TEMP: Taxonomy Expansion with Dynamic Margin Loss through Taxonomy-Paths

Zichen Liu[1], **Hongyuan Xu**[1], **Yanlong Wen**[1] *, **Ning Jiang**[2], **Haiying Wu**[2], **Xiaojie Yuan**[1]
[1]TKLNDST, College of Computer Science, Nankai University, China
[2]Mashang Consumer Finance Co, Ltd
{liuzichen, xuhongyuan, wenyl, yuanxj}@dbis.nankai.edu.cn
{ning.jiang, haiying.wu02}@msxf.com

Abstract

As an essential form of knowledge representation, taxonomies are widely used in various downstream natural language processing tasks. However, with the continuously rising of new concepts, many existing taxonomies are unable to maintain coverage by manual expansion. In this paper, we propose TEMP, a self-supervised taxonomy expansion method, which predicts the position of new concepts by ranking the generated taxonomy-paths. For the first time, TEMP employs pre-trained contextual encoders in taxonomy construction and hypernym detection problems. Experiments prove that pre-trained contextual embeddings are able to capture hypernym-hyponym relations. To learn more detailed differences between taxonomy-paths, we train the model with dynamic margin loss by a novel dynamic margin function. Extensive evaluations exhibit that TEMP outperforms prior state-of-the-art taxonomy expansion approaches by 14.3% in accuracy and 15.8% in mean reciprocal rank on three public benchmarks.

1 Introduction

Taxonomies, tree-structured semantic hierarchies that organize entities by hypernym-hyponym (*is-a*) relations, play an important role in many NLP tasks such as question answering (Yang et al., 2017), query understanding (Hua et al., 2016) and information extraction (Demeester et al., 2016).

Manually curated taxonomies usually face the limited coverage issue, especially when new concepts arise continuously. A low coverage taxonomy can largely hurt the performance of downstream tasks relied on it. Moreover, for maintaining and expanding existing taxonomies, the curation process that requires domain experts is expensive and time-consuming. Thus, we study the automatic taxonomy expansion task (Figure 1): given an existing taxonomy, a text corpus, and a set of concepts, the goal is to expand the taxonomy by inserting concepts into it.

Two common strategies used to study *taxonomy construction and expansion* are pattern-based methods (e.g. the Hearst pattern (Hearst, 1992)) and distributional methods (Yu et al., 2015). Recent evidence suggests that the semantic information or structural features encoding in the representation is an effective way to solve the task, especially probability statistics from a large corpus (Mikolov et al., 2013), semantic information extracted from text data (Yin and Roth, 2018), and properties of hypernym-hyponym relations such as strict partial order (Dash et al., 2020).

Since taxonomies can be formulated as *directed acyclic graphs* (DAGs), the graph structure has been seen as important information for taxonomy expansion and construction in recent works (Shang et al., 2020; Shen et al., 2020). However, according to our observation, the path composed of ancestor nodes in a taxonomy is a more appropriately encoded object in hypernym-hyponym relations. In a tree-structured taxonomy, all the ancestor nodes have an *is-a* relation with the child node. In Figure 1, for example, "*Science*" - "*Systematics*" - "*Biosystematics*" is the taxonomy-path of word "*Biosystematics*". "*Biosystematics*" not only "*is-a*" "*Systematics*" but also "*is-a*" "*Science*". In addition, the serial structure of taxonomy-path is also more appropriate than the graph structure for transformers to encode.

As far as we know, there has been no attempt to take pre-trained contextual encoders (such as BERT (Devlin et al., 2019)) as the core of taxonomy expansion or construction model. Pre-trained contextual encoders have been proved powerful in various NLP tasks such as Question Answer(QA)(Yang et al., 2019), Information Retrieval (IR) (Nogueira and Cho, 2019), Document Classification (Adhikari et al., 2019), etc. Compared with previous encoding approaches, pre-trained contextual en-

* Corresponding author

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3854–3863
November 7–11, 2021.

