Ontology-enhanced Prompt-tuning for Few-shot Learning

Hongbin Ye, Ningyu Zhang
Zhejiang University
AZFT Joint Lab for Knowledge Engine
Hangzhou Innovation Center
Hangzhou, China
tyehongbin,zhangningyu@zju.edu.cn

Shumin Deng
Zhejiang University
AZFT Joint Lab for Knowledge Engine
Hangzhou Innovation Center
Hangzhou, China
231sm@zju.edu.cn

Xiang Chen
Zhejiang University
AZFT Joint Lab for Knowledge Engine
Hangzhou Innovation Center
Hangzhou, China
xiang_chen@zju.edu.cn

Hui Chen, Feiyu Xiong
Alibaba Group
Hangzhou, China
{weidu.ch,feiyu.xfy}@alibaba-inc.com

Xi Chen
Tencent
Shenzhen, China
jasonxchen@tencent.com

Huajun Chen
Zhejiang University & AZFT Joint Lab for Knowledge Engine & Hangzhou Innovation Center
Hangzhou, China
huajunsir@zju.edu.cn

ABSTRACT

Few-shot Learning (FSL) is aimed to make predictions based on a limited number of samples. Structured data such as knowledge graphs and ontology libraries have been leveraged to benefit the few-shot setting in various tasks. However, the priors adopted by the existing methods suffer from challenging knowledge missing, knowledge noise, and knowledge heterogeneity, which hinder the performance for few-shot learning. In this study, we explore knowledge injection for FSL with pre-trained language models and propose ontology-enhanced prompt-tuning (OntoPrompt). Specifically, we develop the ontology transformation based on the external knowledge graph to address the knowledge missing issue, which fulfills and converts structure knowledge to text. We further introduce span-sensitive knowledge injection via a visible matrix to select informative knowledge to handle the knowledge noise issue. To bridge the gap between knowledge and text, we propose a collective training algorithm to optimize representations jointly. We evaluate our proposed OntoPrompt in three tasks, including relation extraction, event extraction, and knowledge graph completion, with eight datasets. Experimental results demonstrate that our approach can obtain better few-shot performance than baselines.

CCS CONCEPTS

• Information systems → Information extraction.

KEYWORDS

Few-shot Learning, Ontology, Prompt-tuning, Relation Extraction, Event Extraction, Knowledge Graph Completion

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.

ACM Reference Format:
Hongbin Ye, Ningyu Zhang, Shumin Deng, Xiang Chen, Hui Chen, Feiyu Xiong, Xi Chen, and Huajun Chen. 2022. Ontology-enhanced Prompt-tuning for Few-shot Learning. In Proceedings of the ACM Web Conference 2022 (WWW ’22), April 25–29, 2022, Virtual Event, Lyon, France. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3485447.3511921

1 INTRODUCTION

Recent years have witnessed the success of deep neural networks, however, open issues remain as they are still prone to labeled data in practice and face unignorable challenges owing to the variations of domains, language, and tasks. These drawbacks lead to the research of an important technique, few-shot learning (FSL), which can significantly improve the learning capabilities of machine intelligence and practical adaptive applications by accessing only a small number of labeled examples. Over the past few years, FSL has been introduced in a wide range of machine learning tasks, such as relation extraction [6, 32, 62, 67, 68], event extraction [10, 47] and knowledge graph completion [77].

To address the few-shot issue, on the one hand, researchers apply the meta-learning strategy to endow the new model the ability to optimize rapidly with the existing training knowledge or leverage transfer learning to alleviate the challenge of data-hungry [4, 66]. Benefiting from the self-supervised pre-training on the large corpus, the pre-train—fine-tune paradigm has become the de facto standard for natural language processing (NLP). It can perform well on downstream tasks with a small amount of task-specific data. On the other hand, structured data such as knowledge systems and ontology libraries can be modeled under the few-shot setting. Note that those prior knowledge are mostly abstract summaries of human experience and can provide vital support for FSL in various domains [12, 23, 70, 75]. In this paper, we focus on injecting knowledge for few-shot learning, and there are still several nontrivial challenges as follows:

• Knowledge Missing. Knowledge injection may not be able to retrieve task-relevant facts due to the incompleteness of the external knowledge base and thus provides no useful or even irrelevant information to downstream tasks. How to enrich task-relevant knowledge for tasks is an important issue.
• **Knowledge Noise.** Previous studies [34, 44, 65] have demonstrated that not all knowledge is beneficial for downstream tasks, and an indiscriminate injection of knowledge may lead to negative knowledge infusion, which is detrimental to the performance of downstream tasks. Thus, context-sensitive and task-specific knowledge selection is critical for knowledge-enhanced learning.

