STEAM: Self-Supervised Taxonomy Expansion with Mini-Paths

Yue Yu[1], Yinghao Li[1], Jiaming Shen[2], Hao Feng[3], Jimeng Sun[2], Chao Zhang[1]

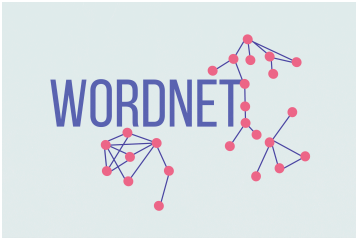
[1]Georgia Institute of Technology

[2]University of Illinois at Urbana-Champaign

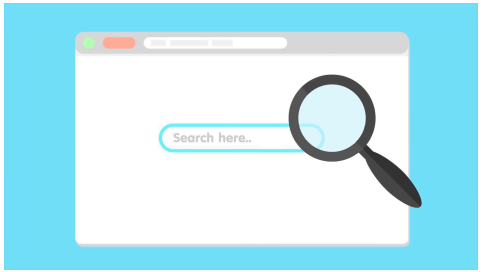
[3] University of Electronic Science and Technology of China

Introduction

- Taxonomies are ubiquitous in real life



- Taxonomies facilitate various downstream applications



Web Search

Product Recommendation

Disease Analytics

Why Taxonomy Expansion?

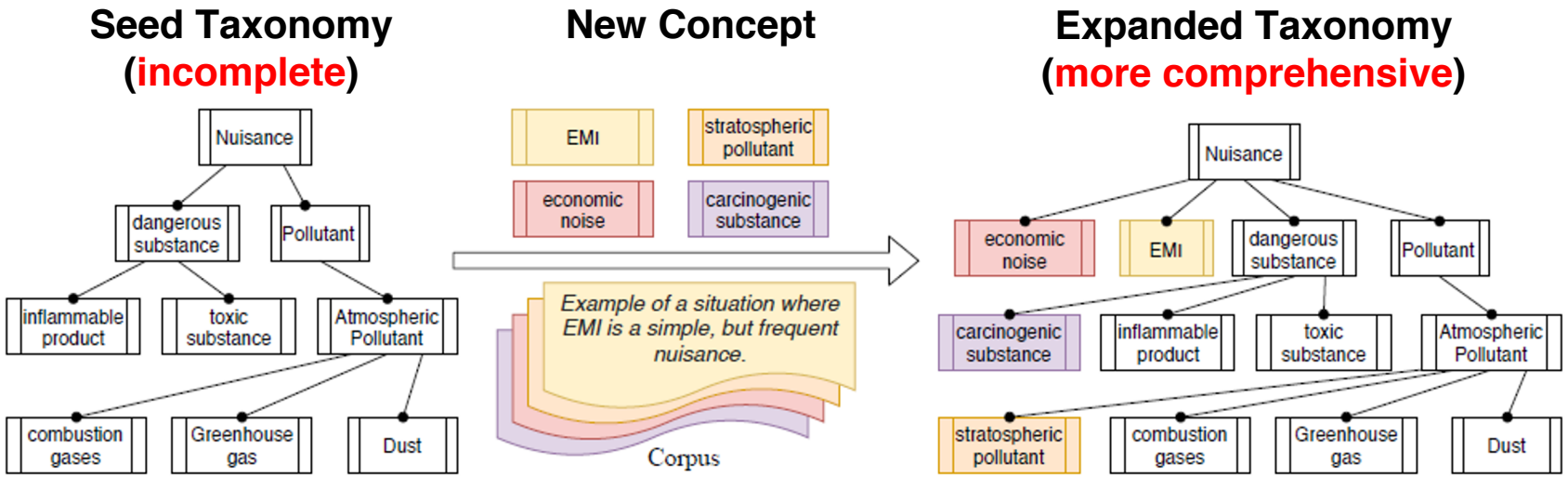
- Domain-specific knowledge is constantly growing, but it is too tedious to rely on human curation

- It is crucial to dynamically maintain and expand the existing taxonomies with the emerging terms

Taxonomy Expansion

Input: An *initial*, *incomplete* seed taxonomy[1] and a set of *new* concepts.