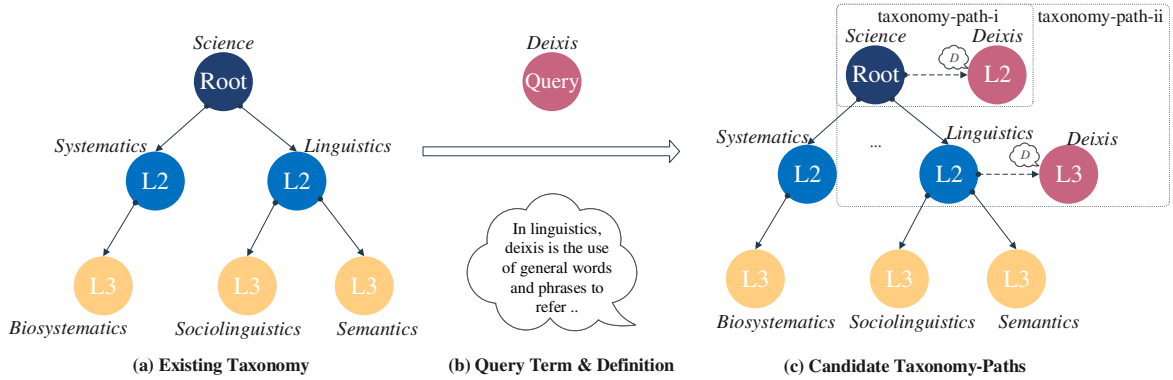


Figure 1: An example of expanding taxonomy. The dash boxes outline two candidates of all possible taxonomy-paths to be predicted.

coders have two main advantages. First, they are capable of deeply encoding textual content and capturing long distance dependencies. Second, most of them have been pre-trained on large text corpora to naturally support tasks using text features. Our proposed method, TEMP[1], is the first to show that fine-tuned pre-trained contextual encoders are able to identify hypernym-hyponym relations. To enhance the understanding of concepts, the model takes the query concept's definition as input besides the taxonomy-path.

The diversity and heterogeneity of hypernym-hyponym relations is another reason for the difficulty of expanding taxonomies (Fu et al., 2014; Manzoor et al., 2020), which makes it hard for the model to learn the similarities and differences between relations on limited datasets. Inspired by the success of ARBORIST (Manzoor et al., 2020), we train the model with dynamic margin ranking loss (MRL) to handle this problem. Previous studies show that margin loss can optimize the model to learn the discriminative deep features (Lin and Xu, 2019) and that dynamic margins set by handcrafted rules can lead the model to learn more similarity information (Feng et al., 2020). Therefore, we design a margin function to calculate the margin between taxonomy-paths based on their semantic similarity.

Contributions. In summary, our major contributions include:

- We propose TEMP, a self-supervised taxonomy expansion method, that is the first to take contextual encoders (such as BERT) as the core of the model for the taxonomy expansion problem.
- We employ the dynamic margin-based ranking loss with a novel dynamic margin function in the TEMP to make the model learn the discriminative difference between taxonomy-paths.
- We take word definitions and taxonomy-paths generated in the existing taxonomy as the input of our model, which means that TEMP doesn't require large-scale corpora but only the definitions of concepts.

Experiments on three benchmarks show that TEMP improves the previous state-of-the-art performance by 14.3% in accuracy and 15.8% in mean reciprocal rank on average.

2 Related Work

Automatic taxonomy construction has been a long-term task in literature in the last few decades. Most existing methods follow the paradigm of constructing taxonomy from scratch. They firstly extract < Hypernym, Hyponym > pairs from raw resources (Gupta et al., 2017) and organize them into a noisy hierarchy to further prune it via constraints like DAG (Fu et al., 2014; Liang et al., 2017b,a). These approaches exploit semantic information and structural features such as lexical-patterns (Hearst, 1992; Nakashole et al., 2012) or distributional embeddings (Yu et al., 2015; Shwartz et al., 2016; Le et al., 2019; Wang and He, 2020) to automatically construct taxonomies. However, recent practical applications have revealed that it is laborious to construct taxonomies from scratch when facing the continuously rising of new concepts, so solutions to the taxonomy expansion task are in urgent need.

[1]short for **T**axonomy **E**xpansion with Dynamic **M**argin Loss through Taxonomy-**P**aths