• **Knowledge Heterogeneity.** The language corpus of downstream tasks is quite different from the injected knowledge leading to two individual vector representations [78]. How to design a special joint training objective to fuse knowledge information is another challenge.

In this study, to address the above-mentioned issues, we explore knowledge injection for FSL with pre-trained language models (PLMs) and propose ontology-enhanced prompt-tuning (OntoPrompt). Note that pre-trained language models can extract plenty of statistical information from large amounts of data. And external knowledge, such as knowledge graphs, is the outcome of human wisdom, which can be good prior to the modeling of statistics.

**Firstly,** we propose **ontology transformation** to enrich and convert structure knowledge to text format. Specifically, we utilize pre-defined templates to convert knowledge to text as prompts. Prompt-tuning can reduce the gap between the pre-training model tasks and downstream tasks. For example, given a sentence s, “Turing entered King’s College, Cambridge in 1931, and then went to Princeton University to study for a doctorate,” we can pack them into a knowledgeable prompt based on ontology: “s. Turing [MASK] King’s College, Cambridge:” PLMs should predict label tokens at the masked position to determine the label of the input. Note that **ontology as prompt** places knowledge of entities/spans into the input text, which is model-agnostic and can be plugged into any type of pre-trained language model.

**Second,** we propose **span-sensitive knowledge injection** to select informative knowledge and mitigate injecting noise. Since not all external knowledge is advantageous for the downstream task, and some irrelevant and noisy knowledge may lead to the changes in the meaning of the original sentence, we leverage a visible matrix based on spans and their corresponding external knowledge to build the knowledge injection. In this way, not all tokens in the input sentences will be influenced by external knowledge.

**Third,** we propose a **collective training** algorithm to jointly optimize representations. Note that the injected external knowledge should be associated with the surrounding context; we add some learnable tokens with random initialization and optimize those tokens as well as injected ontology tokens with language model fixed. Inspired by the previous study [19] that prompt-tuning in the low-data regime is unstable and may obtain poor performance, we further optimize all parameters to collective train the ontology text and input text representations.

Finally, we evaluate our OntoPrompt with three tasks: relation extraction, event extraction, and knowledge graph completion. We leverage the dataset-related ontologies as external knowledge and conduct experiments on eight datasets in the few-shot setting. Experimental results illustrate that our proposed approach can obtain better performance. It should be noted that our approach is model-agnostic, and therefore orthogonal to existing pre-trained language models.

## 2 RELATED WORK

### 2.1 Knowledge-enhanced Learning

Pre-training fine-tuning has become a default paradigm for natural language processing. However, the performance of the knowledge-driven downstream task (for example, question answering or relation extraction) is dependent on structured relational knowledge; thus, the direct fine-tuning of pre-trained LMs yield suboptimal results. Thus, external knowledge graphs have been considered as an indispensable part of language understanding [73], which has inspired knowledge-aware models such as ERNIE [79]. To integrate the ontology knowledge, [24] propose to tackle the zero-shot event detection problem by mapping each event mentioned to a specific type in a target event ontology. [10] propose an event detection framework based on ontology embedding with event correlations, which interoperates symbolic rules with popular deep neural networks. [18] propose a novel ZSL framework called OntoZSL which not only enhances the class semantics with an ontological schema but also employs an ontology-based generative model to synthesize training samples for unseen classes. [53] propose an ontology-guided entity alignment method named OntoEA, where both knowledge graphs and their ontologies are jointly embedded, and the class hierarchy and the class disjointness are utilized to avoid false mappings.

However, the dilemma of knowledge missing, knowledge noise, and knowledge heterogeneity have not been addressed. Concretely, [34] propose K-BERT, which utilizes soft-position and visible matrix to limit the impact of knowledge. [44] propose CokeBERT, which can dynamically select and embed knowledge context according to textual context for PLMs to avoid the effect of redundant and ambiguous knowledge in knowledge graphs that cannot match the input text. [2] propose knowledge-to-text transformation to benchmark commonsense question answering.