Task: Enrich the incomplete taxonomy by *inserting new concepts* into it.



[1]each node in taxonomy represents a concept

Prior Art on Taxonomy Expansion

- Taxonomy Construction Methods [1,2]:
  - Construct taxonomies from scratch – Time Consuming;
  - Cannot preserve the initial taxonomy structures curated by domain experts.

- Unsupervised Taxonomy Expansion Methods [3,4]:
  - Fail to leverage signals in seed taxonomy;
  - Require in-domain expertise for feature selection.

- Self-supervised Taxonomy Expansion Methods [5,6]:
  - Leverage the *parent-child* relationships in existing taxonomy;
  - Only consider the distributed term embeddings as features.

[1] Wang *et al.* A phrase mining framework for recursive construction of a topical hierarchy. KDD'13.
[2] Panchenko *et al*. TAXI: a Taxonomy Induction Method based on Lexico-Syntactic Patterns, Substrings and Focused Crawling. SemEval'16.
[3] Fauceglia *et al*. Automatic Taxonomy Induction and Expansion. EMNLP'19.
[4] Vedula *et al*. Enriching taxonomies with functional domain knowledge. SIGIR'18.
[5] Shen *et al*. TaxoExpan: Self-supervised Taxonomy Expansion with Position Enhanced Graph Neural Network. WebConf'20.
[6] Manzoor et al. Expanding Taxonomies with Implicit Edge Semantics. WebConf'20.

Challenges

- How to better leverage the hierarchical structure in the seed taxonomy?
  - Traditional methods expand the taxonomy via finding hypernym pairs for anchor terms, while neglecting the whole structure of existing taxonomy.

- How to effectively integrate multiple kinds of supervision signals to boost the performance?
  - Traditional methods fail to consider multiple type of information. Distributed term embeddings or features are insufficient to identify the hypernym relationships.

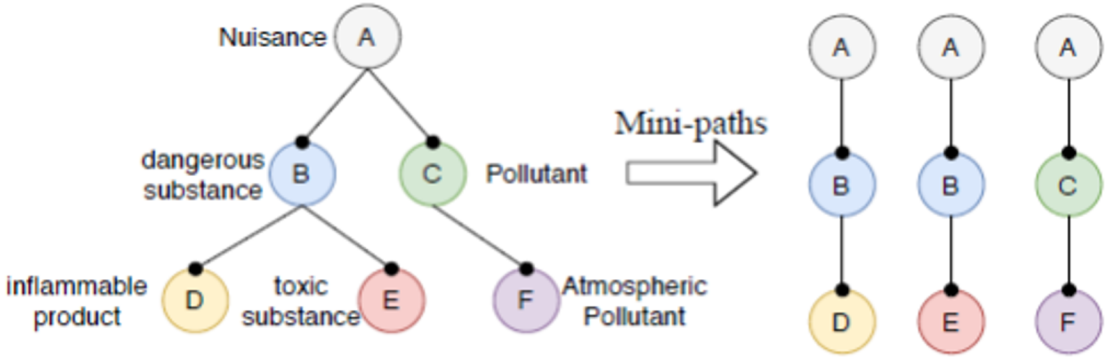
Our Solution

- Creates self-supervised training data by sampling mini-path-based query-anchor pairs from the existing taxonomy.
  - Leverage the hierarchical structure in the seed taxonomy

- Use co-training to aggregate information from multiple views between terms to identify the relationship between terms.
  - Distributed Features
  - Contextual Features
  - Lexico-syntactic Features