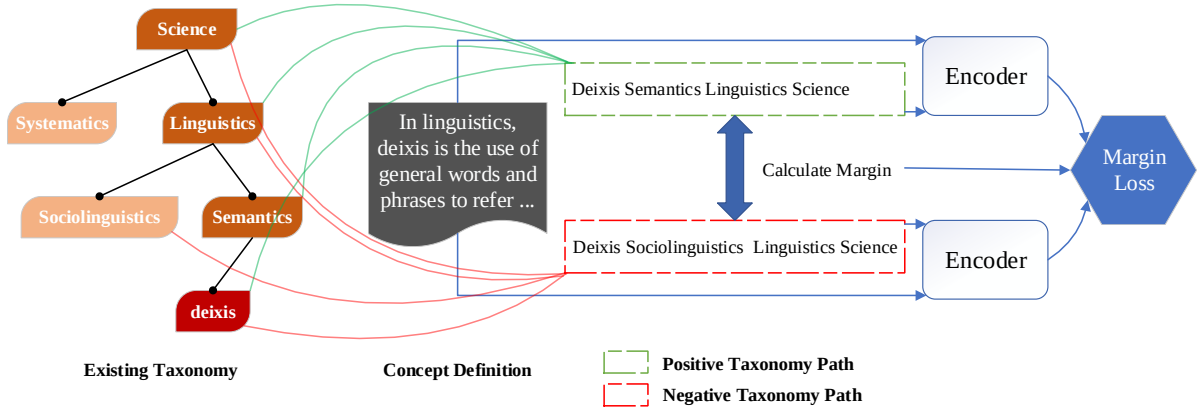


Figure 2: Overview of TEMP structure. Given one node in the existing taxonomy, a pair of positive and negative taxonomy-paths is generated. Through the generated taxonomy-paths, dynamic margin function returns a margin.

Recently, numerous methods have been proposed to solve the aforementioned problem (Shen et al., 2018; Mao et al., 2020; Shen et al., 2020; Yu et al., 2020; Manzoor et al., 2020; Zhang et al., 2021). For example, Shen et al. (2020) proposes a position-enhanced graph neural network framework to encode the local structure of an anchor concept with a noise-robust training objective. Yu et al. (2020) converts candidate anchor positions from the whole existing taxonomy to mini-paths, in which way can better capture and integrate multiple sources of information via a multi-view co-training procedure. Manzoor et al. (2020) first designs a realistic approach to demonstratively model unobserved and heterogeneous edge semantics. Zhang et al. (2021) generalizes expansion task to the more general "one-to-pair" completion task and applies primal and auxiliary scorers based on the neural tensor network to rank candidate anchor positions.

As far as we know, all the existing methods attempt to determine the attachment position by scoring between several nodes, we are the first to take the path as the unit for encoding and calculating scores. Besides, to the best of our knowledge, few state-of-the-art expansion approaches encode information out of supervision information in the existing taxonomy, we take pre-trained contextual encoder as core to aggregate more valuable information and resources such as word definition to improve performance.

3 The TEMP Method

In this section, we describe our proposed method TEMP. First, we introduce taxonomy-path, an important concept in our method(Section 3.1). Our model takes concept definitions and taxonomy-paths as input and relies on the pre-trained contextual encoders as its core (Section 3.2). The parameters of the model are trained by margin ranking loss (MRL) with a dynamic margin function designed for taxonomy expansion (Section 3.3). Finally, we discuss how to sample self-supervision data and fine-tune the model with dynamic margin loss (Section 3.4).

3.1 Taxonomy Paths

The essence of taxonomy expansion is to attach a new concept to the correct position in the existing taxonomy. Therefore, most previous works (Shen et al., 2020; Manzoor et al., 2020; Shen et al., 2018) treat this task as finding the optimum hypernym node for the new concept by measuring the taxonomic relatedness of candidate node-pairs. However, in taxonomies, not only the directed attached node has a hypernym relation with the new concept but also every ancestor node of it does. To preserve more comprehensive information, TEMP finds the correct position by evaluating the generated taxonomy-paths.

Taxonomy-Path: A taxonomy-path P = [root, n_1, n_2, ..., n_D], where D is the depth of of n_D, root is the root node in the taxonomy. n_{i-1} is the parent node of n_i in the taxonomy.

In a tree-structured taxonomy, each node has its unique corresponding taxonomy-path. For a new term, the framework generates the same number of candidate taxonomy-paths as nodes in the existing taxonomy. Then, TEMP ranks all the candidate

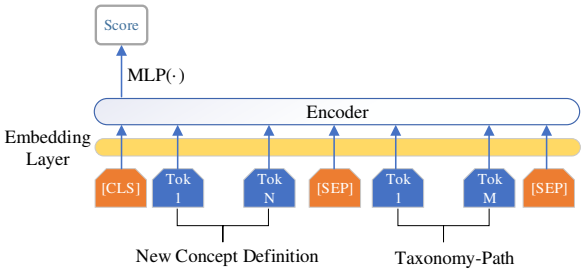


Figure 3: The backbone of TEMP

taxonomy-paths by scoring each of them.

3.2 Model Backbone

We use the pre-trained contextual encoder as the backbone of our model. We exploit the model to encode the definition of the last node in the taxonomy-path besides the taxonomy-path such that the model can capture more semantic information of the query term. The text encoding of TEMP refers to the encoding way in question answering task, with word definition as the question and taxonomy-path as the passage. Take the WordPiece tokenization (Schuster and Nakajima, 2012) used by BERT as an example, to be in line with contextual encoders, the words in taxonomy-path P and the definition sentence S of the last node are concatenated to form the input string as shown in Figure 3. Given the input string, the contextual encoder returns a sequence of vectors:

$$\text{Encoder}(S, P) = v_{[\text{CLS}]}, v_1, ..., v_{[\text{SEP}]}, v_{p_d}, ..., v_{\text{root}}$$

where $v_{[CLS]}$ is the represention vector of the special [CLS] token. We feed the [CLS] represention into a multilayer perceptron (MLP) output layer to evaluate the taxonomy-path.

