

# Completing Taxonomies with Relation-Aware Mutual Attentions

Qingkai Zeng, Zhihan Zhang, Jinfeng Lin, Meng Jiang

†Department of Computer Science and Engineering, University of Notre Dame, Notre Dame, IN, USA  
{qzeng,zzhang23,jlin6,mjiang2}@nd.edu

## ABSTRACT

Taxonomies serve many applications with a structural representation of knowledge. To incorporate emerging concepts into existing taxonomies, the task of taxonomy completion aims to find suitable positions for emerging query concepts. Previous work captured *homogeneous* token-level interactions inside a concatenation of the query concept term and definition using pre-trained language models. However, they ignored the token-level interactions between the term and definition of the query concepts and their related concepts. In this work, we propose to capture *heterogeneous* token-level interactions between the different textual components of concepts that have different types of relations. We design a relation-aware mutual attention module (RAMA) to learn such interactions for taxonomy completion. Experimental results demonstrate that our new taxonomy completion framework based on RAMA achieves the state-of-the-art performance on six taxonomy datasets. This paper belongs to “Application and analysis - Knowledge Graph Construction”, and in the “Novel research paper” category.

## KEYWORDS

taxonomy completion, mutual attention, concept definition, heterogeneous interactions

## 1 INTRODUCTION

Taxonomy is a hierarchical structure that represents the hypernym-hyponym relations or “is-A” relations between concepts or entities. Researchers have been using relational knowledge to identify semantic relevance for web search [9, 27], personalized recommendation [21, 32], and question answering [26].

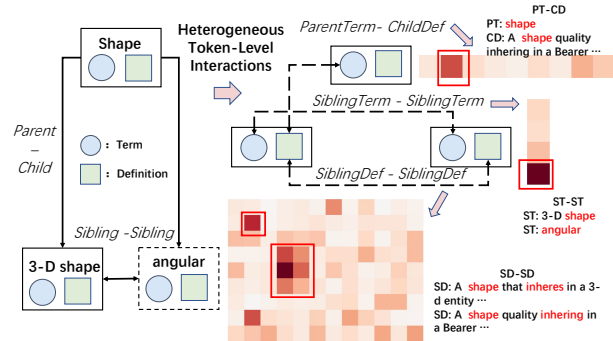
Existing taxonomies are constructed mainly by human experts or through crowd-sourcing. However, manual taxonomy construction is time-consuming, labor-intensive, and limited in knowledge coverage [3, 8]. With the influx of emerging concepts, it is necessary to complete the taxonomies with emerging concepts to keep them up-to-date. The tasks of taxonomy completion [8, 31] are therefore proposed to reduce the manual effort of building and maintaining taxonomies. Such technologies automatically enrich pre-built taxonomies by inserting emerging concepts into the existing relational graph.

Existing taxonomy completion methods used graph learning algorithms to propagate the semantic encodings of concept terms

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

MLG '23, August, 2023, Long Beach, CA

© 2023 Association for Computing Machinery.  
ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00  
<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

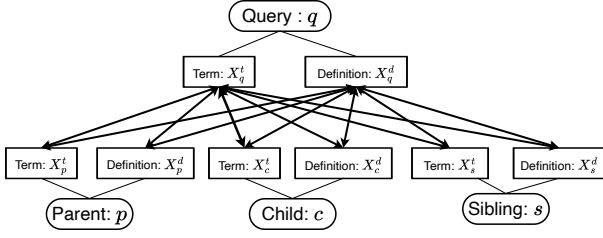


**Figure 1: Heterogeneous token-level interactions in taxonomy: A concept on a taxonomy has two textual components: term (blue) and definition (green). The taxonomic relation between two concepts is deeply embedded in the interactions of tokens in the texts. Different relations and different components create different patterns of interactions that should be modeled by different mutual attention modules in a language model. In this example, three types of interactions are more salient than others: (1) parent’s term and child’s definition, (2) siblings’ terms, and (3) siblings’ definitions.**

over a taxonomy [15, 19, 29, 31]. As pre-trained language models (PLMs) greatly improved various text processing systems, they have been utilized for encoding the concept’s textual description, i.e., a concatenation of term and definition [4, 11, 23, 24]. PLMs incorporated a self-attention mechanism to explore the interactions between the tokens of the concatenated text. Such token-level interactions provided better homogeneous semantic representations of concepts than pure term encodings, which led to a significant improvement on taxonomy completion.

Existing methods that rely solely on homogeneous token-level interactions have two limitations: (1) they ignore the valuable token-level interactions between concepts that have specific relations; (2) they simply concatenate the term and definition texts into a single textual sequence without considering the source of the interacted tokens (i.e., term or definition). To address these limitations, given a pair of concepts, we propose to consider both (1) their roles in a relation and (2) the textual components of concept description (i.e., sources of tokens) to produce taxonomy-contextualized representations.

For a given query concept, we classify various token-level interactions with its neighboring concepts based on their roles (e.g., parent, child, sibling) and the textual components (e.g., term, definition) of the concepts. We name them as *heterogeneous* token-level interactions. In Figure 1, we illustrate the significance of such heterogeneous token-level interactions on the ENvironment Ontology (ENVO). As shown, the term and definition of a query concept have notable correlations with (1) the term of its parent concept, (2) the



**Figure 2: Instead of concatenating term and definition of query concept for self-attention, we utilize up to twelve types of token-level interactions among 8 pieces of text  $\{X\}$  with respect to the relation with query  $(p, c, s)$  and the textual component  $(t$  and  $d)$  by heterogeneous mutual attentions.**

definition of its sibling concept, and (3) the term of its sibling concept. For example, the parent concept “Shape” is referenced in the definition of “3-D Shape,” and the terms of sibling concepts “3-D shape” and “2-D shape” are lexically similar. Additionally, the definitions of sibling concepts “3-D shape” and “angula” were written with similar templates.

Figure 2 shows the twelve types of token-level interactions. In taxonomy completion, we look for a suitable position of query concept  $q$ . Given its term’s word sequence  $X_q^t$  and its definition’s word sequence  $X_q^d$ , a solution should model the interactions between the tokens in the term or definition of a concept that is a parent  $p$ , a child  $c$ , or a sibling  $s$ . In this paper, we use double arrows to denote the types of token-level interactions. For example,  $X_q^d \leftrightarrow X_p^t$  means the interactions between the tokens in the query  $q$ ’s definition  $X_q^d$  and the tokens in its parent node  $p$ ’s term  $X_p^t$ .