Self-supervised training data generation

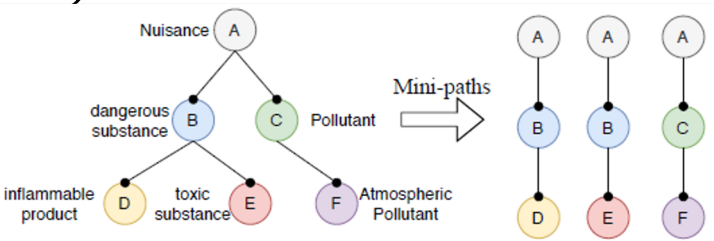
- Mini-path-based training sample generation: sample terms from L different layers of taxonomy to construct mini-paths with length L.
  - Preserving the hierarchical structure of terms (more than parent-child relations)



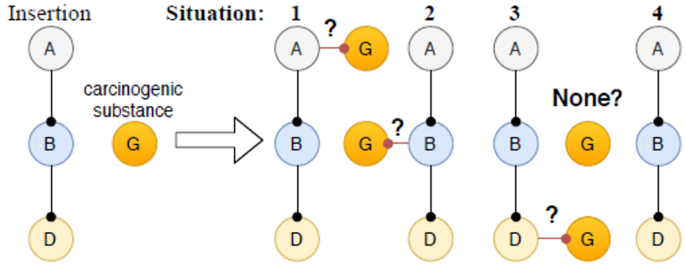
(a) An illustration of mini-paths.

Self-supervised training data generation

- Training Sample Generation from Seed Taxonomy: sampling query terms q and mini-paths P with relative position l.
- Example for mini-path $P = \langle A, B, D \rangle$ $(L = 3)$:
  - For term $q = C$, $l = 0$;
  - For term $q = E$, $l = 1$;
  - For term $q = F$, $l = 3$.
- Build Training Set
  - $X_1 = \langle q = C, P = \langle A, B, D \rangle, l = 1 \rangle$
  - $X_2 = \langle q = E, P = \langle A, B, D \rangle, l = 2 \rangle$
  - $X_3 = \langle q = F, P = \langle A, B, D \rangle, l = 4 \rangle$
  - Keep Negative Sampling rate r.



(a) An illustration of mini-paths.



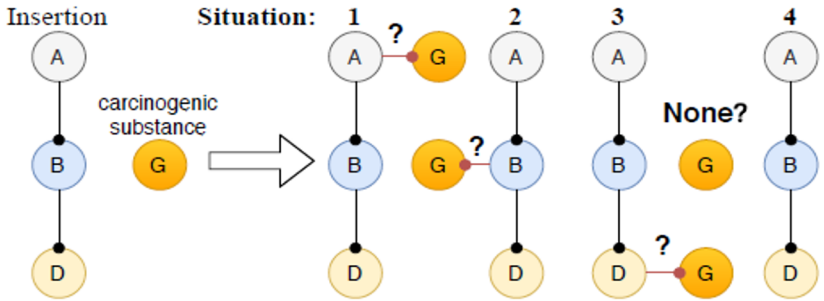
(b) The classification target.

Self-supervised training data generation

- To train the model, sample training set $X = \langle q, P, l \rangle$. For query terms q and mini-path P, predict their relative position l.
- Training Objective:

$$loss = -\sum_{X}\sum_{1}^{L+1} y_i \log \hat{y}_i$$

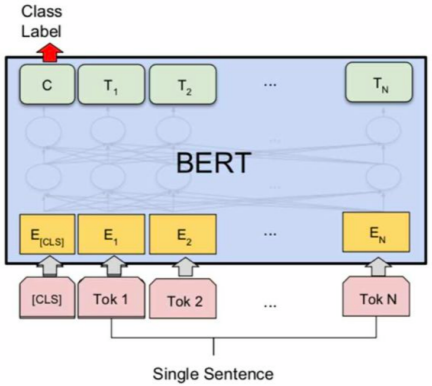
- y_i: Ground Truth Position, $\hat{y}_i$: Predicted Position