Compared with the previous methods (Shwartz and Dagan, 2016; Panchenko et al., 2016; Yu et al., 2020) that normally designed lexical features like **Ends with**, **Contains**, **Suffix match**, **Occurrence frequency**, and so on, we believe that contextual encoders are sufficient to obtain the hierarchical information for the following two reasons: (1) Contextual encoders use subword algorithms for text encoding, such as WordPiece (Schuster and Nakajima, 2012) and Byte-Pair Encoding (Sennrich et al., 2016). So after the taxonomy-path is tokenized, the substring information among terms is intuitively showed to the model. (2) Contextual encoders are pre-trained in large corpora, which makes them empirically powerful even without explicit frequency information.

3.3 Dynamic Margin Loss

We train the model with Margin Ranking Loss (MRL) such that the optimum taxonomy-path is ranked higher than others. Margin Ranking Loss is defined as follows:

$$\mathcal{L} = \sum_{P \in \mathcal{P}^+} \sum_{P' \in \mathcal{P}^-} max(0, f(P') - f(P) + \gamma(P, P')) \quad (1)$$

where $\mathcal{P}^+$ is the set of taxonomy-paths in the taxonomy, $\mathcal{P}^-$ is the set of negative samples, and $\gamma(P, P')$ is a function designed for the margin between positive and negative taxonomy-paths. In traditional MRL, the output of the margin function is a constant value, which is manually set via cross-validation. All the negative taxonomy-paths will be roughly scored the same, which ignores the subtle similarity that is proved useful in both face recognition (Feng et al., 2020) and lexical entailment (Manzoor et al., 2020). To capture the semantic similarity of different taxonomy-paths, we set a dynamic margin function based on the semantic similarity as follows:

$$\gamma(P, P') = (\frac{|P \cup P'|}{|P \cap P'|} - 1) * k \quad (2)$$

where k is a parameter used to adjust margins (usually between 0.1 and 1).

This function is inspired by the word meaning similarity measure proposed by Wu and Palmer (1994). In a tree-structured taxonomy, the intersection of two different taxonomy-paths is the set of common super-concepts at the beginning of both paths. Minimizing the loss also minimizes the number of different nodes between the highest-ranked prediction and the true taxonomy-path. Therefore, the training with the margin function encourages negative taxonomy-paths that are more irrelevant to the last nodes in them to get a lower score. Such a design also fits the Wu&P metric which is introduced in Section 4.1.

3.4 Sampling and Training

In this section, we introduce how TEMP learns using self-supervision from the existing taxonomy.

Sampling. Figure 2 shows an example of generating self-supervision data. Given one leaf node n_q in the existing taxonomy, we take its corresponding

taxonomy-path as a positive sample. Then, we randomly select one node n_r (except its parent) with its corresponding taxonomy-path P_r in the taxonomy. By adding n_q to P_r as its last node, we obtain a negative taxonomy-path P_n. For each leaf node in the existing taxonomy, we generate a pair of positive and negative taxonomy-path. By repeating the above process (with different random choices) for each epoch, we obtain the full self-supervision dataset.

Training. When training, the mini-batch consists of pairs of samples, which means the positive and corresponding negative taxonomy-paths must be fed into the model in the same batch. With the pair of taxonomy-paths as input, the margin function returns the corresponding margin. Then, we calculate the margin loss and update the model parameters.

4 Experiments

In this section, we first introduce the experimental setup (Section 4.1) and report the overall performance compared with baselines (Section 4.2). Then, we study the effectiveness of the key choices in TEMP by ablation experiments (Section 4.3). Furthermore, we discuss the factors that can affect the performance of TEMP (Section 4.4).

4.1 Experimental Setup

Dataset	**Environment**	**Science**	**food**
$\lvert\mathcal{N}\rvert$	261	429	1486
$\lvert\mathcal{L}\rvert$	201	312	1184
$\lvert\mathcal{D}\rvert$	6	8	8
$\lvert\Delta\rvert$	3.78	5.16	5.36

Table 1: Statistics of the taxonomy datasets for evaluation. $|\mathcal{N}|$ and $|\mathcal{L}|$ are the number of nodes and leaf nodes in the taxonomy. $|\mathcal{D}|$ and $|\Delta|$ indicate the depth of the taxonomy and the average depth of leaf nodes respectively.