On this novel idea, we propose a module called RAMA that implements a relation-aware mutual attention mechanism and propose an effective taxonomy completion framework based on this module. RAMA models all the heterogeneous token-level interactions, creating dynamic representations for each concept depending on its role and the type of textual content it interacts with. To prevent performance degradation caused by redundant information, we conduct an empirical study on six real-world taxonomies to determine which types of interactions make a positive impact. Extensive empirical results suggest that RAMA considers four types of interactions ( $X_q^t \leftrightarrow X_c^d$ ,  $X_q^d \leftrightarrow X_p^t$ ,  $X_q^t \leftrightarrow X_s^d$ , and  $X_q^d \leftrightarrow X_s^d$ ).

We conduct experiments on six real-world taxonomies from various domains. RAMA achieves the state of the art in taxonomy completion task. RAMA outperforms baselines by 1.9% in mean reciprocal rank (MRR), 6.1% in precision and 5.0% in recall. Ablation studies further demonstrate that the relation-aware mutual attention mechanism effectively captures the knowledge from the four selected types of token-level interactions.

The main contributions of this work are summarized as follows:

- We designed a comprehensive study to investigate twelve types of token-level interactions to improve taxonomy completion, while past work ignored heterogeneity. Empirical study on six real-world taxonomies shows that four types of the interactions have positive impacts on taxonomy completion performance.
- We propose a new framework on a novel module RAMA that incorporates a relation-aware mutual attention mechanism

to utilize heterogeneous token-level interactions between terms and definitions effectively.

- Extensive experiments demonstrate that RAMA significantly improves the performance of taxonomy completion tasks on six datasets from various domains.

## 2 PROBLEM DEFINITION

In this section, we present the key concepts used in this paper and formally define the taxonomy completion problem:

*Definition 2.1 (Taxonomy).* We follow the definition of taxonomy in [31]. In our taxonomy, each concept has its associated definition. A taxonomy  $\mathcal{T} = (\mathcal{V}, \mathcal{E})$  is a directed acyclic graph composed by a vertex set  $\mathcal{V}$  and an edge set  $\mathcal{E}$ . Each vertex  $v$  is associated with a set of descriptive text  $X_v = \{X_v^t, X_v^d\}$ , where  $X_v^t$  is its term and  $X_v^d$  is its definition. Meanwhile, each directed edge  $\langle p, c \rangle \in \mathcal{E}$  represents a parent-child relationship that points to a vertex  $c$  from its most exact hypernymy vertex  $p$ .

*Definition 2.2 (Taxonomy Completion).* Given a set of emerging concepts  $\mathcal{V}'$ , taxonomy completion aims to add them into an existing *seed taxonomy*  $\mathcal{T}_0 = (\mathcal{V}_0, \mathcal{E}_0)$ . The goal is to enrich  $\mathcal{T}_0$  to be a larger taxonomy  $\mathcal{T} = (\mathcal{V}_0 \cup \mathcal{V}', \mathcal{E}')$ . To insert each query concept  $q \in \mathcal{V}'$ , we identify a proper candidate position  $u(q) = \langle p(q), c(q) \rangle \in \mathcal{E}_0$  (simply denoted by  $u, p$ , and  $c$ ), remove the edge  $\langle p, c \rangle$ , and add two new edges  $\langle p, q \rangle$  and  $\langle q, c \rangle$ . So, we have:

$$\mathcal{E}' = \mathcal{E}_0 \setminus_{q \in \mathcal{V}'} \{ \langle p, c \rangle \} \cup_{q \in \mathcal{V}'} \{ \langle p, q \rangle, \langle q, c \rangle \}.$$

Therefore, after the insertion of  $q$  at the candidate position  $u, p$  becomes the parent concept of  $q$  and  $c$  becomes the child concept of  $q$ . Note that  $p$  or  $c$  could be a placeholder node in case concepts need to be inserted as root or leaf nodes.

## 3 PROPOSED FRAMEWORK

**Overall architecture.** Figure 3 illustrates the architecture of the proposed framework. It handles any combination of heterogeneous token-level interactions between definitions and terms in a pair of concepts. It is composed of three main components: (1) Relation-Aware Mutual Attention (RAMA) module, which captures token-level interactions between the two concepts; (2) Homogeneous Representation Generation module, which generates representations of the concepts; and (3) Scoring module, which is used for both training and inference.

The working flow of these three modules in the framework is as follows: the input consists of a query concept from the emerging concept set and its potential positions within the existing seed taxonomy. Each concept has two textual components: term and definition. The RAMA module inputs the interaction pairs under examination and obtains relation-aware embeddings for each pair. The Homogeneous Representation Generation module takes the concatenation of the concept’s definition and term as input, generating a code representation for each concept. The Scoring module inputs both the code representations and relation-aware embeddings to encode parent-child and sibling-sibling relations. It ultimately outputs a fitting score for each potential position of the query concept. We then place the query concept in the potential position that receives

