-CS-WIKI** (CS) and **MAG-PSY-WIKI** (PSY), derived from subgraphs of the Microsoft Academic Graph (MAG) (Sinha et al. 2015). Concept definitions are taken from prior work (Yu et al. 2020; Jiang et al. 2023; Wang et al. 2021; Liu et al. 2021; Arous, Dolamic, and Cudré-Mauroux 2023). Following (Yu et al. 2020), we randomly sample 20% of leaf nodes as test nodes (with valid parents) and use the remaining nodes as the seed taxonomy for self-supervised training. Since taxonomies such as CS and PSY allow multiple parents, but Wu&P similarity assumes a tree, we derive a spanning tree by duplicating subtrees for multi-parent nodes, ensuring a cycle-free structure with a unique ancestor path for each node.

4.2 Baseline Methods

We evaluate `QuantTaxo` against a range of baselines based on classical embeddings utilizing pretrained language models, graph neural networks, and a few prompting-based methods, all of which encode semantic and structural features. We use the following baselines: • **BERT+MLP** (Kenton and Toutanova 2019) encodes term surface forms with BERT and feeds the resulting vectors to a multi-layer perceptron to classify hypernym relationship. • **TAXI** (Panchenko et al. 2016) extracts candidate hypernyms via lexical patterns and substring heuristics, then prunes them to produce an acyclic taxonomy without ranking. • **Arborist** (Manzoor et al. 2020) models heterogeneous edge semantics and trains with a large-margin ranking loss whose margin adapts dynamically. • **TaxoExpan** (Shen et al. 2020) represents an anchor node by encoding its ego network with a graph neural network and scores parent–child pairs through a log-bilinear feed-forward layer. • **TMN** (Zhang et al. 2021) fuses auxiliary and primary signals using a neural tensor network and refines embeddings via a channel-wise gating mechanism. • **STEAM** (Yu et al. 2020) ensembles graph, contextual, and lexical-syntactic features in

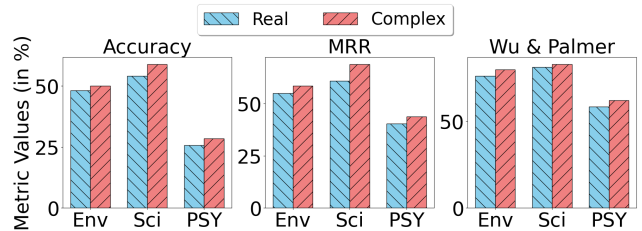


Figure 3: Performance comparison of real and complex-valued embeddings across ‘Env’, ‘Sci’ and ‘PSY’.

a multi-view co-training framework to score hypernymy links. • **BoxTaxo** (Jiang et al. 2023) learns box embeddings and evaluates parent candidates with geometric and probabilistic losses derived from hyper-rectangle volumes. • **Musubu** (Takeoka, Akimoto, and Oyamada 2021) fine-tunes language-model classifiers on Hearst-pattern phrases that contain both query and parent terms. • **TEMP** (Liu et al. 2021) encodes full root-to-parent paths and trains with a dynamic-margin loss to represent query nodes.

Baselines such as TMN are designed for taxonomy completion. They insert a query q between a parent–child pair and therefore score triplets $f(p, c, q)$. Our setting is taxonomy expansion, where q is attached as a leaf under a parent and no child c is available. For a fair comparison, we follow (Wang et al. 2022) and adapt these baselines by instantiating c with a dummy placeholder (e.g., a blank/sentinel token). This converts their triplet scorer to an effective leaf-attachment scheme while preserving their original scoring function.