Datasets. We evaluate TEMP using all the three English datasets in Semeval-2016 task 13 [2] (Bordea et al., 2016) . These datasets correspond to human-curated concept taxonomies of three different domains: environment, science, food (summarized in Table 1). We follow the setup as in Yu et al. (2020) that uses the randomly-growed taxonomies for self-supervised learning and the rest 20% leaf concepts for testing.

[2]https://alt.qcri.org/semeval2016/task13/

Metrics. When testing, TEMP ranks all candidate taxonomy-paths for each test concept. For the ith node in n testing nodes, We denote the ground truth taxonomy-path and the highest-ranked taxonomy-path as y_i and $\hat{y}_i$ respectively. Following previous works (Yu et al., 2020; Shen et al., 2020; Jurgens and Pilehvar, 2016), we use these metrics:

(1) **Accuracy (Acc)** measures the counting of the exactly predicted taxonomy-path.

$$\text{Acc} = \frac{1}{n}\sum_{i=1}^{n}(y_i = \hat{y}_i)$$

(2) **Mean reciprocal rank (MRR)** calculates the average of reciprocal ranks of the true taxonomy-path.

$$\text{MRR} = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\text{rank}(y_i)}$$

(3) **Wu & Palmer similarity (Wu&P)** measures the semantic similarity between the predicted taxonomy-path and the truth taxonomy-path, calculated as

$$\text{Wu\&P} = \frac{1}{n}\sum_{i=1}^{n}\frac{2|y_i \cap \hat{y}_i|}{|y_i| + |\hat{y}_i|}$$

Baseline Methods. We compare with the following methods:

- **BERT+MLP**: A distributional method that takes terms embeddings from a pre-trained but not fine-tuned BERT and then feeds them into a Multi-Layer Perceptron (MLP) to predict their relations. The experimental results come from Yu et al. (2020).
- **TaxoExpan** (Shen et al., 2020): A self-supervised method for taxonomy expansion that adopts position-enhanced graph neural networks (GNNs) to encode local structure and InfoNCE loss for robust learning.
- **STEAM** (Yu et al., 2020): One state-of-the art taxonomy expansion framework which extracts features for query-anchor pairs from three views based on mini-path anchor format and is trained by a multi-view co-training procedure.
- **TMN** (Zhang et al., 2021): A one-to-pair matching model which leverages auxiliary and primal signals using the base model neural

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
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
TMN	35.0	43.6	54.0	41.9	53.2	75.9	34.7	47.2	65.9
TEMP-BERT	49.0	62.0	75.9	54.4	64.6	84.6	45.2	57.1	78.3
TEMP-ELECTRA	**49.2**	**63.5**	**77.7**	**57.8**	**67.5**	**85.3**	**47.6**	**60.5**	**81.0**

Table 2: Baseline comparison on the three datasets (in %).

Dataset	Environment			Science			Food		
Metric	Acc	MRR	Wu&P	Acc	MRR	Wu&P	Acc	MRR	Wu&P
No Definition	48.7	61.9	71.9	34.1	46.7	70.5	32.4	43.7	64.8
BCELoss	10.5	26.6	57.4	16.4	31.4	64.2	8.0	18.0	49.8
Con-Margin	33.3	49.6	68.7	44.4	57.7	80.7	42.5	54.4	74.0
No Path	48.3	62.1	76.5	44.4	58.2	78.9	43.6	55.3	74.8
TEMP-BERT	**49.0**	**62.0**	**75.9**	**54.4**	**64.6**	**85.3**	**45.2**	**57.1**	**78.3**

Table 3: Results of ablation experiments on the three datasets (in %).

tensor network. It regulates concept embedding via the channel-wise gating mechanism to boost performance.

Implementation Details. The baseline method experimented by us, TMN, is obtained from the code published by the original authors[3]. Because the implementation of TMN needs validation data to set the training epochs, we use 10% terms for validating and 10% for testing. For each benchmark, we try various learning rates and report the best performance. To reduce the randomness, we evaluated TEMP five times on five differently divided test sets and training sets for each dataset and report the average performance. The hyperparameter k in Equation 2 is set to 0.2 on the three datasets. In the experiments of TEMP, all the pretrained contextual encoders are of *base* size with 12 layers [4]. We fine-tune the model with a batch size of 64 (which means 32 pairs of positive and negative samples). The optimizer is Adam with learning rate 2e-5, $\beta_1 = 0.9$, $\beta_2 = 0.999$ which is recommended by the authors of BERT.

The definitions of concepts used in training and testing are automatically gathered from the corresponding Wikipedia pages. We use the first line on the page as the word's definition. For each multi-word concept without a corresponding Wikipedia page, the definitions of the words that make the concept up are concatenated as its definition.