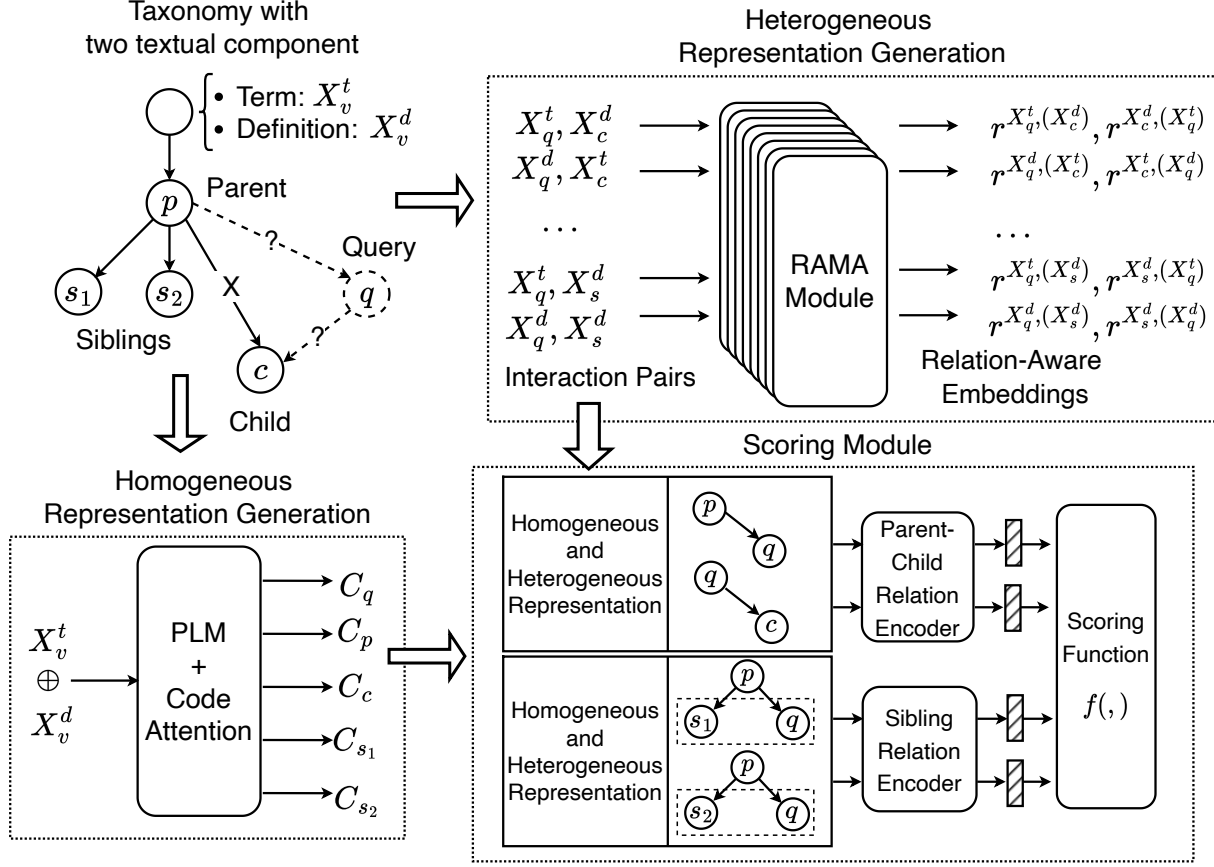


Figure 3: Our framework has the RAMA module to improve taxonomy completion. Each unified RAMA module is responsive to a combination of interactions mentioned in Figure 2 to generate heterogeneous representations. The scoring module feeds both homogeneous and heterogeneous representations into parent-child and sibling relation encoders.

the highest fitting score, resulting in an enriched taxonomy that incorporates these emerging concepts.

### 3.1 Relation-Aware Mutual Attention

In Figure 2, we have introduced twelve heterogeneous token-level interactions between the term/definition of a query concept and the counterparts from its potential parent, child, and sibling concepts, if the query concept is inserted in the candidate position. To leverage the token-level interactions within the diverse relationships for taxonomy completion, we propose a relation-aware mutual attention (RAMA) module. For ease of reference, we use  $(\alpha, \beta)$  to represent any heterogeneous interaction between the query concept and the anchor concept. Specifically, the anchor concept can be the parent, child, or sibling concept of the query concept. The symbols  $\alpha$  and  $\beta$  refer to one of textual components (term or definition) of the query and anchor concepts respectively, which can be formally written as  $\alpha \in \{X_q^t, X_q^d\}$  and  $\beta \in \{X_p^t, X_p^d, X_c^t, X_c^d, X_{s_1}^t, X_{s_2}^d\}$ .

We illustrate our relation-aware mutual attention mechanism in Figure 4. To enhance the representations of the query concept and the anchor concept, we propose a relation-aware mutual attention mechanism that captures token-level interactions between the two concepts. Essentially, the mutual attention module generates

mutually dependent representations of the query concept and the anchor concept by enabling tokens from different sources to attend to one another.

Given a type of heterogeneous interaction  $(\alpha, \beta)$ , we extract two matrices,  $\mathbf{Q} \in \mathbb{R}^{l_\alpha \times d}$  and  $\mathbf{V} \in \mathbb{R}^{l_\beta \times d}$ , from a pre-trained language model. Here,  $l_\alpha$  and  $l_\beta$  are sequence lengths of  $\alpha$  and  $\beta$ , respectively, and  $d$  is the dimension of the embedding vectors. To capture the token-level interactions between  $\alpha$  and  $\beta$ , we calculate the correlation matrix  $\mathbf{W}^{(\alpha, \beta)} \in \mathbb{R}^{l_\alpha \times l_\beta}$  as follows:

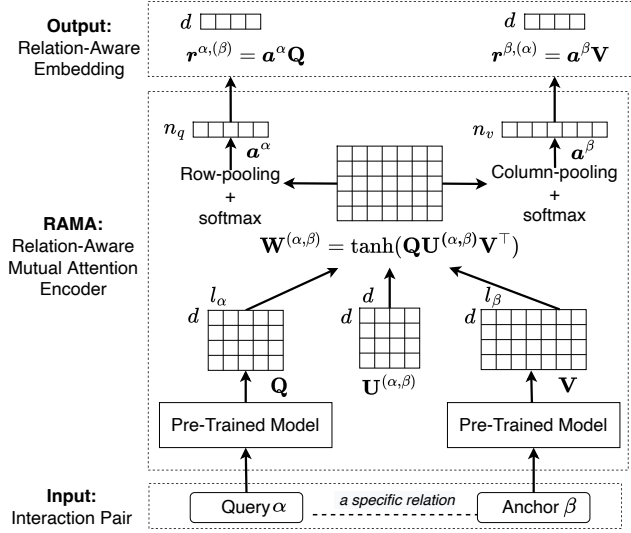
$$\mathbf{W}^{(\alpha, \beta)} = \tanh(\mathbf{Q}\mathbf{U}^{(\alpha, \beta)}\mathbf{V}^\top),$$

where  $\mathbf{U} \in \mathbb{R}^{d \times d}$  is a learnable parameter matrix. Each element  $w_{i,j}$  in  $\mathbf{W}^{(\alpha, \beta)}$  represents the pairwise correlation score between two tokens in  $\alpha$  and  $\beta$ .

For each token in the term and/or definition of query concept  $\alpha$ , we calculate its importance weight  $g_i^\alpha$  as the average of its correlation scores with all tokens in  $\beta$ . We perform the same operation on  $\beta$  to obtain  $g_j^\beta$ :

$$g_i^\alpha = \text{MEAN}(w_{i,1}, w_{i,1}, \dots, w_{i,n}),$$

$$g_j^\beta = \text{MEAN}(w_{1,j}, w_{2,j}, \dots, w_{m,j}).$$



**Figure 4: The details of the Relation-Aware Mutual Attention (RAMA) module: RAMA takes a pair of concepts with their textual descriptions and generates a relation-aware embedding vector as output, capturing the token-level interactions between the two concepts.**

To incorporate the token-level interaction with the original semantic information, we aggregate the original representations of  $\alpha$  and  $\beta$  using these importance weights. Specifically, the importance weights are normalized using the softmax function before we apply a weighted sum on the original representations:

$$\mathbf{r}^{\alpha, (\beta)} = \mathbf{a}^{\alpha} \mathbf{Q}, \quad \mathbf{r}^{\beta, (\alpha)} = \mathbf{a}^{\beta} \mathbf{V}, \quad (1)$$

$$\text{where } a_i^{\alpha} = \frac{\exp(g_i^{\alpha})}{\sum_{j=1}^{l_{\alpha}} \exp(a_j^{\alpha})}, \quad a_i^{\beta} = \frac{\exp(g_i^{\beta})}{\sum_{j=1}^{l_{\beta}} \exp(a_j^{\beta})}.$$

**3.1.1 Unification of Heterogeneous Token-level Interactions.** RAMA uses an independent parameter matrix,  $\mathbf{U}^{(\alpha, \beta)}$ , to model each of the twelve token-level interactions represented by  $\mathbf{r}^{\alpha, (\beta)}$  and  $\mathbf{r}^{\beta, (\alpha)}$  in Eq. (2). However, it is worth noting that the parent-child and child-parent relationships are symmetric in nature. Thus, we apply a **Unification** operation for these symmetric token-level interactions. Unification is the process of sharing the parameter matrices of two symmetrical token-level interactions through a unified operation. In our final model, if Unification is not enabled, separate parameter matrices are used to model  $X_q^t \leftrightarrow X_c^d$  and  $X_q^d \leftrightarrow X_p^t$ . However, based on the experimental results shown in Table 2, we chose to unify the attention matrices for symmetrical interactions in RAMA to attain a more efficient and effective model.