Different from their approaches, we integrate ontology knowledge into pre-trained language model fine-tuning with prompts. We propose a novel ontology transformation to enrich the missing knowledge and utilize span-sensitive knowledge injection to mitigate the noisy knowledge. We further optimize those heterogeneous representations with collective training.

### 2.2 Few-shot Learning

Few-shot learning aims to improve the learning capabilities for machine intelligence, and practical adaptive applications with only a small number of training instances [66]. Our proposed approach corresponds to the other few-shot methods, including: (1) Intermediate training [39, 60], which supplements the pre-trained LMs with further training on the data-rich supervised tasks. (2) Metalearning [9, 11, 62], in which the quantities of the auxiliary tasks are optimized. (3) Semi-supervised learning [54], which leverages unlabeled samples.

#### 2.2.1 Relation Extraction

Relation extraction aims to identify the relation between entity pairs based on a given contextual text
Figure 1: Ontology knowledge injection for relation extraction (RE), event extraction (EE) and knowledge graph completion (KGC) (Best viewed in color.).

[28, 64]. In order to reduce the cost of labeling, previous studies utilize distant supervision based on the knowledge graph to generate labeled examples automatically [69, 74]. More recent few-shot relation extraction approaches leverage prototype network [8], multi-level matching and aggregation, relational twin network and meta-learning [15].

2.2.2 Event Extraction. Event extraction aims to identify event triggers with arguments from text, which is generally formulated as a classification or structured prediction problem [37]. To address the few-shot issue for even extraction, [33] utilize external knowledge from frame net and propose a global inference approach to improve the event detection performance. More recently, [31, 51, 58] reformulate the event extraction task as machine reading comprehension (MRC), which can achieve better performance in few-shot setting than vanilla fine-tuning. Although MRC has the same wide range of application scenarios as our method, we automatically construct templates through ontology knowledge.

2.2.3 Knowledge Graph Completion. Knowledge graph completion can be achieved by link prediction of the knowledge graph [72, 76]. Previous researchers have proposed various link prediction methods which encode entities and relations into the low-dimensional continuous space, such as TransE [3], TransR [30], TransD [26], TransH [52], etc. DistMult [56] analyze the matrix and simplify the RESCAL approach, while ComplEx [48] extend DistMult to the complex domain. RotatE [45] define each relation as the transformation and rotation from the source entity to the target entity in the complex space. KG-BERT [57] define the description of the entities and relations in the triple as input and leverage BERT to calculate the score of the triple. For few-shot knowledge graph completion, lots of studies [42, 63, 66] have been proposed. [66] propose a general framework called Weighted Relation Adversarial Network (RAN), which utilizes an adversarial procedure to help adapt knowledge/features learned from high resource relations to different but related low resource relations. [77] propose generalized relation learning (GRL), which utilizes semantic correlations between relations to serve as a bridge to connect semantically similar relations.

2.3 Prompt-tuning

A new fine-tuning methodology named prompt-tuning has arisen: adapting the pre-trained language model directly as a predictor through completion of a cloze task. Prompt-tuning for pre-trained language models is a rapidly emerging field in natural language processing [40, 46, 71] and have attracted lots of attention. Originally from GPT-3, prompt-tuning has been applied to various of tasks including relation extraction [20], event extraction [21, 59], named entity recognition [5, 7], entity typing [13], and so on. To facilitate the labor-intensive prompt engineering, [43] propose AUTOPROMPT, which can search prompts based on a gradient method to select label words and templates. More recent works including P-tuning [36], Prefix-tuning [29] also propose to leverage continuous templates, which is more effective than discrete prompt search.

Recently, some studies have tried to integrate external knowledge into prompt designing. [20] propose an approach called PTR, which leverages logic rules to construct prompts with sub-prompts for many-class text classification. [22] propose an approach to incorporate external knowledge graph into the verbalizer with calibration. [6] propose a knowledge-aware prompt-tuning approach that injects knowledge into prompt template design and answer construction. Different from those approaches, we regard prompts as a bridge between text and knowledge and focus on addressing the issues of knowledge missing, knowledge noise, and knowledge heterogeneity.

3 METHODOLOGY

3.1 Preliminaries

Our approach OntoPrompt is a general framework that can be applied to widespread applications as shown in Figure 2. We evaluate our approach in three popular tasks, namely, relation extraction (RE), event extraction (EE), and knowledge graph completion (KGC). To inject ontology knowledge, we introduce ontology transformation to enrich and convert structure knowledge into raw texts (ontology text). We regard those raw texts as auxiliary prompts and append them to the input sequences and prompt templates. Note that those ontology texts, including textual descriptions, can provide semantic information about the ontology of the mentioned
The divorce settlement called for Giuliani to pay Hanover more than $6.8 million.