[3]https://github.com/JieyuZ2/TMN

[4]We used https://huggingface.co/transformers

4.2 Experimental Results

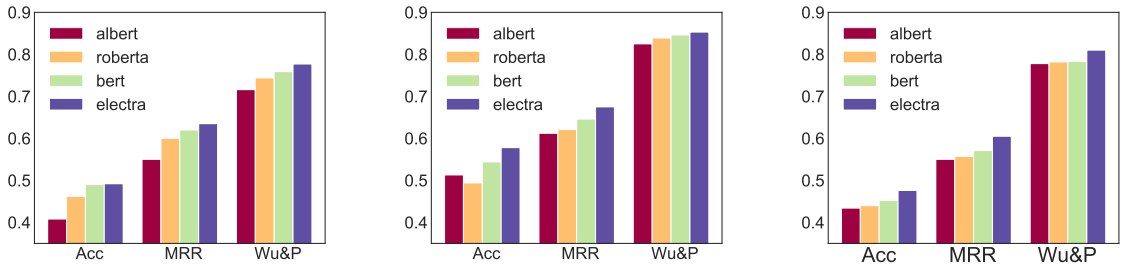
Table 2 reports the performance of TEMP based on the most representative contextual encoder, BERT and the contextual encoder that achieves the best performance, ELECTRA, and the baseline methods on the three benchmarks.

We summarize the evaluation results of the expansion task on the datasets in Table 2. As shown, TEMP-ELECTRA achieves the best performance on the three datasets and improves the state-of-the-art TMN model by 14.3%, 15.8% and 16.1% for Acc, MRR, Wu&P on average.

4.3 Ablation Studies

We perform ablation studies to analyze the effectiveness of the key choices in TEMP: (1) optimizing the margin loss by semantic similarity dynamic margin function; (2) using word definitions for taxonomy expansion; and (3) predicting the attachment by encoding taxonomy-paths. Since BERT is currently the most representative contextual encoder, all the experiments in ablation studies are based on BERT. We design the following experiments and report the results in Table 3.

The Effect of Dynamic Margin Function. We restrict TEMP to use a constant margin (Con-Margin). We experiment with different margin values and report the best performance. In the ex-



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

Figure 4: Results for different contextual encoders over three datasets.

perimental results, the dynamic margin function doesn't greatly improve the performance in the food dataset as it does in the other datasets. For this result, there are two possible reasons: (1) Semantic similarity is more important on a small dataset. In other words, with large training data, the model can learn the discriminative features with a constant margin. (2) The function can't improve a lot on flat datasets. The food dataset has the same depth as the science dataset but its number of nodes is more than three times the number of nodes in the science dataset, which means that the food dataset is very flat.

The Effect of Margin Loss. We modify TEMP to minimize Binary Cross-Entropy Loss (BCELoss). We find that the usage of margin loss is the main reason for the performance of TEMP.

$$BCELoss = -\frac{1}{n}\sum_{i=1}^{n}[y_i log(\hat{y}_i) + (1-y_i) log(1-\hat{y}_i)] \quad (3)$$

The Effect of Definition. We remove the definition from the input of TEMP (No Definition). From the results, one can see that the definitions improve the performance a lot on science and food datasets but not on the environment dataset. The poor quality definitions of the environment dataset may lead to this result. There are more than half of the words that are multi-words without a Wikipedia page in the dataset. Besides, the performance of TEMP without word definitions is also closed to the performance of prior state-of-the-art methods. It proves that BERT captures the hypernym-hyponym relations between terms to a relatively good degree.

The Effect of Encoding Paths. We modify the input of TEMP from taxonomy paths to the relation pairs (No Path). The experiments shows that the effect of encoding the taxonomy-paths is more significant on the deeper taxonomies.

4.4 Discussions

In this subsection, we discuss the following three factors that affect the effect of the model: (1) pre-trained encoders (2) parameter k (3) the number of sibling nodes of test terms.

Effect of Pre-trained Encoder. Figure 4 shows the preformance of TEMP on three datasets based on different pre-trained contextual encoders with the same experiment setup included ALBERT (albert-base-v2; Lan et al. (2019)), RoBERTa (roberta-base; Liu et al. (2019)), BERT (bert-base-uncased), ELECTRA(electra-base-discriminator). The performance of different encoders on different domain datasets shows consistency, and ELECTRA achieves the best performance on all datasets among the experimented contextual encoders. Another observation is that RoBERTa doesn't achieve better performance than BERT like it did on other tasks. The possible reason for it is that the text encoding algorithm used by RoBERTa, Byte-Pair Encoding is weaker in its ability to capture the substring information than WordPiece, the algorithm used by the other three encoders.