## 3.2 Homogeneous Representation Generation

In RAMA, each type of textual component of a concept is initially encoded using a pre-trained language model to generate its original representations. However, to improve the performance of the model, we have integrated two additional components: the relation-aware mutual attention module and the code attention mechanism. While the relation-aware mutual attention module focuses on identifying and modeling the heterogeneous interactions between the

tokens in the query’s textual component and those in its anchor concepts, the code attention mechanism proposed in [6] is employed to reinforce the homogeneous interactions within the two types of textual components of the concept itself. This allows the model to effectively analyze and utilize both heterogeneous and homogeneous interactions in the descriptive text to improve its overall performance.

**3.2.1 PLM-based Text Encoding.** For an arbitrary concept vertex  $v$  that represents any vertex from the set  $\{q, p, c, s_1, s_2\}$ , the pre-trained model takes both the term  $X_v^t$  and definition  $X_v^d$  as input to generate a semantic representation  $\mathbf{D}_v$ . As we believe that terms and definitions provide complementary information that tend to result in a richer representation than using any part alone, we concatenate them with special token [CLS] and [SEP] and then encode the resulting input sequence using a pre-trained BERT model:

$$\mathbf{D}_v = \text{BERT}([\text{CLS}]X_v^t[\text{SEP}]X_v^d[\text{SEP}]) \in \mathbb{R}^{l_v \times d},$$

where  $l_v$  is the length of combined input sequence, and  $d$  is the dimension of BERT representations. We choose DistilBERT [16] as the pre-trained language model for SemEval2016 taxonomies, and SciBERT [2] for the taxonomies in Graphine [10].

**3.2.2 Code Attention.** In order to improve the correlation between the term and definition within a concept, we have adopted the design of PolyEncoders [6] in RAMA. The method utilizes learnable codes  $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_m$  which are attended to the representation  $\mathbf{D}_v$  generated by the pre-trained language model. This results in a code representation  $\mathbf{C}_v$  of size  $m \times d$ , which captures the correlation between the term and definition:

$$a_{e_j}^i = \frac{\exp(\mathbf{e}_j \mathbf{D}_v^{i \cdot})}{\sum_{h=1}^{l_v} \exp(\mathbf{e}_j \mathbf{D}_v^{h \cdot})},$$

$$\mathbf{C}_v^{j \cdot} = \sum_{i=1}^{l_v} a_{e_j}^i \mathbf{D}_v^{i \cdot}. \quad (2)$$

For each concept, the code representation  $\mathbf{C}_v$  is created by incorporating information from its own term and definition. This representation contains multiple aspects of the concept’s semantic information from the homogeneous token-level interaction, specifically the correlation between the term and definition.

## 3.3 Scoring Module for Training and Inference

The scoring module has two encoders: Parent-child relation encoder and Sibling relation encoder. These encoders score the fitness between the query  $q$  and the candidate position  $u$ . Both encoders take in the code representations from homogeneous token-level interaction (see Eq. (2)) as well as the relation-aware mutual representations from heterogeneous token-level interaction (see Eq. (1)) as inputs. The Parent-child relation encoder calculates the representations of the parent-child relation between the query  $q$  and the candidate position  $u$ , and the Sibling relation encoder calculates the representations of the sibling relation. The fitness score between  $q$  and  $u$  is then calculated based on these representations of the two types of taxonomic relations.

**3.3.1 Parent-child relation encoder.** The parent-child relation encoder module of RAMA is responsible for calculating the parent-child relation representation for final scoring. This module uses a two-layer Transformer architecture as the encoder. Given a query concept  $q$ , its potential parent concept  $p$ , and child concept  $c$ , the module aims to learn the parent-child relational representations  $\mathbf{h}_{p,q} \in \mathbb{R}^d$  and  $\mathbf{h}_{q,c} \in \mathbb{R}^d$ , respectively. The input to the encoder is created by concatenating the code representations and the relation-aware mutual representations of  $q$ ,  $p$ , and  $c$  into a single sequence:

$$\mathbf{h}_{p,q} = \text{TRANSFORMER} \left( \left[ C_p^\top, C_q^\top, \bigoplus_{\alpha \in \mathcal{X}_q, \beta \in \mathcal{X}_p} [\mathbf{r}^{\alpha,(\beta)\top}, \mathbf{r}^{\beta,(\alpha)\top}] \right]^\top \right),$$

$$\mathbf{h}_{q,c} = \text{TRANSFORMER} \left( \left[ C_q^\top, C_c^\top, \bigoplus_{\alpha \in \mathcal{X}_q, \beta \in \mathcal{X}_c} [\mathbf{r}^{\alpha,(\beta)\top}, \mathbf{r}^{\beta,(\alpha)\top}] \right]^\top \right),$$

where both  $[\cdot, \cdot]$  and  $\bigoplus$  stand for concatenation, we set  $\mathbf{e}_{CLS}$  as a learnable placeholder for downstream tasks. The function  $\text{TRANSFORMER}(\cdot)$  incorporates a sinusoidal positional embedding to make the input order sensitive. The transformer input is comprised of the stacked code representation  $C$  and all relation-aware mutual representations  $\mathbf{r}^{\alpha,(\beta)}$ ,  $\mathbf{r}^{\beta,(\alpha)}$ . The output embeddings of the CLS tokens are taken as  $\mathbf{h}_{p,q}$  and  $\mathbf{h}_{q,c}$ , which represent the parent-child relation representation. To minimize the number of parameters, the same transformer is used to encode  $\mathbf{h}_{p,q}$  and  $\mathbf{h}_{q,c}$ .