4.3 Evaluation Metrics

Given a query set \mathcal{C} , let $\{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_{|\mathcal{C}|}\}$ be the predicted parent nodes and $\{y_1, y_2, \dots, y_{|\mathcal{C}|}\}$ the corresponding ground-truth labels. Following prior work (Jiang et al. 2023; Liu et al. 2021), we evaluate the performance of `QuantTaxo` and all baselines using three standard metrics, namely, **Ac-**

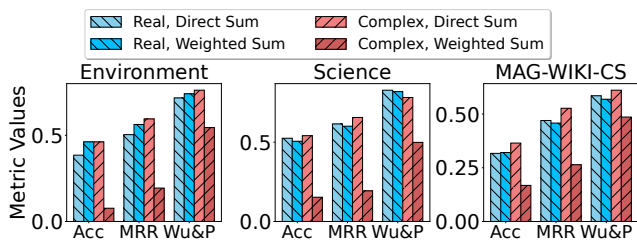


Figure 4: Performance comparison of direct and weighted sum in real-valued and complex-valued mixture models across ‘Env’, ‘Sci’, and ‘WIKI’ benchmarks.

curacy (Acc) = $\text{Hit}@1 = \frac{1}{|C|} \sum_{i=1}^{|C|} \mathbb{I}(y_i = \hat{y}_i)$, where $\mathbb{I}(\cdot)$ is the indicator function, **Mean Reciprocal Rank (MRR)** = $\frac{1}{|C|} \sum_{i=1}^{|C|} \frac{1}{\text{rank}(y_i)}$ and **Wu & Palmer Similarity (Wu&P)** (Wu and Palmer 1994) = $\frac{1}{|C|} \sum_{i=1}^{|C|} \frac{2 \times \text{DEPTH}(\text{LCA}(\hat{y}_i, y_i))}{\text{DEPTH}(\hat{y}_i) + \text{DEPTH}(y_i)}$, where $\text{DEPTH}(\cdot)$ denotes the node’s depth in the taxonomy.

4.4 Implementation Details

We implement `QuanTaxo` in PyTorch, with baselines, excluding BERT+MLP, sourced from the respective repositories of their original authors. All training and inference tasks are conducted on an 80GB NVIDIA A100 GPU to ensure high computational efficiency. For the implementation, we utilize `bert-base-uncased` as the default pre-trained model, with the hidden layer size W_2 set to 64 and a dropout rate of 0.1. The maximum padding length for inputs is fixed at 128, and the optimizer used is AdamW, with a learning rate of 2×10^{-5} for BERT fine-tuning and of 1×10^{-3} for training the remaining weights. Training is performed using a batch size of 128 over a maximum of 100 epochs. The sizes of density matrices are kept the same as the dimensions of `bert-base-uncased`, i.e., 768.

5 Experimental Results

5.1 Main Results

Table 2 shows that `QuanTaxo` consistently outperforms prior state-of-the-art models based on classical embeddings and structural summaries across all evaluation metrics. Early methods like BERT+MLP rely on surface-level names and classical embeddings, ignoring structural cues. First-generation models (e.g., Musubu, TAXI) add lexical and semantic features but struggle with semantic ambiguity. Second-generation methods (e.g., TaxoExpan, STEAM) include structural signals yet still use classical embeddings, limiting semantic depth. Recent baselines like TEMP use path-based transformers but remain bound to classical embeddings. BoxTaxo explores geometric embeddings but is limited by instability and traditional representations. `QuanTaxo` surpasses these baselines by leveraging quantum embeddings to capture semantic entanglements without explicit structural inputs. Its strong performance, even on larger datasets, highlights the representational power and robustness of quantum semantics in modeling taxonomic hierarchy and coherence.

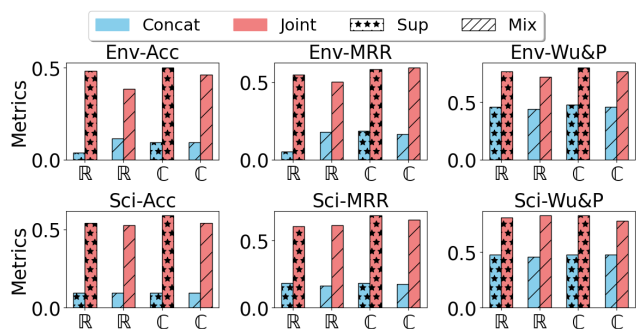


Figure 5: Comparison of parent-child joint representation and concatenation for scoring on ‘Env’ and ‘Sci’. ‘Sup’ \rightarrow superposition module, while ‘Mix’ \rightarrow mixture module. $\mathbb{R} \rightarrow$ real embedding while $\mathbb{C} \rightarrow$ complex embedding.