Effect of k. k is the parameter in dynamic margin function (equation 2). Figure 5 shows the effect of k on the Science dataset with BERT as the context encoder. As observed, when $0.1 \leq k \leq 1$, there is little difference in performance among various k. The obtained performance for different k also indicates that TEMP is not sensitive to the parameter k and has the advantage of robustness. We also try to use some larger k, experiments show that when $k > 10$, the loss doesn't converge.

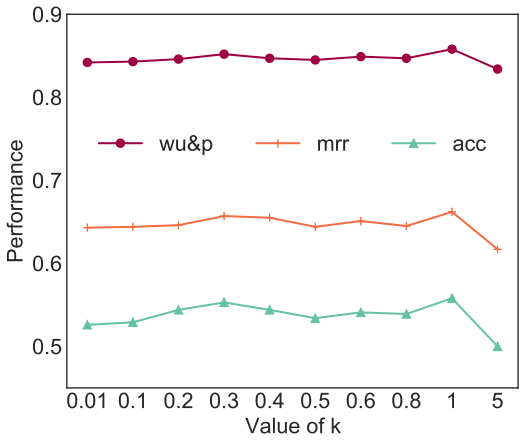


Figure 5: The performance of TEMP-BERT on the science dataset when varying k.

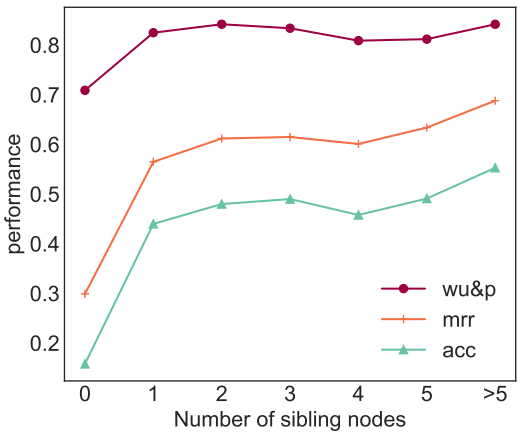


Figure 6: The performance of TEMP-BERT on the science dataset when varying the number of sibling nodes of test terms.

Effect of Sibling Nodes. To evaluate the effect of sibling nodes of test nodes in the self-supervised training data, we do the experiment in which the parent node of each test node retains a constant number of child nodes in the training taxonomy. In the '> 5' experiment, all the parents of test nodes have more than 5 child nodes in training data. Figure 6 shows the experimental results on the science dataset with BERT as the contextual encoder. From the experimental results, we get the following observations and conclusions: (1) As the number of sibling nodes in the training data increases, the performance of TEMP generally increases, which means that the sibling nodes in the test data make the model better learn the hypernym-hyponym relations. (2) When there is no sibling node in the training data, the performance in Acc and MRR is very low. However, compared with the other results with similar performance in Acc and MRR, it gets a higher score in Wu&P. This means that in this case, TEMP doesn't rank the ground-truth high, but the highest-ranked term is similar to the ground-truth in the taxonomy, such as the parent node of the ground-truth.

5 Conclusion

We proposed TEMP, a self-supervised method for taxonomy expansion, that relies on the pre-trained contextual encoder as its core. TEMP takes the definition of the query concept and the generated taxonomy-path as input to predict the attachment position. The model is trained by a margin ranking loss with a novel dynamic margin function to better capture the semantic similarity between taxonomy-paths. Experiments on three datasets from different domains show that TEMP outperforms state-of-the-art methods. Further ablation studies show that our key choices in TEMP have an effect on the performance in varying degrees especially the use of margin loss.

For future work, we plan to design sampling methods for TEMP to improve its performance and robustness. We also want to do interpretability studies about the effect of margin loss in model training.

Acknowledgements

This research is supported by Chinese Scientific and Technical Innovation Project 2030 (No. 2018AAA0102100), National Natural Science Foundation of China (No. 62077031, 61772289, U1936206). We thank the reviewers for their constructive comments.

References

Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. 2019. Docbert: Bert for document classification. *arXiv preprint arXiv:1904.08398*.

Georgeta Bordea, Els Lefever, and Paul Buitelaar. 2016. Semeval-2016 task 13: Taxonomy extraction evaluation (texeval-2). In *Proceedings of the 10th international workshop on semantic evaluation (semeval-2016)*, pages 1081–1091.

Sarthak Dash, Md Faisal Mahbub Chowdhury, Alfio Gliozzo, Nandana Mihindukulasooriya, and Nicolas Rodolfo Fauceglia. 2020. Hypernym detection using strict partial order networks. In *Proceedings of*

the AAAI Conference on Artificial Intelligence, volume 34, pages 7626–7633.