**3.3.2 Sibling relation encoder.** This module computes the sibling-sibling relation representation for the final scoring. We use a two-layer MLP to encode the sibling relation representation  $\mathbf{h}_{q,s} \in \mathbb{R}^d$ . The input for each concept is a concatenation of the first code representation from Eq. (2) and their relation-aware mutual representation:

$$\mathbf{h}_{q,s_1} = \text{MLP} \left( \left[ C_q^1, C_{s_1}^1, \bigoplus_{\alpha \in \mathcal{X}_q, \beta \in \mathcal{X}_{s_1}} [\mathbf{r}^{\alpha,(\beta)}, \mathbf{r}^{\beta,(\alpha)}], |C_{s_1}^1 - C_q^1| \right] \right),$$

$$\mathbf{h}_{q,s_2} = \text{MLP} \left( \left[ C_q^1, C_{s_2}^1, \bigoplus_{\alpha \in \mathcal{X}_q, \beta \in \mathcal{X}_{s_2}} [\mathbf{r}^{\alpha,(\beta)}, \mathbf{r}^{\beta,(\alpha)}], |C_{s_2}^1 - C_q^1| \right] \right).$$

**3.3.3 Scoring function.** The final scoring function  $f$  is computed by assessing the compatibility between the query concept  $q$  and the candidate position  $u$  using parent-child relation embeddings and sibling relation embeddings. We combine all of these representations as the input to a two-layer MLP, and the output is the final fitting score between the query and the candidate position. The calculation is as follows:

$$f(q, u) = \text{MLP} \left( [\mathbf{h}_{p,q}, \mathbf{h}_{q,c}, \mathbf{h}_{q,s_1}, \mathbf{h}_{q,s_2}] \right).$$

To avoid the need for additional labeled data, we utilize the existing taxonomy  $\mathcal{T}_0$  to generate self-supervised training data for RAMA. During the training process, each concept  $v \in \mathcal{V}_0$  is treated as a query concept. For each query concept  $q$  and its corresponding correct candidate position  $u$  in the existing taxonomy, we construct this pair as a positive example. Additionally, we constructed  $N$  negative examples  $u^*$  by randomly selecting adjacent concepts of  $q$ . Each training batch is composed of  $N+1$  pairs (one positive and  $N$  negative), represented as  $\mathcal{B} = \{(q_i, u_i, y_i) \mid i \in \{1, \dots, |\mathcal{B}|\}\}$ , where

	#Concepts	#Edges	Depth
SemEval2016			
SemEval-Sci	429	452	8
SemEval-Env	261	261	6
SemEval-Food	1,486	1576	8
Graphine			
ARO (Antibiotic Resistance Ontology)	4,549	4,638	8
ENVO (Environment Ontology)	4,121	4,465	13
DOID (Human Disease Ontology)	8,479	10,867	11

**Table 1: Statistics of six taxonomy data sets.**

$y_i \in \{0, 1\}$  is the label for each training instance. The loss function  $\mathcal{L}$  is based on the cross-entropy of  $f(q, u)$  as follows:

$$\mathcal{L} = - \sum_{(q_i, u_i, y_i) \in \mathcal{B}} y_i \log(f(q_i, u_i)) + (1 - y_i) \log(1 - f(q_i, u_i)).$$

**3.3.4 Selection of Heterogeneous Token-level Interactions.** As shown in Figure 2, a query concept has twelve possible heterogeneous token-level interactions with its anchor concepts (parent, child, and sibling). Our goal is to incorporate these interactions into our proposed module, RAMA, to enhance the taxonomy completion task. Theoretically, RAMA is able to encode any combination of the twelve interactions. We employ the **Selection** operation to minimize redundant information obtained from utilizing all twelve interactions. Selection refers to the process of identifying a subset of the twelve token-level interactions in order to reduce redundant information that may negatively impact the performance of taxonomy completion task. In practice, we will choose four specific types of token-level interactions out of the twelve based on the results presented in Figure 5. Specifically, we will conduct experiments with various types of interactions and select the types of interactions that have positive impact.

Four types of interactions used by RAMA are:

- (1)  $X_q^t \leftrightarrow X_c^d$ : The interaction between the term of query concept  $q$  and the definition of its child concept  $c$ .
- (2)  $X_q^d \leftrightarrow X_p^t$ : The interaction between the definition of query concept  $q$  and the term of its parent concept  $p$ .
- (3)  $X_q^d \leftrightarrow X_s^d$ : The interaction between the definitions of query concept  $q$  and its sibling concept  $s$ .
- (4)  $X_q^t \leftrightarrow X_s^t$ : The interaction between the terms of query concept  $q$  and its sibling concept  $s$ .

## 4 EXPERIMENTS

Our proposed framework is evaluated on six real-world taxonomies. The experiments aim to address three research questions (RQs):

- **RQ1:** Which token-level interactions would have a positive or negative impact to enhance the performance of taxonomy completion task?
- **RQ2:** How does the performance of RAMA compare to state-of-the-art baselines in taxonomy completion?

- **RQ3:** Which token-level interactions within the RAMA have the most significant impact on the effectiveness of taxonomy completion task?

## 4.1 Experimental Settings

**4.1.1 Datasets.** We evaluate the performance of taxonomy completion methods using six taxonomies from SemEval2016 [3] and Graphine [10]. The statistics of these taxonomies are presented in Table 1. The SemEval2016 includes three taxonomies in the domains of environment, science, and food, while Graphine includes 227 taxonomies in the biomedical domain. To evaluate the effectiveness of RAMA across domains, we select the Antibiotic Resistant Ontology (ARO), Environment Ontology (ENVO), and Human Disease Ontology (DOID) in our experiments. All concepts and definitions included in these taxonomies are provided by human experts.

**4.1.2 Baselines.** On taxonomy completion task, we compared the performance of RAMA with the following baseline methods:

- **TMN [31]:** It proposes a one-to-pair matching mechanism as opposed to the one-to-one matching used in other methods. Additionally, it uses a channel-wise gating function to enhance performance by regulating concept embeddings.
- **GenTaxo [30]:** This generation-based model encodes taxonomic relations into sentences and subgraphs to generate concepts at valid positions in an existing taxonomy.
- **TaxoEnrich [7]:** It encodes pseudo-sentences using a pre-trained language model to aggregate taxonomic relation information from taxonomy-paths and siblings.
- **QEN [24]:** It utilizes text descriptions with a pre-trained language model for concept representations and uses quadruple evaluation to evaluate full taxonomic relations.

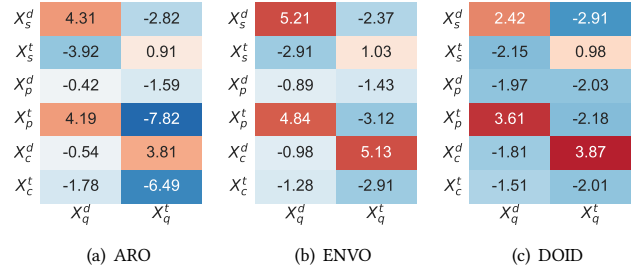
**4.1.3 Implementation Details.** In the implementation of RAMA, the AdamW optimizer [12] is utilized for training the model. A linear warm-up is applied with the learning rate linearly increasing to  $5e-5$  in the first 10% of training steps. The model is trained for a total of 100 epochs based on the validation loss. All experiments were conducted on an NVIDIA RTX 3090 24GB GPU.