5.2 Ablation Studies

`QuanTaxo` comprises three key modules: a complex embedding projector, a quantum module, and a joint representation module. We investigate the effect of different configurations on the performance of taxonomy expansion.

Impact of Real and Complex Representations on Quantum Modeling. We assess the effectiveness of complex-valued embeddings in `QuanTaxo` by comparing them to real-valued embeddings that use only amplitude and omit phase information. As shown in Figure 3, complex embeddings consistently outperform real-valued ones across all metrics. By incorporating both amplitude and phase, they enable richer, context-aware representations via quantum interference, leading to improved accuracy, higher MRR through better parent ranking, and superior Wu & Palmer similarity by preserving hierarchical structure. These results highlight the importance of amplitude-phase interactions in capturing nuanced relational semantics, making complex embeddings integral to the success of `QuanTaxo`.

Impact of Direct Sum vs. Weighted Sum in Mixture Models. We compare direct and weighted summation in real- and complex-valued mixture models across three benchmarks (Section 3.2, Figure 4). While both methods perform similarly in the real space, weighted summation significantly underperforms in the complex space. This is due to its disruption of the balance between amplitude and phase, which are crucial for capturing semantics in complex embeddings (Mönning and Manandhar 2018). Misaligned phases can cause destructive interference, distorting the representation and leading to information loss.

Comparison of Parent-Child Joint Representation and Concatenation for Scoring. We compare joint representation against concatenation for parent-child scoring on the SemEval16-Env and SemEval16-Sci benchmarks (Fig. 5). Across all settings, joint representation consistently outperforms concatenation. Notably, complex-valued embeddings with joint representation yield the best results, effectively capturing hierarchical and semantic relationships. While real-valued embeddings also benefit from the joint

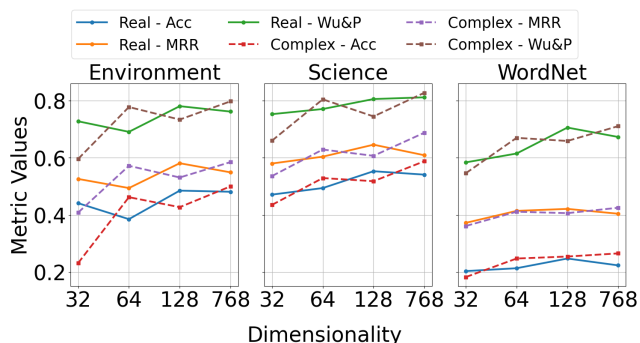


Figure 6: Effect of density matrix dimensionality on model performance across ‘Env’, ‘Sci’ and ‘CS’.

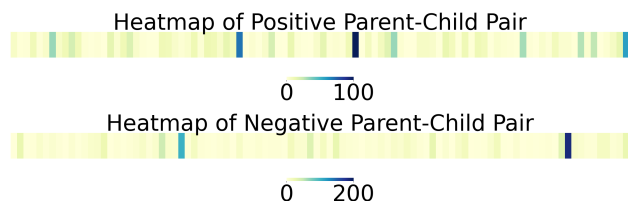


Figure 7: Heatmaps of diagonals of joint representation of positive \langle “environmental impact”, “environmental policy” \rangle and negative \langle “environmental impact”, “environmental research” \rangle pairs, from SemEval16-Env.

setup, they remain slightly behind their complex counterparts. These findings highlight the superiority of joint representation—especially with complex embeddings—over simple concatenation for modeling relational structure.

Effect of Dimensionality of Density Matrix on Performance. We analyze the impact of density matrix dimensionality on model performance (Fig. 6) by projecting BERT embeddings into varying dimensions and computing their outer products. Performance generally improves with higher dimensionality. However, for complex-valued embeddings, performance stabilizes early—for instance, results at dimension 64 closely match those at 768. In contrast, real-valued embeddings continue to benefit from increased dimensions. These findings underscore the efficiency and robustness of complex-valued embeddings in capturing hierarchical and relational semantics, even at lower dimensions.