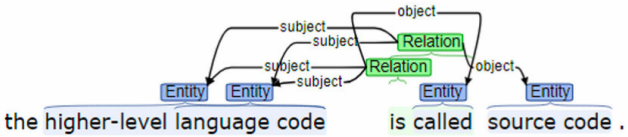


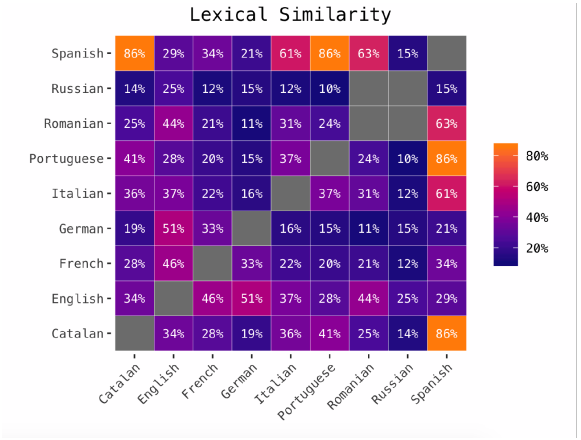
(b) The classification target.

Multi-view Co-training: Feature Extraction

- Three sets of features from three views:







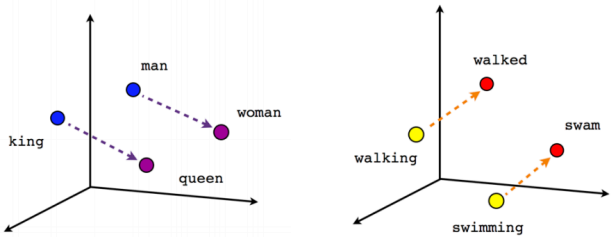
1. Distributed features
word embedding similarities

2. Contextual features
from dependency parses

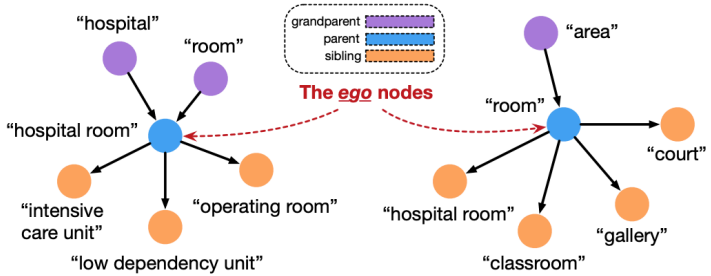
3. Lexical similarities
From linguistic patterns

Multi-view Co-training: Feature Extraction

- **View 1: Distributed Features $\mathbf{h}_d$ [1,2]**
  - Use BERT embeddings for each term as initialized representations.



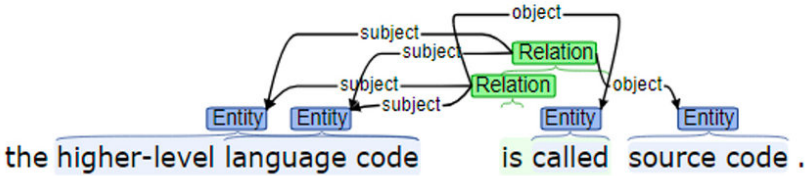
  - Use position-enhanced graph attention network (PGAT) to propagate the embeddings for the terms in the seed taxonomy [3].



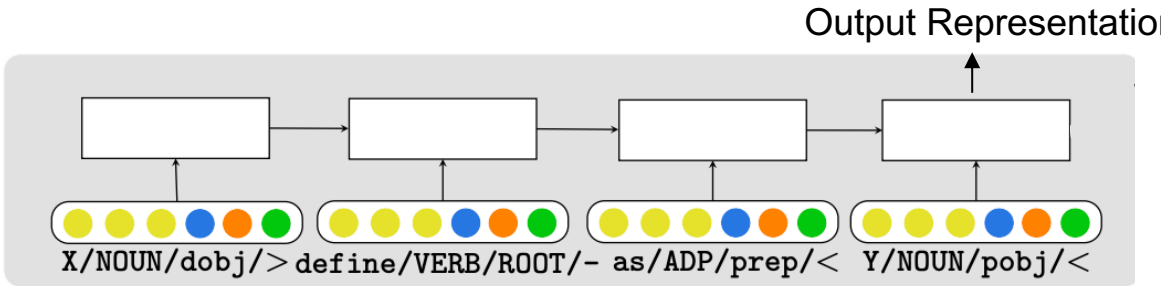
[1] Fu *et al*. Learning semantic hierarchies via word embeddings. ACL'14.
[2] Baroni *et al*. Entailment above the word level in distributional semantics. EACL'12.
[3] Shen *et al*. TaxoExpan: Self-supervised Taxonomy Expansion with Position Enhanced Graph Neural Network. WebConf'20.