**4.1.4 Evaluation Methods.** We evaluate the performance of RAMA and the baselines for taxonomy completion and expansion tasks using several commonly used ranking metrics. Similarly, for the taxonomy completion task, we adopt the evaluation methods used in prior work [24, 30, 31] and report MRR, Precision and Recall as evaluation metrics.

## 4.2 The Selection of Heterogeneous Correlations (RQ1)

As mentioned in Figure 2, in this work, we consider a total of 12 heterogeneous token-level interactions related to the query concept and the target taxonomy. In order to identify which interactions have positive impacts on taxonomy enrichment tasks, we first apply each individual interaction independently in our RAMA module. The performance difference after adding each type of interaction is shown in Figure 5.

The experimental results from the three Graphine taxonomies revealed consistent observations. For clarity in the following text,



**Figure 5: Each heatmap shows the relative performance difference (in percentage) after adding a specific individual interaction to the RAMA module. Red cells indicate positive performance differences (i.e., MRR relative improvement) while blue cells indicate negative performance differences. According to the heatmap, the interactions of  $X_q^d \leftrightarrow X_p^t$ ,  $X_q^t \leftrightarrow X_c^d$ ,  $X_q^d \leftrightarrow X_s^d$  and  $X_q^t \leftrightarrow X_s^t$  are shown to have a positive influence on performance, while others have a negative impact on the original model.**

we use  $X_p^t \leftrightarrow X_c^d$  to represent the parent term-child definition relationship, which encompasses both  $X_q^t \leftrightarrow X_c^d$  and  $X_q^d \leftrightarrow X_p^t$ . Furthermore, to standardize, we use  $X_s^t \leftrightarrow X_s^t$  to represent the sibling term relationship  $X_q^t \leftrightarrow X_s^t$ , and  $X_s^d \leftrightarrow X_s^d$  to represent the sibling definition relationship  $X_q^d \leftrightarrow X_s^d$ .

The  $X_p^t \leftrightarrow X_c^d$  and  $X_s^d \leftrightarrow X_s^d$  interactions significantly contributed to the performance. Compared to  $X_p^t \leftrightarrow X_c^d$  and  $X_s^d \leftrightarrow X_s^d$ , the  $X_s^t \leftrightarrow X_s^d$  interaction showed a slight but consistent improvement across all taxonomies. On the other hand, the other interactions mentioned in Figure 5 had a negative impact on the performance of the taxonomy completion task. This suggests that the token-level interactions in these other interactions do not have strong correlations to the taxonomy completion task and provide redundant information to the model. So, we include  $X_p^t \leftrightarrow X_c^d$ ,  $X_s^d \leftrightarrow X_s^d$  and  $X_s^t \leftrightarrow X_s^d$  interactions in the final RAMA model.

## 4.3 Performance on Taxonomy Completion (RQ2)

In our experiments, we compare the performance of our RAMA model to all the baseline models mentioned in Section 4.1.2. We present the experimental results on six SemEval and Graphine datasets for the taxonomy completion task. As shown in Table 2, RAMA consistently achieves the state-of-the-art performance on all six benchmarks for both tasks. We have three main observations as follow:

First, the performance of GenTaxo was significantly worse than the other baselines. Because it did not have access to the set of emerging concepts, causing low-quality generated concepts. These concepts negatively affected its performance, indicating that the terms of emerging concepts contain valuable taxonomic relation information.

Second, QEN and TaxoEnrich were the top-performing baseline methods. They used pre-trained language models to encode the

Dataset Metric			ARO			ENVO			DOID		
			MRR	Precision	Recall	MRR	Precision	Recall	MRR	Precision	Recall
			43.1	18.5	14.2	46.7	21.2	18.4	48.1	23.6	21.4
			-	11.5	8.9	-	12.9	9.6	-	12.4	9.3
			43.7	18.9	14.5	47.1	20.8	21.3	49.6	24.2	22.8
			43.3	<u>19.2</u>	<u>14.8</u>	<u>48.3</u>	<u>22.9</u>	<u>22.1</u>	<u>50.2</u>	<u>24.5</u>	<u>23.1</u>
Unification	Selection		MRR	Precision	Recall	MRR	Precision	Recall	MRR	Precision	Recall
	-	-	42.1	18.7	13.6	44.8	18.6	17.9	45.7	20.8	19.2
RAMA	✓	-	41.5	17.5	11.6	43.3	18.9	16.7	46.9	22.1	20.6
	-	✓	44.8	20.1	<b>15.3</b>	48.3	23.8	22.7	49.6	25.1	22.6
	✓	✓	<b>45.6</b>	<b>20.5</b>	15.1	<b>49.2</b>	<b>24.3</b>	<b>23.2</b>	<b>50.8</b>	<b>25.6</b>	<b>23.9</b>
Dataset Metric			SemEval16-Env			SemEval16-Sci			SemEval16-Food		
			MRR	Precision	Recall	MRR	Precision	Recall	MRR	Precision	Recall
			34.2	17.4	16.1	32.9	18.1	16.8	31.2	17.1	13.5
			-	9.6	10.1	-	7.4	8.7	-	7.9	9.2
			<u>36.7</u>	19.8	<u>18.6</u>	34.2	18.5	16.7	34.2	17.6	14.3
			36.2	<u>20.1</u>	18.2	<u>35.9</u>	<u>20.1</u>	<u>17.4</u>	<u>34.9</u>	<u>18.3</u>	<u>14.9</u>
Unification	Selection		MRR	Precision	Recall	MRR	Precision	Recall	MRR	Precision	Recall
	-	-	33.1	15.8	17.1	33.5	18.8	17.1	31.9	17.4	13.6
RAMA	✓	-	33.6	16.5	16.6	33.2	18.9	16.3	32.1	17.1	13.9
	-	✓	36.2	20.1	18.6	36.9	20.1	18.4	35.1	18.3	14.6
	✓	✓	<b>36.9</b>	<b>20.5</b>	<b>18.9</b>	<b>37.8</b>	<b>20.9</b>	<b>19.7</b>	<b>35.6</b>	<b>18.8</b>	<b>15.1</b>

Table 2: Performance on taxonomy completion: Bold for the highest among all. Underlined for the best baseline. All metrics presented in percentages (%). The MRR of GenTaxo [30] is not available since it generates the term for candidate positions instead of a ranking list for potential parents. The results also demonstrate the effectiveness of the interaction Selection and mutual attention Unification operations for parent-child relationship and child-parent relationship.

textual contents, in order to better learn and extract useful information. QEN performed better than TaxoEnrich in the majority of datasets (4 out of 6), indicating that the definition of concepts in QEN includes more useful taxonomic relation information than the pseudo-sentences used in TaxoEnrich.

