# Prompt Tuning for Few-shot Relation Extraction via Modeling Global and Local Graphs

**Zirui Zhang**[1], **Yiyu Yang**[1], **Benhui Chen**[1,2]
[1]Dali University, Dali, Yunnan, China
[2]Lijiang Normal College, Lijiang, Yunnan, China
zzr@stu.dali.edu.cn, yangyiyu@dali.edu.cn, bhchen@dali.edu.cn

**Abstract**

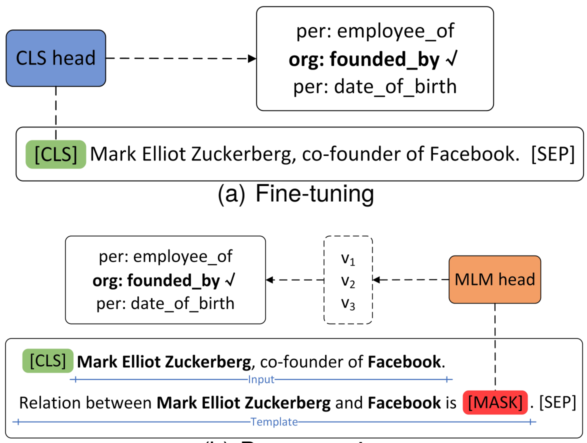
Recently, prompt-tuning has achieved very significant results for few-shot tasks. The core idea of prompt-tuning is to insert prompt templates into the input, thus converting the classification task into a masked language modeling problem. However, for few-shot relation extraction tasks, how to mine more information from limited resources becomes particularly important. In this paper, we first construct a global relation graph based on label consistency to optimize the feature representation of samples between different relations. Then the global relation graph is further divided to form a local relation subgraph for each relation type to optimize the feature representation of samples within the same relation. This fully uses the limited supervised information and improves the tuning efficiency. In addition, the existence of rich semantic knowledge in relation labels cannot be ignored. For this reason, this paper incorporates the knowledge in relation labels into prompt-tuning. Specifically, the potential knowledge implicit in relation labels is injected into constructing learnable prompt templates. In this paper, we conduct extensive experiments on four datasets under low-resource settings, showing that this method achieves significant results.

**Keywords:** Relation Extraction, Knowledge Graph, Few-shot, Prompt Tuning, Relation Graph

## 1. Introduction

Relation Extraction (RE) is a fundamental task in Natural Language Processing (NLP) to detect relations between entities in a sentence. With the rise of a series of pre-trained language models (PLMs)(Devlin et al., 2018; Liu et al., 2019; Lewis et al., 2019; Raffel et al., 2020), fine-tuning PLMs has become the primary approach for relation extraction(Joshi et al., 2020; Xue et al., 2021; Zhou and Chen, 2021). The core idea of standard fine-tuning is shown in Fig. 1(a). However, fine-tuning requires adding additional classifiers on top of PLM and further training the model under the classification objective. Therefore their performance relies heavily on time-consuming and labor-intensive annotated data with poor generalization performance. In addition, the capabilities of PLMs may not be fully utilized as the training objectives of PLMs are different from the downstream tasks.

Prompt-tuning(Brown et al., 2020; Schick and Schütze, 2020a,b; Liu et al., 2023) has been proposed and proven effective, especially in low-resource scenarios(Gao et al., 2020; Scao and Rush, 2021). The core idea of prompt-tuning, as shown in Fig. 1(b), is to transform the goal of a downstream task into one closer to that of a pre-trained task. This is done by designing a template to reformulate the input examples into perfect-form phrases and linguistics to map labels to candidate words. The labels of the input examples can be determined by predicting mask tokens.




(a) Fine-tuning

(b) Prompt-tuning

Figure 1: Examples of fine-tuning and prompt-tuning core ideas

Despite the success of prompt-tuning on low-resource relation extraction tasks, they have paid little attention to exploiting the relations between a limited number of labeled samples. In addition, there is rich semantic knowledge in relation labels and rich structural knowledge between relation triples that cannot be ignored. The KnowPrompt(Chen et al., 2022) inspires this work, and in addition to injecting knowledge into the templates, this paper pays more attention to the relations between the samples. Specifically, in this paper, we first optimize the features of samples between different relations by constructing a global relation graph. Each node represents a labeled sample, and the edges between nodes represent the similarity between



*LREC-COLING 2024*, pages 13233–13242
20-25 May, 2024. 

the two samples. The boundaries are optimized in the global relation graph by determining whether two samples are from the same relation and regularising the similarity of the representations learned through PLM between every two samples. Secondly, this paper further optimizes the features of samples within the same relation by constructing a local relation subgraph. Specifically, this paper divides the global relation graph into multiple local relation subgraphs and translates this task into learning the similarity between the subgraphs and the target instances. The operation procedure is to represent the subgraph as a feature through an aggregation function and then calculate the similarity with the target instance. For example, sum pooling on the representation of all nodes in the subgraph is a practical and popular aggregation scheme. This method can improve the performance of prompt-based tuning strategy in low-resource relation extraction tasks.