Multi-view Co-training: Feature Extraction

- **View 2: Contextual Features $\mathbf{h}_c$ [1]**
  - Extract the dependency paths between term pairs from co-occur sentences.



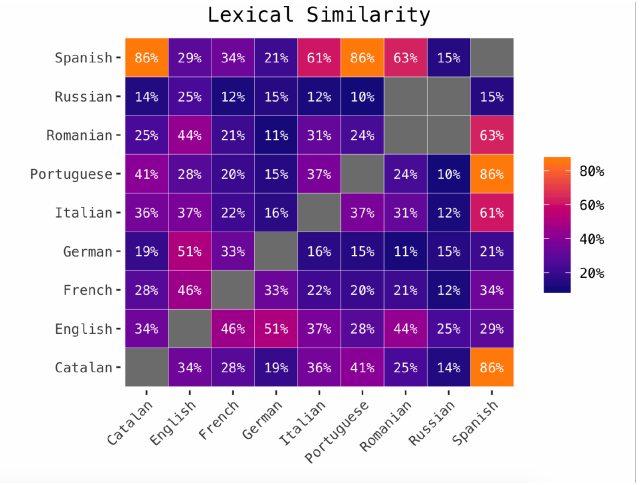
  - Feed the dependency path (a sequence) to an LSTM encoder, and use the representation of the LSTM's last hidden layer as its representation.



[1] Shwartz *et al*. Improving Hypernymy Detection with an Integrated Path-based and Distributional Method. ACL'16.

Multi-view Co-training: Feature Extraction

- **View 3: Lexico-syntactic Features $\mathbf{h}_s$ [1]**
  - Manually select lexical-syntactic features as follows:
    - Ends with
    - Contains
    - Suffix match
    - Longest Common Substring
    - Length Difference
    - Normalized Frequency Difference
  - Concatenate the score for each feature as the final representation.



[1] Zhang *et al*. Learning Concept Taxonomies from Multi-modal Data. ACL'16.

Multi-view Co-training: Feature Aggregation

- Each View has its own advantage and disadvantages

Features	Pros ☺	Cons 😐
Distributed Features $\mathbf{h}_d$	High coverage over terms	Not precise enough
Contextual Features $\mathbf{h}_c$	Capture the relation information between two terms	Limited coverage over term pairs
Lexico-syntactic Features $\mathbf{h}_s$	Encode linguistic information well for matched terms	Too rigid to cover all patterns; Have limited coverage

- It is necessary to combine different features together to aggregate the complementary signals.