Event trigger: Event trigger (trigger) is a word that can trigger the occurrence of an event, and it is the most …

Event trigger is [MASK], event type is [MASK], argument are [MASK].

Knowledge Graph Completion

When the prompt is fed into the MLM, the model can obtain the probability distribution $p([\text{MASK}] | X_{\text{prompt}})$ of the candidate class, $y \in Y$ as:

$$p(y | X_{\text{prompt}}) = \sum_{w \in Y} p([\text{MASK}] = w | X_{\text{prompt}})$$

3.2 General Framework with Prompt-Tuning

Let $X_{\text{in}} = \{x_1, x_2, \ldots, x_L\}$ be a sentence, where $x_i$ is the $i^{th}$ token in the input sentence and $L$ is the number of tokens. Specifically, $X_{\text{in}}$ is converted to a fixed token sequence $X_{\text{in}}$ and then mapped to a sequence of hidden vectors $\{h_k \in \mathbb{R}^d\}$. Given the input sequence, $X_{\text{in}} = [\text{CLS}]X_{\text{in}}[\text{SEP}]$, the vanilla fine-tuning approaches leverage a generic head layer over [CLS] embeddings (e.g., an MLP layer) to predict an output class. In this paper, we follow previous prompt-tuning studies [17] and use a task-specific pattern string ([Template] $\mathcal{T}$) to coax the model into producing a textual output corresponding to a given class (label token $M(Y)$). Specifically, $X_{\text{prompt}}$, containing one [MASK] token is directly tasked with the MLM input as:

$$X_{\text{prompt}} = [\text{CLS}]X_{\text{in}}[\text{SEP}]\mathcal{T}[\text{SEP}]$$

3.3 Ontology Transformation

In this paper, the ontology representation is denoted as $O = \{C, E, D\}$, $C$ is a set of concepts, $E$ is the connected edge between the ontologies, and $D$ is the textual description of each ontology (The ontological schema contains a paragraph of textual descriptions, and they are lexically meaningful information of concepts, which can also be represented by triples using properties, e.g., rdfs:comment). The concept set here refers to domain-specific concepts. For example,
we leverage the type ontology related to the mentioned entity in relation extraction and event extraction. We utilize the domain (i.e., head entity types) and range (i.e., tail entity types) constraints in knowledge graph completion.

Regarding the differences in downstream tasks, we leverage different sources of ontology for each task for ontology transformation. We first extract the ontology of each instance from external knowledge graphs and then transform those ontologies (rdfs:comment) into raw texts as auxiliary prompts.

3.3.1 Applying to Relation Extraction. We utilize the MUC (Message Understanding Conference) [49] which defines the concept of named entities. Note that named entities can provide important type information, which is beneficial for relation extraction. We utilize those definitions as textual descriptions in ontology schema. Specifically, we use [CLS] < InputText > [SEP] < Template > [SEP] < OntologyText > [SEP] as the default input sequence. We construct placeholders for entities in the < OntologyText > and replace those placeholders with external textual descriptions. Moreover, we leverage the path between entity pairs from the ontology as meta-relation text to enhance the < OntologyText > as shown in Figure 2.

We further add learnable tokens as virtual tokens on both sides of the [MASK] so that the model can automatically learn the most suitable words for the prompt. Specifically, we use several unused tokens\(^1\) [u1]-[u4] (e.g., unused or special token in the vocabulary) for virtual tokens.

3.3.2 Applying to Event Extraction. We follow the work of [25] to build the ontology of event extraction. We first leverage the ACE event schema\(^2\) as our base event ontology. The ACE2005 corpus includes the rich event annotations currently available for 33 types. However, in real-world scenarios, there may be thousands of types of events. In order to facilitate inference in scenarios with limited training samples, we integrate the 33 ACE event types and argument roles with 1,161 frames from FrameNet and construct a bigger event ontology for event extraction. We manually map the selected frames to event ontology. Similarly, we use [CLS] < InputText > [SEP] < Template > [SEP] < OntologyText > [SEP] as the default format for the input sequence. We construct placeholders for triggers and type in the < OntologyText >.