Thomas Demeester, Tim Rocktäschel, and Sebastian Riedel. 2016. Lifted rule injection for relation embeddings. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1389–1399.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Yushu Feng, Huan Wang, Haoji Roland Hu, Lu Yu, Wei Wang, and Shiyan Wang. 2020. Triplet distillation for deep face recognition. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 808–812. IEEE.

Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209.

Amit Gupta, Rémi Lebret, Hamza Harkous, and Karl Aberer. 2017. Taxonomy induction using hypernym subsequences. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1329–1338.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Coling 1992 volume 2: The 15th international conference on computational linguistics*.

Wen Hua, Zhongyuan Wang, Haixun Wang, Kai Zheng, and Xiaofang Zhou. 2016. Understand short texts by harvesting and analyzing semantic knowledge. *IEEE transactions on Knowledge and data Engineering*, 29(3):499–512.

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)*, pages 1092–1102.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Matthew Le, Stephen Roller, Laetitia Papaxanthos, Douwe Kiela, and Maximilian Nickel. 2019. Inferring concept hierarchies from text corpora via hyperbolic embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3231–3241.

Jiaqing Liang, Yanghua Xiao, Yi Zhang, Seung-won Hwang, and Haixun Wang. 2017a. Graph-based wrong isa relation detection in a large-scale lexical taxonomy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Jiaqing Liang, Yi Zhang, Yanghua Xiao, Haixun Wang, Wei Wang, and Pinpin Zhu. 2017b. On the transitivity of hypernym-hyponym relations in data-driven lexical taxonomies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Ting-En Lin and Hua Xu. 2019. Deep unknown intent detection with margin loss. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5491–5496.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Emaad Manzoor, Rui Li, Dhananjay Shrouty, and Jure Leskovec. 2020. Expanding taxonomies with implicit edge semantics. In *Proceedings of The Web Conference 2020*, pages 2044–2054.

Yuning Mao, Tong Zhao, Andrey Kan, Chenwei Zhang, Xin Luna Dong, Christos Faloutsos, and Jiawei Han. 2020. Octet: Online catalog taxonomy enrichment with self-supervision. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2247–2257.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*, pages 3111–3119.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. Patty: A taxonomy of relational patterns with semantic types. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1135–1145.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Alexander Panchenko, Stefano Faralli, Eugen Ruppert, Steffen Remus, Hubert Naets, Cédrick Fairon, Simone Paolo Ponzetto, and Chris Biemann. 2016. TAXI at SemEval-2016 task 13: a taxonomy induction method based on lexico-syntactic patterns, substrings and focused crawling. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 1320–1327, San Diego, California. Association for Computational Linguistics.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152. IEEE.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Chao Shang, Sarthak Dash, Md Faisal Mahbub Chowdhury, Nandana Mihindukulasooriya, and Alfio Gliozzo. 2020. Taxonomy construction of unseen domains via graph-based cross-domain knowledge transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2198–2208.

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*, pages 486–497.

Jiaming Shen, Zeqiu Wu, Dongming Lei, Chao Zhang, Xiang Ren, Michelle T Vanni, Brian M Sadler, and Jiawei Han. 2018. Hiexpan: Task-guided taxonomy construction by hierarchical tree expansion. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2180–2189.

Vered Shwartz and Ido Dagan. 2016. CogALex-V shared task: LexNET - integrated path-based and distributional method for the identification of semantic relations. In *Proceedings of the 5th Workshop on Cognitive Aspects of the Lexicon (CogALex - V)*, pages 80–85, Osaka, Japan. The COLING 2016 Organizing Committee.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398.

Chengyu Wang and Xiaofeng He. 2020. Birre: learning bidirectional residual relation embeddings for supervised hypernymy detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3630–3640.

Zhibiao Wu and Martha Palmer. 1994. Verb semantics and lexical selection. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 133–138.

Shuo Yang, Lei Zou, Zhongyuan Wang, Jun Yan, and Ji-Rong Wen. 2017. Efficiently answering technical questions—a knowledge graph approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with bertserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77.

Wenpeng Yin and Dan Roth. 2018. Term definitions help hypernymy detection. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 203–213.

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*, pages 1026–1035.

Zheng Yu, Haixun Wang, Xuemin Lin, and Min Wang. 2015. Learning term embeddings for hypernymy identification. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Jieyu Zhang, Xiangchen Song, Ying Zeng, Jiaze Chen, Jiaming Shen, Yuning Mao, and Lei Li. 2021. Taxonomy completion via triplet matching network. In *AAAI*.