Last, our proposed module, RAMA, achieved the best results on all benchmarks and outperformed previous state-of-the-art models in all metrics. Specifically, on the ENVO dataset, RAMA outperformed the best baseline model, QEN, by 1.9%, 6.1%, 5.0% on MRR, precision, recall metrics, respectively. This superior performance is attributed to the method’s ability to capture taxonomic relation information through the aggregation of heterogeneous and homogeneous token-level interactions. This information leads to better performance on all the datasets.

#### 4.4 Ablation Study (RQ3)

We conducted an ablation study on the six datasets mentioned above to further verify the effectiveness of these three types of token-level interactions ( $X_p^t \leftrightarrow X_c^d$ ,  $X_s^t \leftrightarrow X_s^d$  and  $X_s^d \leftrightarrow X_s^t$ ) modeled by RAMA. We created variants of RAMA by removing specific token-level interaction from its input.

As shown in Table 3, in experiments on three Graphine datasets, removing any token-level interaction harms RAMA’s performance.

This proves that all three types of interaction provide additional information that is useful to taxonomy expansion and taxonomy completion. Our observations are as follows:

First, removing  $X_s^d \leftrightarrow X_s^t$  leads to the most significant decrease in performance across all metrics. This indicates the definition of the potential sibling concept plays the most critical role in identifying the position of the query concept. In comparison to the full model on the ARO dataset, the absence of interaction between sibling definitions decreases MRR, Precision, and Recall for taxonomy completion by 4%, 9%, and 5%, respectively. The significance of the interaction between sibling definitions is also supported by Figure 5, which shows that incorporating only token-level interaction between sibling definitions results in the greatest improvement for both taxonomy enrichment tasks.

Second, the parent-child relationship  $X_p^t \leftrightarrow X_c^d$  has the second significant contribution to RAMA’s performance. Removing the token-level interaction between the query concept with its potential parent and child concepts leads to a non-negligible deterioration on all datasets.

Lastly, removing the  $X_s^t \leftrightarrow X_s^d$  interaction also decreases performance of RAMA to a small extent. For instance, the RAMA variant without  $X_s^t \leftrightarrow X_s^d$  input exhibits only a slight decrease (approximately 0.5 points lower than the full model) in both MRR and

Dataset				ARO			ENVO			DOID		
Metric (Taxonomy Completion)				MRR	Precision	Recall	MRR	Precision	Recall	MRR	Precision	Recall
$X_p^t \leftrightarrow X_c^d$	$X_s^t \leftrightarrow X_s^t$	$X_s^d \leftrightarrow X_s^d$										
RAMA	✓	✓	-	43.8	18.7	14.2	47.4	21.5	21.8	49.3	23.6	22.4
	-	✓	✓	44.6	19.8	14.7	48.1	23.1	22.3	49.6	24.7	22.8
	✓	-	✓	45.1	20.1	14.9	48.6	23.8	22.9	50.1	25.1	23.3
	✓	✓	✓	45.6	20.5	15.1	49.2	24.3	23.2	50.8	25.6	23.9

**Table 3: An evaluation of the impact of token-level interactions on the performance of RAMA through an ablation study. The results show that parent term ( $X_p^t \leftrightarrow X_c^d$ ) and sibling definition information ( $X_s^d \leftrightarrow X_s^d$ ) are both valuable for the taxonomy completion task, with a minimal difference in performance between removing either interaction. While the sibling term interaction ( $X_s^t \leftrightarrow X_s^t$ ) has a limited impact on overall performance, it is still important to include it in the full model.**

precision on the ARO dataset. Moreover, it only exhibits a minimal decrease of 0.2 compared to the recall of the full model. This aligns with Figure 5 in which the interaction between  $X_q^t$  and  $X_s^t$  displays the smallest improvement among all selected interactions.

## 5 RELATED WORK

### 5.1 Taxonomy Expansion and Completion

In the field of taxonomy expansion, several works have been proposed to place emergent concepts in appropriate positions within seed taxonomies. Aly *et al.* used hyperbolic embeddings to refine existing taxonomies [1]. Fauceglia *et al.* proposed a hybrid approach combining linguistic patterns, semantic web, and neural networks [5]. Manzoor *et al.* incorporated implicit edge semantics in the representation of concepts [15]. Shen *et al.* proposed a position-enhanced graph neural network to leverage neighborhood information [19]. Ma *et al.* extended this framework to hyperbolic space [14]. Yu *et al.* applied a mini-path-based classifier using distributional, contextual, and lexical-syntactic features [29]. Liu *et al.*, Wang *et al.* and Takeoka *et al.* proposed various methods to enhance taxonomy expansion by using pre-trained language models [11, 20, 23]. Liu *et al.* used definition and taxonomy-paths to capture taxonomic relation information. Wang *et al.* proposed a novel ego-tree structure to fully capture taxonomic relation information. Takeoka *et al.* focused on addressing the problem in low-resource settings.

Taxonomy completion task generalized the expansion task to non-leaf concepts via finding both the parent and the child for the query concept. Zhang *et al.* proposed a channel-wise gating function and neural tensor networks as matching modules [31]. Jiang *et al.* used pseudo sentences to aggregate taxonomic relation information from taxonomy-paths and sibling concepts [7]. Wang *et al.* used text descriptions and pre-trained language models to learn the representations of both existing concepts and emergent concepts [24]. In contrast to ranking-based methods, Zeng *et al.* used a generative model to generate the query concept to be added [30].

### 5.2 Multi-Source Attention

A related line of work to the techniques used in our model is multi-source attention, in which features from multiple sources of input interact with each other to boost mutual information. A major

group of work in this area is in the question answering domain, where representations of the question and the candidate answer (or the evidence context) mutually attend to each other to fuse information from both directions. An early example of this is given by Santos *et al.* [17], who used a pooling layer on a two-way attention module to extract the most important features for decision making. Other approaches aimed to reduce information loss by using multi-source attention weights to obtain co-dependent representations. For example, Seo *et al.* [18] and Yu *et al.* [28] performed bidirectional attention flow between the query and context to produce query-aware context representations and context-aware query representations. This kind of multi-source attention mechanism can also be extrapolated to other tasks that involve multiple sources of input. For example, in network embedding learning, the input sources can be text descriptions for different nodes [22]. And in multi-modality settings, the input sources can be visual features (extracted from images or videos) and textual features (questions, captions, comments, etc.), as mentioned in visual question answering [13] and multi-modal text generation approaches [25, 33].

## 6 CONCLUSIONS

In this paper, we investigated twelve types of relational interactions in a hierarchical structure and found that only three or four of them are beneficial for the taxonomy completion task. We proposed RAMA that improved relational token-level interactions among the terms and definitions in concepts for the taxonomy completion task by learning these types of interactions through a relational mutual attention mechanism. Our experimental results demonstrated that RAMA achieved state-of-the-art performance on six real-world taxonomies.