Similar to relation extraction, we also leverage the learnable virtual token to enhance the prompt representation.

3.3.3 Applying to Knowledge Graph Completion. We use the corresponding items obtained from the external Wikidata query as the source of ontology and extract the textual description\(^3\). Following KG-BERT [57], we regard knowledge graph completion as a triple classification task and concatenate entities and relations as an input sequence. Similar to the above-mentioned tasks, we use [CLS] < InputText > [SEP] < Template > [SEP] < OntologyText > [SEP] as the default input sequence.

We also use the learnable virtual token to enhance the prompt representation. In inference, we rank the output scores according to the maximum probability of sentences predicted by the language model and count the hits of gold standard entities to evaluate the performance of the approach.

3.4 Span-sensitive Knowledge Injection

As observed by [34], excessive knowledge injection may introduce some noise into the original text and cause a performance decay. In order to solve this problem, we propose span-sensitive knowledge injection as shown in Figure 3. Given input text \(X_{\text{in}} = \{x_1, x_2, ..., x_L\}\) with \(L\) tokens, we use a visibility matrix to limit the impact of the knowledge injection on the input text. In the language model architecture, the attention mask matrix with self-attention weight is added before the softmax layer. Therefore, we modify the attention mask matrix \(M\) as follows:

\[
M_{ij} = \begin{cases} 
0 & x_i, x_j \in x \\
0 & x_i, x_j \in x^o \\
0 & x_i, x_j \in x^o \text{ and } i = p_k \\
-\infty & \text{otherwise}
\end{cases}
\]

\(^1\)The number of tokens is a hyperparameter.

\(^2\)https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/englishevents-guidelines-v5.4.3.pdf

\(^3\)https://www.wikidata.org/
where $x_i$ and $x_j$ are tokens from $x_{input}$ and ontology text, separately. $-\infty$ in attention mask matrix $M_{ij}$ blocks token $i$ from attending to token $j$, and 0 allows token $i$ to attend to token $j$. $x_i$ can attend to $x_j$ if both tokens belong to the $x_{input}$, or both tokens belong to the text description of the same ontology $x_o$, or $x_i$ is the token at the span position of entity $e_i$ in $x_{input}$ and $x_j$ is from its ontology description text $x^o$. $p_2i$ indicates the position of the mentioned span (e.g., entities in relation extraction and knowledge graph completion, triggers, or arguments in event extraction) in the input text.

3.5 Collective Training

Note that those injected tokens from ontology transformation should be associated with the input sequence. Thus, we introduce collective training to optimize ontology tokens and input sequence. First, we initialize the ontology tokens with real word embeddings and optimize those ontology tokens with the language model fixed. Then, we optimize all the parameters of the model, including the language model and ontology tokens. Note that our model does not rely on the specific model architecture; thus, it can be plugged into any kind of pre-trained language model like BERT, BART.

4 EXPERIMENTS

In this section, we introduce the extensive experiments of OntoPrompt in three tasks: relation extraction, event extraction, and knowledge graph completion. The experimental results show that OntoPrompt can perform better than baselines in both the standard supervised and few-shot settings.

4.1 Datasets

For relation extraction, we choose a variety of datasets to evaluate OntoPrompt, including sentence-level extraction datasets such as TACRED-Revisit [1], SemEval-2010 Task 8, Wiki80 and dialogue-level extraction dataset DialogueRE [61]. The detailed statistics of each relation extraction dataset are shown in Table 1.

For event extraction, we evaluate our OntoPrompt model with the ACE 2005 dataset [14], which defines 33 different event types and 35 semantic roles. We use the same data split and pre-processing step following [50] in which 529/30/40 newswire documents are used for training/dev/test set.

For knowledge graph completion, we use several standard knowledge graph completion datasets, including UMLS which has various categories, WN18RR based on WordNet, and FB15K-237 based on Freebase. FB15K is a subset of the large-scale knowledge graph Freebase. In FB15K-237, the triples that can be reversed are deleted because these triples are more challenging to distinguish. Compared with FB15K, WN18RR has more entities and fewer types of relations. Since the UMLS dataset is a vertical domain dataset, there are fewer entities than the previous two knowledge graphs. Detailed statistics are shown in Table 2.