Multi-view Co-training: Feature Aggregation

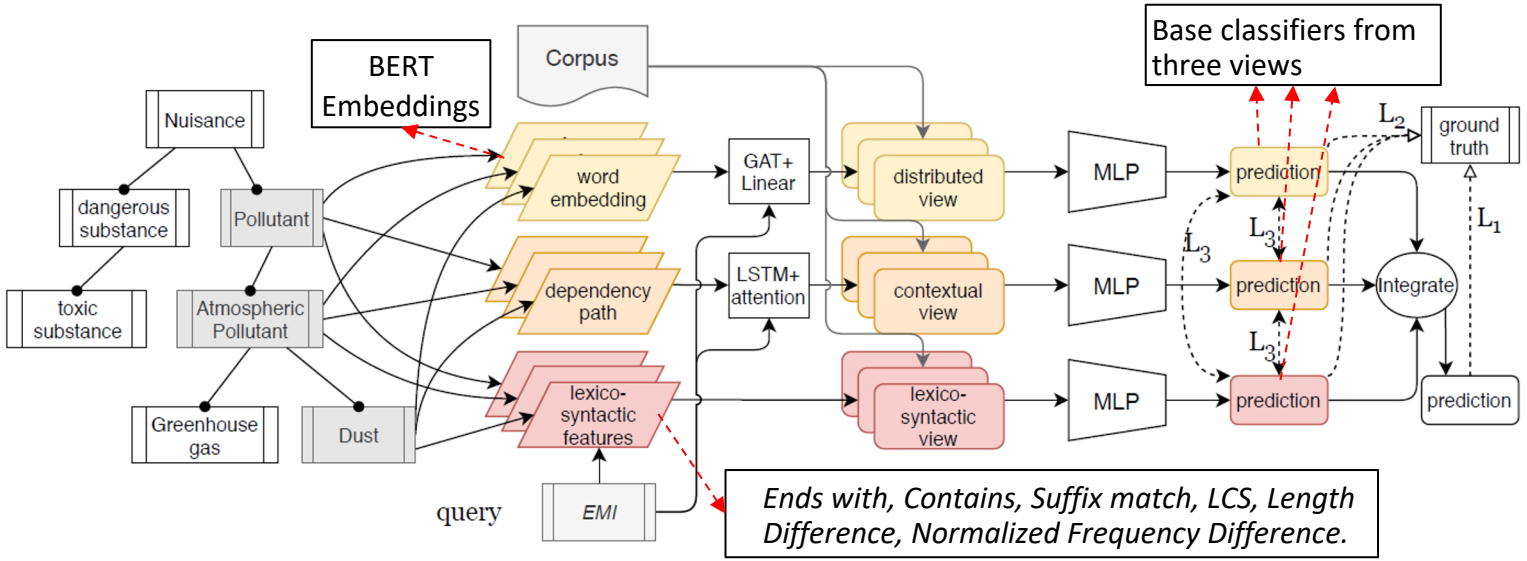
- Simply concatenating $\mathbf{h}_d, \mathbf{h}_c, \mathbf{h}_s$ is not good enough:
  - The three views have different dimensionality and distributions;
  - One view can provide dominant signals over the other two views.
- Make predictions $\mathbf{y}_d, \mathbf{y}_c, \mathbf{y}_s$ based on three views as

$$\mathbf{y}^d = f_d(\mathbf{h}_d) = \mathbf{W}_2^d(\sigma(\mathbf{W}_1^d\mathbf{h}_d + \mathbf{b}_1^d) + \mathbf{b}_2^d),$$
$$\mathbf{y}^c = f_c(\mathbf{h}_c) = \mathbf{W}_2^c(\sigma(\mathbf{W}_1^c\mathbf{h}_c + \mathbf{b}_1^c) + \mathbf{b}_2^c),$$
$$\mathbf{y}^s = f_c(\mathbf{h}_s) = \mathbf{W}_2^s(\sigma(\mathbf{W}_1^s\mathbf{h}_s + \mathbf{b}_1^s) + \mathbf{b}_2^s),$$

- Aggregating three base classifiers by averaging over their predictions

$$\mathbf{y}^{\text{agg}} = f_{\text{agg}}\left(\mathbf{y}^d, \mathbf{y}^c, \mathbf{y}^s\right) = \text{softmax}\left(\frac{1}{3}\left(\mathbf{y}^d + \mathbf{y}^s + \mathbf{y}^r\right)\right)$$

Multi-view Co-training: Objective



- Co-training Objective **Overall Loss** $\ell = \ell_1 + \lambda\ell_2 + \mu\ell_3$

$$\ell_1 = -\sum_{i=1}^{N}\sum_{j=1}^{C} \mathrm{y}_{ij} \log \mathrm{y}_{ij}^{\mathrm{agg}}$$

Loss for the final prediction

$$\ell_2 = -\sum_{u\in\{d,c,s\}}\sum_{i=1}^{N}\sum_{j=1}^{C} \mathrm{y}_{ij} \log \mathrm{y}_{ij}^{u}$$

Loss for three base classifiers

$$\ell_3 = \sum_{u,v\in\{d,s,r\}}\sum_{i=1}^{N} \left\|\mathrm{y}_i^u - \mathrm{y}_i^v\right\|^2$$

Consistency Loss for three base classifiers

Model Inference

- Given a new query term $q \in$ C, we traverse all the mini-paths P and calculate the scores for all anchor terms $p \in P$ based on the aggregated final prediction score y.