## ACKNOWLEDGMENT

This work was supported by NSF IIS-2119531, IIS-2137396, IIS-2142827, IIS-2234058, CCF-1901059, and ONR N00014-22-1-2507.



## REFERENCES

- [1] Rami Aly, Shantanu Acharya, Alexander Ossa, Arne Köhn, Chris Biemann, and Alexander Panchenko. 2019. 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*. 4811–4817.
- [2] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3615–3620.
- [3] Georgeta Bordea, Els Lefever, and Paul Buitelaar. 2016. Semeval-2016 task 13: Taxonomy extraction evaluation (texteval-2). In *Proceedings of the 10th international workshop on semantic evaluation (semeval-2016)*. 1081–1091.
- [4] Catherine Chen, Kevin Lin, and Dan Klein. 2021. Constructing Taxonomies from Pretrained Language Models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 4687–4700.
- [5] Nicolas Rodolfo Faucella, Alfio Gliozzo, Sarthak Dash, Md Faisal Mahbub Chowdhury, and Nandana Mihindukulasooriya. 2019. Automatic taxonomy induction and expansion. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*. 25–30.
- [6] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. 2019. Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring. In *International Conference on Learning Representations*.
- [7] Minhao Jiang, Xiangchen Song, Jieyu Zhang, and Jiawei Han. 2022. TaxoEnrich: Self-Supervised Taxonomy Completion via Structure-Semantic Representations. In *Proceedings of the ACM Web Conference 2022*. 925–934.
- [8] David Jurgens and Mohammad Taher Pilehvar. 2016. Semeval-2016 task 14: Semantic taxonomy enrichment. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*. 1092–1102.
- [9] Bang Liu, Weidong Guo, Di Niu, Jinwen Luo, Chaoyue Wang, Zhen Wen, and Yu Xu. 2020. GIANT: scalable creation of a web-scale ontology. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 393–409.
- [10] Zequn Liu, Shukai Wang, Yiyang Gu, Ruiyi Zhang, Ming Zhang, and Sheng Wang. 2021. Graphine: A Dataset for Graph-aware Terminology Definition Generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 3453–3463.
- [11] Zichen Liu, Hongyuan Xu, Yanlong Wen, Ning Jiang, Haiying Wu, and Xiaojie Yuan. 2021. TEMP: Taxonomy Expansion with Dynamic Margin Loss through Taxonomy-Paths. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 3854–3863.
- [12] Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.
- [13] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. 2016. Hierarchical question-image co-attention for visual question answering. *Advances in neural information processing systems* 29 (2016).
- [14] Mingyu Derek Ma, Muhao Chen, Te-Lin Wu, and Nanyun Peng. 2021. HyperExpan: Taxonomy Expansion with Hyperbolic Representation Learning. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. 4182–4194.
- [15] Emaad Manzoor, Rui Li, Dhananjay Shroutry, and Jure Leskovec. 2020. Expanding taxonomies with implicit edge semantics. In *Proceedings of The Web Conference 2020*. 2044–2054.
- [16] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108* (2019).
- [17] Cicero dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. 2016. Attentive pooling networks. *arXiv preprint arXiv:1602.03609* (2016).
- [18] Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *ICLR*.
- [19] Jiaming Shen, Zhihong Shen, Chenyan Xiong, Chi Wang, Kuansan Wang, and Jiawei Han. 2020. TaxoExpan: Self-supervised taxonomy expansion with position-enhanced graph neural network. In *Proceedings of The Web Conference 2020*. 486–497.
- [20] Kunihiro Takeoka, Kosuke Akimoto, and Masafumi Oyamada. 2021. Low-resource taxonomy enrichment with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2747–2758.
- [21] Yanchao Tan, Carl Yang, Xiangyu Wei, Chaochao Chen, Longfei Li, and Xiaolin Zheng. 2022. Enhancing recommendation with automated tag taxonomy construction in hyperbolic space. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. IEEE, 1180–1192.
- [22] Cunhao Tu, Han Liu, Zhiyuan Liu, and Maosong Sun. 2017. Cane: Context-aware network embedding for relation modeling. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1722–1731.
- [23] Suyuchen Wang, Ruihui Zhao, Xi Chen, Yefeng Zheng, and Bang Liu. 2021. Enquire one’s parent and child before decision: Fully exploit hierarchical structure for self-supervised taxonomy expansion. In *Proceedings of the Web Conference 2021*. 3291–3304.
- [24] Suyuchen Wang, Ruihui Zhao, Yefeng Zheng, and Bang Liu. 2022. QEN: Applicable Taxonomy Completion via Evaluating Full Taxonomic Relations. In *Proceedings of the ACM Web Conference 2022*. 1008–1017.
- [25] Pengcheng Yang, Zhihan Zhang, Fuli Luo, Lei Li, Chengyang Huang, and Xu Sun. 2019. Cross-modal commentator: Automatic machine commenting based on cross-modal information. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2680–2686.
- [26] Shuo Yang, Lei Zou, Zhongyuan Wang, Jun Yan, and Ji-Rong Wen. 2017. Efficiently answering technical questions—a knowledge graph approach. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- [27] Xiaoxin Yin and Sarthak Shah. 2010. Building taxonomy of web search intents for name entity queries. In *Proceedings of the 19th international conference on World wide web*. 1001–1010.
- [28] Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohamad Norouzi, and Quoc V Le. 2018. QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. In *ICLR*.
- [29] Yue Yu, Yinghao Li, Jiaming Shen, Hao Feng, Jimeng Sun, and Chao Zhang. 2020. Steam: Self-supervised taxonomy expansion with mini-paths. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 1026–1035.
- [30] Qingkai Zeng, Jinfeng Lin, Wenhao Yu, Jane Cleland-Huang, and Meng Jiang. 2021. Enhancing taxonomy completion with concept generation via fusing relational representations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 2104–2113.
- [31] Jieyu Zhang, Xiangchen Song, Ying Zeng, Jiaye Chen, Jiaming Shen, Yuning Mao, and Lei Li. 2021. Taxonomy completion via triplet matching network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 4662–4670.
- [32] Yuchen Zhang, Amr Ahmed, Vanja Josifovski, and Alexander Smola. 2014. Taxonomy discovery for personalized recommendation. In *Proceedings of the 7th ACM international conference on Web search and data mining*. 243–252.
- [33] Zhihan Zhang, Zhiyi Yin, Shuhuai Ren, Xinhang Li, and Shicheng Li. 2020. Dca: Diversified co-attention towards informative live video commenting. In *CCF International Conference on Natural Language Processing and Chinese Computing*. Springer, 3–15.