### Table 1: Relation Extraction Dataset statistics.

| Dataset     | #Relations | #Triples | #No-relation |
|-------------|------------|----------|--------------|
| SemEval-2010 Task 8 | 19 | 10,717 | 17.4% |
| TACRED-Revisit | 42 | 106,264 | 79.5% |
| Wiki80 | 80 | 56,000 | - |
| DialogueRE | 36 | 1,788 | - |

### Table 2: Knowledge Graph Completion Dataset statistics.

| Dataset     | #Relations | #Entities | #Train | #Dev | #Test |
|-------------|------------|-----------|--------|------|-------|
| FB15k-237  | 237 | 14,541 | 1,896 | 17,535 | 2,000 |
| WN18RR  | 18 | 40,943 | 88 | 3,034 | 3,134 |
| UMLS  | 46 | 135 | 329 | 652 | 661 |

For knowledge graph completion, we use several standard knowledge graph completion datasets, including UMLS which has various categories, WN18RR based on WordNet, and FB15K-237 based on Freebase. FB15K is a subset of the large-scale knowledge graph Freebase. In FB15K-237, the triples that can be reversed are deleted because these triples are more challenging to distinguish. Compared with FB15K, WN18RR has more entities and fewer types of relations. Since the UMLS dataset is a vertical domain dataset, there are fewer entities than the previous two knowledge graphs. Detailed statistics are shown in Table 2.

### Table 3: Results on relation extraction dataset for OntoPrompt (F1 score). We report the results of fine-tuning BERT_LARGE with entity markers. We use $K = 8, 16, 32$ (# examples per class). Full represents the full training set.

| Dataset     | Model   | K = 8      | K = 16     | K = 32     | Full     |
|-------------|---------|------------|------------|------------|----------|
|            |         | 24.8       | 43.5       | 63.9       | 87.4     |
|            | GDPNET  | 25.3       | 43.5       | 65.2       | 88.0     |
|            | OntoPrompt | 52.6 (±7.0) | 65.0 (±9.0) | 83.0 (±7.0) | 89.1 (±1.0) |
| TACRED-Revisit | Fine-tuning | 7.4       | 15.3       | 25.4       | 74.8     |
|            | GDPNET  | 7.9       | 17.8       | 26.1       | 77.2     |
|            | OntoPrompt | 28.8 (±0.0) | 33.1 (±1.0) | 34.8 (±3.7) | 78.2 (±4.0) |
| Wiki80     | Fine-tuning | 46.1       | 60.5       | 70.1       | 85.1     |
|            | GDPNET  | 47.4       | 62.3       | 70.5       | 87.0     |
|            | OntoPrompt | 68.7 (±2.1) | 75.6 (±3.0) | 79.1 (±4.0) | 87.9 (±9.0) |
| DialogueRE | Fine-tuning | 29.3       | 41.1       | 49.5       | 58.3     |
|            | GDPNET  | 30.1       | 42.5       | 49.7       | 65.2     |
|            | OntoPrompt | 41.5 (±2.2) | 47.3 (±4.0) | 52.4 (±4.7) | 66.1 (±9.0) |
Table 4: Knowledge graph completion result (8-shot) on WN18RR, FB15K-237 and UMLS. The best score is in bold.

| Model      | WN18RR  | FB15K-237 (mini) | UMLS       |
|------------|---------|------------------|------------|
|            | MR ↓    | Hit@10 ↑         | MR ↓       | Hit@10 ↑   |
| TransE [3] | 19313.0 | 0.0004           | 5847.4     | 0.0925     |
| TransR [30] | 20861.0 | 0.0023           | 5970.5     | 0.0903     |
| TransH [52] | 19272.9 | 0.0015           | 6102.3     | 0.0802     |
| DistMult [56] | 20671.6 | 0.0003           | 5791.7     | 0.0059     |
| Complex [48] | 20318.6 | 0.0000           | 6451.8     | 0.0046     |
| RotatE [45] | 20162.1 | 0.0003           | 7365.8     | 0.0066     |
| KG-BERT [57] | 2937.2  | 0.1175           | 2023.4     | 0.0451     |
| RAN [66]   | 3150.2  | 0.0769           | 3320.5     | 0.0072     |
| GRL [77]   | 2913.3  | 0.0900           | 2913.5     | 0.0300     |

OntoPrompt | 1442.6 (−1470.7) | 0.1344 (+0.0169) | 714.5 (−1308.9) | 0.111 (+0.0185) | 27.2 (−7.0) | 0.3448 (+0.0066)