- Average over all paths to obtain the final score for anchor term $\hat{p}$

$$y_{\hat{p}} = \frac{1}{|\hat{\mathcal{P}}|} \sum_{P \in \hat{\mathcal{P}}} y^{P}_{q,\hat{p}},$$

- Rank all anchor terms and select the term p^* with the highest score as predicted parent for q

$$p^* = \arg\max_{p \in \mathcal{V}_0} y_p.$$

Experiments

- Evaluate over three taxonomies in SemEval 2016: Environment, Science and Food. Use 20% leaf terms as test data and other 80% terms as existing taxonomy.

Dataset	Environment			Science			Food		
Metric	Acc	MRR	Wu&P	Acc	MRR	Wu&P	Acc	MRR	Wu&P
BERT+MLP	11.1	21.5	47.9	11.5	15.7	43.6	10.5	14.9	47.0
TAXI	16.7	-	44.7	13.0	-	32.9	18.2	-	39.2
HypeNet	16.7	23.7	55.8	15.4	22.6	50.7	20.5	27.3	63.2
TaxoExpan	11.1	32.3	54.8	27.8	44.8	57.6	27.6	40.5	54.2
STEAM	**36.1**	**46.9**	**69.6**	**36.5**	**48.3**	**68.2**	**34.2**	**43.4**	**67.0**

- **BERT+MLP**: Based on distributional features;
- **TAXI**: Taxonomy construction method w./ heuristic patterns;
- **HypeNet**: hypernym extraction method w./ distributional and contextual features.
- **TaxoExpan**: SOTA self-supervised taxonomy expansion method based on distributional features & PGAT.