5.3 Distribution of Density Matrices

We analyze the diagonal features \vec{x}_{diag} from the joint representation to assess parent-child relationship scoring. Using the SemEval16-Env dataset and the query term *environmental impact* (true parent: *environmental policy*), we compare its density matrix with that of the lowest-scoring negative parent, *environmental research*. As shown in Figure 7, positive pairs exhibit higher intensity and more uniform diagonal values (ranging 0–100), reflecting strong semantic and hierarchical alignment. In contrast, negative pairs show lower intensity, irregular patterns, and sparse high values, indicating weak or absent coherence. These patterns highlight the

Dataset	Query	Ground Truth	Predicted Parent	Score
Env	dust	atmospheric pollutant	environment	0.809
	non-polluting vehicle	pollution control measures	environment	0.825
	groundwater	water	water	0.999
	tropical zone	climatic zone	climatic zone	0.999
Sci	radiobiology	biology	science	0.961
	enzymology	biochemistry	genetics	0.978
	microbiology	biology	biology	0.999
	nuclear physics	physics	physics	1.000

Table 3: Examples of `QuanTaxo`’s predictions with confidence scores across benchmarks.

effectiveness of the joint representation in capturing and distinguishing meaningful parent-child relationships.

5.4 Case Study

In this section, we present an error analysis with case studies to evaluate the effectiveness of the `QuanTaxo` framework on SemEval-2016 datasets (Table 3). For each benchmark, we examine two correct and two incorrect predictions, leading to two key insights on the strengths of quantum embeddings in modeling semantic and hierarchical relationships. `QuanTaxo` performs well on clear and well-defined concepts such as *groundwater*, *tropical zone* and *microbiology*, accurately retrieving appropriate anchor nodes due to the clarity of their definitions. In contrast, it struggles with ambiguous or underspecified terms like *enzymology*, and *non-polluting vehicle*, where insufficient definitions hinder precise semantic grounding. Notably, incorrect predictions generally yield low matching scores, indicating that quantum embeddings avoid forming spurious entanglements in uncertain cases. These lower scores can serve as a heuristic for identifying potentially unreliable predictions in the absence of ground truth.

6 Conclusion

We propose `QuanTaxo`, a taxonomy expansion framework that uses quantum embeddings to model hierarchical and semantic structure. It consists of three core components: a complex embedding projector, a quantum representation module, and a joint representation module. By projecting entities into a complex-valued Hilbert space, `QuanTaxo` captures rich relationships through superposition and entanglement, which are well-suited for modeling hierarchy and polysemy. The quantum module encodes nuanced semantic overlaps, while the joint module fuses parent and child density matrices for coherent representation. Experiments on benchmarks show that `QuanTaxo` outperforms classical embedding methods and structure-aware models. Ablation studies highlight the impact of quantum representations and complex embeddings in enhancing semantic modeling. Error analysis further reveals `QuanTaxo`’s robustness in handling ambiguous entities. Overall, `QuanTaxo` demonstrates the promise of quantum-inspired approaches for scalable and accurate taxonomy modeling.

Acknowledgements

We gratefully acknowledge support from the Prime Minister’s Research Fellowship (PMRF), which funded this research.