Table 5: F1 score (%) on few-shot learning. The reported F1 score refers to the result of the event argument classification (event extraction) in ACE2005.

| Model      | 1%     | 5%     | 10%    | 20%    | Full   |
|------------|--------|--------|--------|--------|--------|
| dbRNN [41] | 8.1    | 17.2   | 24.1   | 58.7   |        |
| JMEE [35]  | 8.9    | 20.3   | 28.4   | 60.3   |        |
| DYGIE++ [50] | 5.3    | 15.7   | 23.4   | 51.4   |        |
| MQAEE [51] | 5.2    | 27.3   | 32.1   | 38.1   | 53.4   |
| TEXT2EVENT [38] | 3.4    | 19.8   | 25.3   | 36.9   | 49.8   |

OntoPrompt 25.6 (+20.4) 40.1 (+12.8) 47.5 (+15.7) 50.1 (+12.0) 55.3 (+5.0)

Figure 4: Ablation study results of OntoPrompt.

For event extraction, from Table 5, we report the F1 score results of event extraction in extremely low-resource scenarios (training with less than 20% data, with the similar setting to [31]). Notably, OntoPrompt yields advantages in few-shot event extraction. To be specific, OntoPrompt can obtain 25.6% F1 with 1% data, in comparison to 5.2% in MQAEE and 3.4% in TEXT2EVENT. Although the performance of OntoPrompt on the full sample is slightly weaker than that of JMEE, which relies on external data augmentation, we believe that OntoPrompt can effectively identify triggers and arguments with less data dependence. Besides, we notice that OntoPrompt has a faster convergence speed (See details in appendix). Compared with MQAEE and TEXT2EVENT, which need a certain amount of data to fine-tune the language model, OntoPrompt only leverages 20% of the training data to achieve similar satisfactory performance.

For knowledge graph completion, from Table 4, we report the performance of multiple knowledge graph completion approaches in the few-shot setting (8 samples per relationship, 8-shot). The experimental results show that OntoPrompt can achieve the best performance, proving our proposed model’s superiority. Although OntoPrompt has a slight improvement in the few-shot setting on UMLS, it has an increase of 6.6% with the hits@10 on FB15K-237 compared to KG-BERT. We think the improvement of FB15K-237 is mainly due to OntoPrompt’s ability to fully utilize the implicit fact knowledge obtained from external ontology.

Although we evaluate OntoPrompt with the above three tasks, our proposed approach can also be applied to other tasks such as text classification, question answering with suitable ontologies.

4.4 Ablation Study

To further prove the effects of different modules in OntoPrompt, we conduct the ablation study and report the experimental results in Figure 4. w/o span-sensitive KI indicate the model without the visible matrix; thus, all tokens can see each other. w/o span-sensitive KI & collective training refers to the model without the visible matrix, and all parameters are optimized. We observe that all models without each component have a performance decay, which demonstrates the effectiveness of our approach.

4.5 Case Study

To further analyze the collective training of virtual tokens, we conduct the nearest neighbor word embedding search for the virtual token in the sentence to project the best optimized virtual token into a readable natural language. From Table 6, we can observe the following findings:

1) The OntoPrompt can customize a unique virtual token for each ontology in different event subtypes, while the virtual tokens of

We only report the performance of event argument classification due to page limit.

Due to the huge computation in ranking during the inference stage, we construct and evaluate the models on a small test set called FB15K-237 (mini), baselines are implemented by https://github.com/thunlp/OpenKE
Table 6: Case study results on the ACE2005 dataset. We map the virtual token into the readable natural language (red) in the input examples, demonstrating the intuitiveness of OntoPrompt. It is worth noting that OntoPrompt can customize a unique virtual token for each prompt based on ontology with different event subtypes. In contrast, the virtual tokens of models without knowledge injection are randomly scattered over the entire dataset.