Ablation Study: Effect of Co-Training

Concat three features

w.o. information from some view

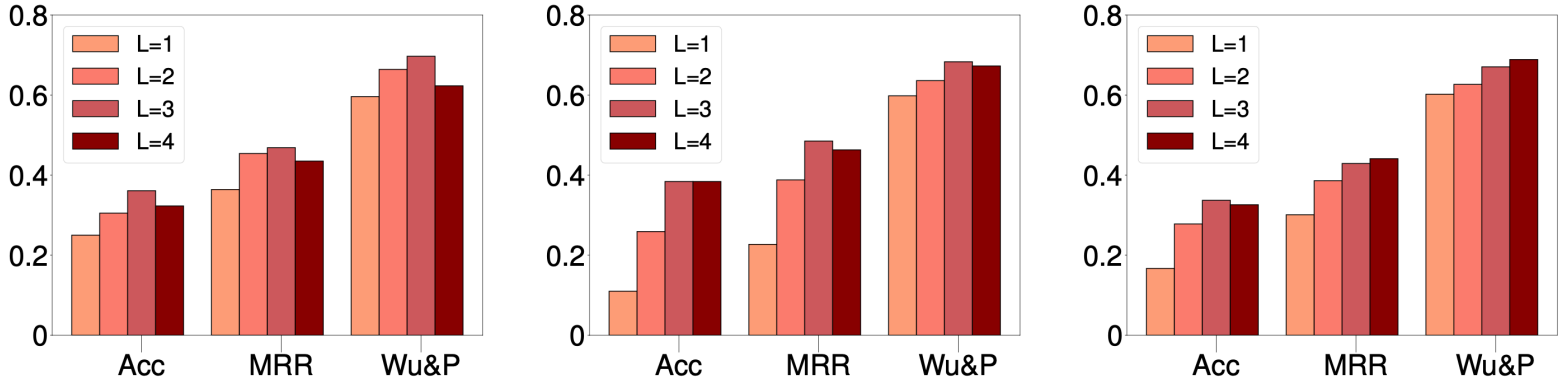
w.o. Co-training

w.o. information from some view

Dataset	Environment			Science			Food		
Metric	Acc	MRR	Wu&P	Acc	MRR	Wu&P	Acc	MRR	Wu&P
CONCAT	25.0	40.3	64.2	20.4	25.8	51.1	15.5	23.8	49.6
CONCAT-D	30.6	38.6	63.7	11.1	20.1	48.1	23.1	28.9	55.4
CONCAT-C	27.7	37.4	57.8	13.5	25.7	53.3	25.3	31.2	58.3
CONCAT-L	11.1	31.4	57.7	13.5	23.7	39.1	8.30	13.4	40.1
STEAM-Co	25.0	41.0	66.3	32.7	45.3	64.4	31.1	40.7	65.1
STEAM-D	13.8	32.0	54.3	23.1	32.9	60.0	20.1	31.5	60.8
STEAM-C	11.1	26.8	49.2	32.7	44.5	67.2	19.3	29.7	59.3
STEAM-L	11.1	27.5	51.6	23.1	36.5	62.1	12.7	22.6	56.7
STEAM	**36.1**	**46.9**	**69.6**	**36.5**	**48.3**	**68.2**	**34.2**	**43.4**	**67.0**

1. Simple concat cannot fully harvest the information from each view.
2. Co-training encourages better learning of base classifiers.
3. All views contribute to the final performance.

Ablation Study: Effect of Path Length L



(a) Environment (b) Science (c) Food

- The performance of STEAM stably increases with L when L is small.
- When L increase from 3 to 4, the performance gain becomes small. There is even performance drop on Environment Dataset.

Case Study

Gold Parent: Physics		
View	Score	Rank
Distributed	0.812	11
Contextual	0.947	12
Lexico-syntactic	0.640	15
STEAM Output	**0.799**	**1**

(a) term **Electrostatics** (SCI)

Gold Parent: Fruit Juice		
View	Score	Rank
Distributed	0.720	25
Contextual	0.921	14
Lexico-syntactic	0.656	15
STEAM Output	**0.765**	**1**

(b) term **Nectar** (Food)

Gold Parent: Mammal		
View	Score	Rank
Distributed	0.416	116
Contextual	0.987	1
Lexico-syntactic	0.615	31
STEAM Output	**0.672**	**1**

(c) term **Whale Marine** (EN)

Gold Parent: Medicine		
View	Score	Rank
Distributed	0.741	51
Contextual	0.959	2
Lexico-syntactic	0.614	14
STEAM Output	**0.771**	**1**

(d) term **Podiatry** (SCI)

Gold Parent: Red Wine		
View	Score	Rank
Distributed	0.468	169
Contextual	0.493	24
Lexico-syntactic	0.329	228
STEAM Output	**0.430**	**43**

(e) term **Chianti** (Food)

Gold Parent: Sea Bed		
View	Score	Rank
Distributed	0.387	35
Contextual	0.568	22
Lexico-syntactic	0.483	127
STEAM Output	**0.479**	**37**

(f) term **Inshore Grounds** (EN)

(a, b): our model can integrate the weak signals to rank the ground-truth to top.

(c, d): our model can rectify the noisy signal by leveraging information from the other views.

(e, f): our model cannot perform well when all views cannot make accurate predictions.

Summary

- We propose a method STEAM for self-supervised taxonomy expansion
  - Design mini-path structure to well preserve the self-supervision signals in the existing taxonomy.
  - Use co-training to effectively aggregate multi-view features.
- Future Work
  - Only consider leaf terms in taxonomy expansion now. How to consider more complex scenarios in taxonomy expansion (e.g. inserting non-leaf nodes)?
  - The taxonomies in our experiments are small. How to improve the efficiency and scalability of our method?
- Code and Data
  - https://github.com/yueyu1030/STEAM

Thank you!

Questions?