References

- Abboud, R.; Ceylan, İ. İ.; Lukasiewicz, T.; and Salvatori, T. 2020. BoxE: A Box Embedding Model for Knowledge Base Completion. In *NeurIPS*, volume 33, 9649–9661.
- Arous, I.; Dolamic, L.; and Cudré-Mauroux, P. 2023. Taxo-Complete: Self-Supervised Taxonomy Completion Leveraging Position-Enhanced Semantic Matching. In *Proceedings of the ACM Web Conference 2023*, WWW ’23, 2509–2518. New York, NY, USA: Association for Computing Machinery. ISBN 9781450394161.
- Balkır, E. 2014. Using density matrices in a compositional distributional model of meaning. *Master’s thesis, University of Oxford*.
- Bansal, M.; Burkett, D.; de Melo, G.; and Klein, D. 2014. Structured Learning for Taxonomy Induction with Belief Propagation. In Toutanova, K.; and Wu, H., eds., *Proceedings of the 52nd ACL*, 1041–1051. Baltimore, Maryland: Association for Computational Linguistics.
- Berant, J.; Alon, N.; Dagan, I.; and Goldberger, J. 2015. Efficient Global Learning of Entailment Graphs. *Computational Linguistics*, 41(2): 221–263.
- Blacoe, W.; Kashefi, E.; and Lapata, M. 2013. A quantum-theoretic approach to distributional semantics. In *NAACL*, 847–857.
- Bordea, G.; Lefever, E.; and Buitelaar, P. 2016. SemEval-2016 Task 13: Taxonomy Extraction Evaluation (TExEval-2). In *SemEval*, 1081–1091. San Diego, California: Association for Computational Linguistics.
- Born, M.; and Wolf, E. 2013. *Principles of optics: electromagnetic theory of propagation, interference and diffraction of light*. Elsevier.
- Ganea, O.; Becigneul, G.; and Hofmann, T. 2018. Hyperbolic Entailment Cones for Learning Hierarchical Embeddings. In Dy, J.; and Krause, A., eds., *ICML*, volume 80, 1646–1655. PMLR.
- Goddard, C. 1994. *Semantic and lexical universals: Theory and empirical findings*. John Benjamins.
- Gonçalves, R. S.; Horridge, M.; Li, R.; Liu, Y.; Musen, M. A.; Nyulas, C. I.; Obamos, E.; Shrouy, D.; and Temple, D. 2019. Use of OWL and Semantic Web Technologies at Pinterest. In *The Semantic Web – 18th ISWC 2019, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part II*, 418–435. Berlin, Heidelberg: Springer-Verlag. ISBN 978-3-030-30795-0.
- Jiang, M.; Song, X.; Zhang, J.; and Han, J. 2022. TaxoEnrich: Self-Supervised Taxonomy Completion via Structure-Semantic Representations. In *WWW*, 925–934. New York, NY, USA: ACM. ISBN 9781450390965.
- Jiang, S.; Yao, Q.; Wang, Q.; and Sun, Y. 2023. A Single Vector Is Not Enough: Taxonomy Expansion via Box Embeddings. In *WWW*, 2467–2476. New York, NY, USA: Association for Computing Machinery. ISBN 9781450394161.
- Jurgens, D.; and Pilehvar, M. T. 2015. Reserating the awesome-tastic: An automatic extension of the WordNet taxonomy for novel terms. In Mihalcea, R.; Chai, J.; and Sarkar, A., eds., *NAACL*, 1459–1465. Denver, Colorado: Association for Computational Linguistics.
- Karamanolakis, G.; Ma, J.; and Dong, X. L. 2020. TXtract: Taxonomy-Aware Knowledge Extraction for Thousands of Product Categories. 8489–8502.
- Kenton, J. D. M.-W. C.; and Toutanova, L. K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, 2. Minneapolis, Minnesota.
- Liu, X.; Song, Y.; Liu, S.; and Wang, H. 2012. Automatic taxonomy construction from keywords. In *SIGKDD*, 1433–1441. New York, NY, USA: Association for Computing Machinery. ISBN 9781450314626.
- Liu, Z.; Xu, H.; Wen, Y.; Jiang, N.; Wu, H.; and Yuan, X. 2021. TEMP: Taxonomy Expansion with Dynamic Margin Loss through Taxonomy-Paths. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *EMNLP*, 3854–3863. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.
- Manzoor, E.; Li, R.; Shrouy, D.; and Leskovec, J. 2020. Expanding Taxonomies with Implicit Edge Semantics. In *WWW*, 2044–2054. New York, NY, USA: Association for Computing Machinery. ISBN 9781450370233.
- Mao, Y.; Zhao, T.; Kan, A.; Zhang, C.; Dong, X. L.; Faloutsos, C.; and Han, J. 2020. Octet: Online Catalog Taxonomy Enrichment with Self-Supervision. In *SIGKDD*, 2247–2257. New York, NY, USA. ISBN 9781450379984.
- Mönnig, N.; and Manandhar, S. 2018. Evaluation of complex-valued neural networks on real-valued classification tasks. *arXiv preprint arXiv:1811.12351*.
- Nickel, M.; and Kiela, D. 2017. Poincaré Embeddings for Learning Hierarchical Representations. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.
- Nielsen, M. A.; and Chuang, I. L. 2010. *Quantum computation and quantum information*. Cambridge university press.
- Panchenko, A.; Faralli, S.; Ruppert, E.; Remus, S.; Naets, H.; Fairon, C.; Ponzetto, S. P.; and Biemann, C. 2016. TAXI at SemEval-2016 Task 13: a Taxonomy Induction Method based on Lexico-Syntactic Patterns, Substrings and Focused Crawling. In *SemEval*, 1320–1327. San Diego, California: Association for Computational Linguistics.
- Shen, J.; Shen, Z.; Xiong, C.; Wang, C.; Wang, K.; and Han, J. 2020. TaxoExpan: Self-supervised Taxonomy Expansion with Position-Enhanced Graph Neural Network. In *WWW*, 486–497. New York, NY, USA: Association for Computing Machinery. ISBN 9781450370233.
- Sinha, A.; Shen, Z.; Song, Y.; Ma, H.; Eide, D.; Hsu, B.-J. P.; and Wang, K. 2015. An Overview of Microsoft Academic Service (MAS) and Applications. In *Proceedings of the 24th International Conference on World Wide Web*, WWW ’15