| Model          | Input Example                                                                 | Event Type  | Model F1 |
|---------------|-------------------------------------------------------------------------------|-------------|----------|
| BERT          | [CLS] Everyone, everywhere should have the basic right to elect their government. [SEP] | Personnel.Elect | 49.1     |
| BERT + Ontology | [CLS] Everyone, everywhere should have the basic right to elect their government. [SEP] Event trigger: Event trigger (trigger) is a word that can trigger the occurrence of an event ... [SEP] | Personnel.Elect | 50.2     |
| OntoPrompt (-Onto) | [CLS] Everyone, everywhere should have the basic right to elect their government. [SEP] Event trigger is <Trigger>, event type is <Type>, argument are [prohibits] [handed] [MASK] [relics] [handed] ... [SEP] Person: named person or family ... [SEP] | Personnel.Elect | 52.3     |
| OntoPrompt (-VIR) | [CLS] Everyone, everywhere should have the basic right to elect their government. [SEP] Event trigger is [MASK], event type is [MASK], argument are [MASK] ... [SEP] Person: The Person is conceived of as independent of other specific individuals with whom they have relationships and ... [SEP] | Personnel.Elect | 52.9     |
| OntoPrompt (-Both) | [CLS] Everyone, everywhere should have the basic right to elect their government. [SEP] Event trigger is [MASK], event type is [MASK], argument are [MASK] ... [SEP] | Personnel.Elect | 47.5     |
| OntoPrompt      | [CLS] Everyone, everywhere should have the basic right to elect their government. [SEP] Event trigger is [MASK], event type is [MASK], argument are [discussed] [receptionist] [MASK] [moaned] [teachings] ... [SEP] Person: the person elected. The Person is conceived of as independent of other specific individuals with whom they have relationships and ... [SEP] | Person:Elect | 55.3     |
|                | [CLS] Here, prisoners were tortured. [SEP] Event trigger is [MASK], event type is [MASK], argument are [focuses] [ethiopia] [MASK] [brownists] [explores] ... [SEP] Place: where the injuring takes place. Place designates relatively stable bounded area of the world which have permanent relative ... [SEP] | Life.Injure  |          |
|                | [CLS] But it has nothing to do whether we should go to war with Iraq or not. [SEP] Event trigger is [MASK], event type is [MASK], argument are [angrily] [Iraqi] [MASK] [hijits] [reddish] ... [SEP] Attacker: The attacking/instigating agent. An assailant physically attacks a Victim ... [SEP] | Conflict.Attack |        |

model without ontology knowledge are random tokens scattered in the entire dataset. This indicates that injecting external knowledge can benefit prompt template representation.

2) The integration of ontology knowledge can improve the base pre-trained model to varying degrees, e.g., BERT (+1.1%) and OntoPrompt (+3.0%), which further prove that the ontology knowledge is helpful to the downstream tasks such as event extraction.

3) The removal of virtual template tokens reduces our OntoPrompt performance (-2.4%). We believe that the virtual token can learn the implicit information of the task-specific knowledge and adjust the appropriate prompt according to the context. At the same time, the removal of the virtual template token and the ontology knowledge module both will result in a more severe performance decay (-7.8%), which demonstrates that the virtual template token and the ontology knowledge information both have a positive impact on the model performance.

4.6 Discussion

Notably, our OntoPrompt can be viewed as an approach to leverage prompts as a bridge to inject knowledge. Previous studies such as RAG [27] has the same intuition to retrieve and concatenate relevant texts as knowledge. That knowledge can be directly revised and expanded, and accessed knowledge can be inspected and interpreted. Apart from those approaches, we argue that prompts transformed from ontology can consist of more density knowledge. Moreover, we utilize span-sensitive knowledge injection to filter noisy information. However, our model cannot handle those complex or structure knowledge such as OWL reasoning rules, description logic and represent them as raw texts. We will leave this for future works.

5 CONCLUSION AND FUTURE WORK

We focus on ontology knowledge injection and propose an ontology-enhanced prompt-tuning (OntoPrompt) approach, which can be applied to relation extraction, event extraction, and knowledge graph completion tasks. Experimental results illustrate that OntoPrompt can obtain better results than baselines in eight datasets. The method in this paper verifies the effectiveness of ontology knowledge as prompt guidance.

In the future, we plan to apply our approach to more applications such as text generation, question answering. We will also try to combine the proposed method with semi-supervised learning algorithms to make better use of large amounts of unlabeled data. In addition, we will try to inject the more complex knowledge such as symbolic rules in the knowledge graph into the proposed model to construct more powerful prompt templates.

ACKNOWLEDGMENTS

We want to express gratitude to the anonymous reviewers for their hard work and kind comments. This work is funded by National Key R&D Program of China (Funding No.SQ2018YFC000004), NSFC1846204/NSFCU19B2027, Zhejiang Provincial Natural Science Foundation of China (No. LGG22F030011), Ningbo Natural Science Foundation (2021J190), and Yongjiang Talent Introduction Programme.