Companion, 243–246. New York, NY, USA: Association for Computing Machinery. ISBN 9781450334730.

Snow, R.; Jurafsky, D.; and Ng, A. 2004. Learning Syntactic Patterns for Automatic Hypernym Discovery. In Saul, L.; Weiss, Y.; and Bottou, L., eds., *Advances in Neural Information Processing Systems*, volume 17. MIT Press.

Sordoni, A.; Bengio, Y.; and Nie, J.-Y. 2014. Learning concept embeddings for query expansion by quantum entropy minimization. In *AAAI*, volume 28.

Takeoka, K.; Akimoto, K.; and Oyamada, M. 2021. Low-resource Taxonomy Enrichment with Pretrained Language Models. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *EMNLP*, 2747–2758. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Velardi, P.; Faralli, S.; and Navigli, R. 2013. OntoLearn Reloaded: A Graph-Based Algorithm for Taxonomy Induction. *Computational Linguistics*, 39(3): 665–707.

Von Neumann, J. 2018. *Mathematical foundations of quantum mechanics: New edition*, volume 53. Princeton university press.

Wang, C.; Danilevsky, M.; Desai, N.; Zhang, Y.; Nguyen, P.; Taula, T.; and Han, J. 2013. A phrase mining framework for recursive construction of a topical hierarchy. In *SIGKDD*, 437–445. New York, NY, USA: Association for Computing Machinery. ISBN 9781450321747.

Wang, S.; Zhao, R.; Chen, X.; Zheng, Y.; and Liu, B. 2021. Enquire One’s Parent and Child Before Decision: Fully Exploit Hierarchical Structure for Self-Supervised Taxonomy Expansion. In *WWW*, 3291–3304. New York, NY, USA: Association for Computing Machinery. ISBN 9781450383127.

Wang, S.; Zhao, R.; Zheng, Y.; and Liu, B. 2022. Qen: Applicable taxonomy completion via evaluating full taxonomic relations. In *WWW*, 1008–1017.

Wu, Z.; and Palmer, M. 1994. Verb Semantics and Lexical Selection. In *ACL*, 133–138. Las Cruces, New Mexico, USA.

Yu, Y.; Li, Y.; Shen, J.; Feng, H.; Sun, J.; and Zhang, C. 2020. STEAM: Self-Supervised Taxonomy Expansion with Mini-Paths. In *SIGKDD*, 1026–1035. New York, NY, USA: Association for Computing Machinery. ISBN 9781450379984.

Zhang, C.; Tao, F.; Chen, X.; Shen, J.; Jiang, M.; Sadler, B.; Vanni, M.; and Han, J. 2018a. TaxoGen: Unsupervised Topic Taxonomy Construction by Adaptive Term Embedding and Clustering. In *KDD*, 2701–2709. New York, NY, USA: Association for Computing Machinery. ISBN 9781450355520.

Zhang, J.; Song, X.; Zeng, Y.; Chen, J.; Shen, J.; Mao, Y.; and Li, L. 2021. Taxonomy completion via triplet matching network. In *AAAI*, volume 35, 4662–4670.

Zhang, P.; Niu, J.; Su, Z.; Wang, B.; Ma, L.; and Song, D. 2018b. End-to-end quantum-like language models with application to question answering. In *AAAI*, volume 32.