This paper conducts extensive experiments on four widely used relation extraction datasets in a low-resource scenario setting. The method's effectiveness is demonstrated by achieving better performance compared with a range of recent baseline models. The contributions of this paper can be summarised as follows:

(i) In order to make full use of the limited sample resources, the features of the samples between different relations are optimized by constructing a global relation graph.

(ii) To further constrain the feature representation of the samples, the features of the samples within the same relation are optimized by constructing local relation subgraphs.

(iii) Extensive experiments on four widely used public datasets demonstrate the effectiveness of the present method in low-resource settings.

## 2. Related work

Relation extraction refers to extracting the relation between two entities given their related contexts. It plays an essential role in information extraction and knowledge base construction. Early approaches include pattern-based approaches(Huffman, 1995), CNN/RNN-based approaches(Zhou et al., 2016) and graph-based approaches(Guo et al., 2021). A prototypical network(Snell et al., 2017) is the more widely used metric-based meta-learning framework for few-shot relation extraction. It learns prototype vectors for each relation through several examples and then compares the similarity between query instances and prototype vectors of candidate relations for prediction. For example, Gao et al.(Gao et al., 2019) proposed an attention-based hybrid prototype network for noisy training samples in few-shot learning. Ye et al.(Ye and Ling, 2019) further proposed a multilevel matching and aggregation network for few-shot relation extraction. Peng et al.(Peng et al., 2020) demonstrated the effectiveness of applying metrics-based approaches to pre-trained models. Recent studies have used PLM for relation extraction tasks(Li et al., 2020; Wang et al., 2020; Ye et al., 2021). However, this paradigm remains sub-optimal due to the gap between pre-training and downstream tasks.

With the introduction of the GPT series, language model prompting emerged. By further exploring many prompt-tuning methods for relation extraction, a series of studies(Ben-David et al., 2021; Lester et al., 2021; Lu et al., 2021; Reynolds and McDonell, 2021) have been proposed and demonstrated the effectiveness of prompts. PET(Schick and Schütze, 2020a) reformulates input samples as perfect-form phrases to help language model runs understand a given task. ADAPET(Tam et al., 2021) modifies the goal of PET to provide more intensive supervision. PTR(Han et al., 2022) uses logic rules with multiple sub-prompts to encode prior knowledge in prompt-tuning. KnowPrompt(Chen et al., 2022) integrates knowledge between relation labels into prompt-tuning for relation extraction and provides co-optimization for better performance. In this paper, we would like to highlight the difference between KnowPrompt and the present method: while KnowPrompt only considers the introduction of knowledge, the present method pays more attention to the relations between a limited number of labeled samples along with the introduction of knowledge. This allows the present method to achieve better performance in low-resource scenarios.

## 3. Background

The relation extraction task $T$ with label space $R$ consists of three datasets: (i) The training dataset $D_{train} = \{(x_i, r_i)\}$ contains several labeled examples. $x_i$ is the sequence of samples and $r_i$ is the corresponding relation labels. (ii) The validation dataset $D_{dev}$. (iii) The test dataset $D_{test}$ contains unlabeled samples to be predicted.

For each sample instance $x$, the template maps $x$ to the prompt input $x_{prompt} = T(\cdot)$. Specifically, the template $T(\cdot)$ includes the location and number of additional words to add. In addition to preserving the original $token$ in $x$, one or more $[MASK]$ are placed in the $x_{prompt}$ prompt for the language model to populate with tagged words. For example, a sentence in a binary sentiment classification task:

$$x_i = [CLS]\ s\ [SEP]$$

Setting the template $T(\cdot) =$"$It\ is\ [MASK]$" maps $x$ to:

$$x_{prompt} = [CLS]\ s.\ It\ is\ [MASK].\ [SEP]$$

$[CLS]$ and $[SEP]$ are special start and end tokens. The hidden vector of $[MASK]$ is ob-

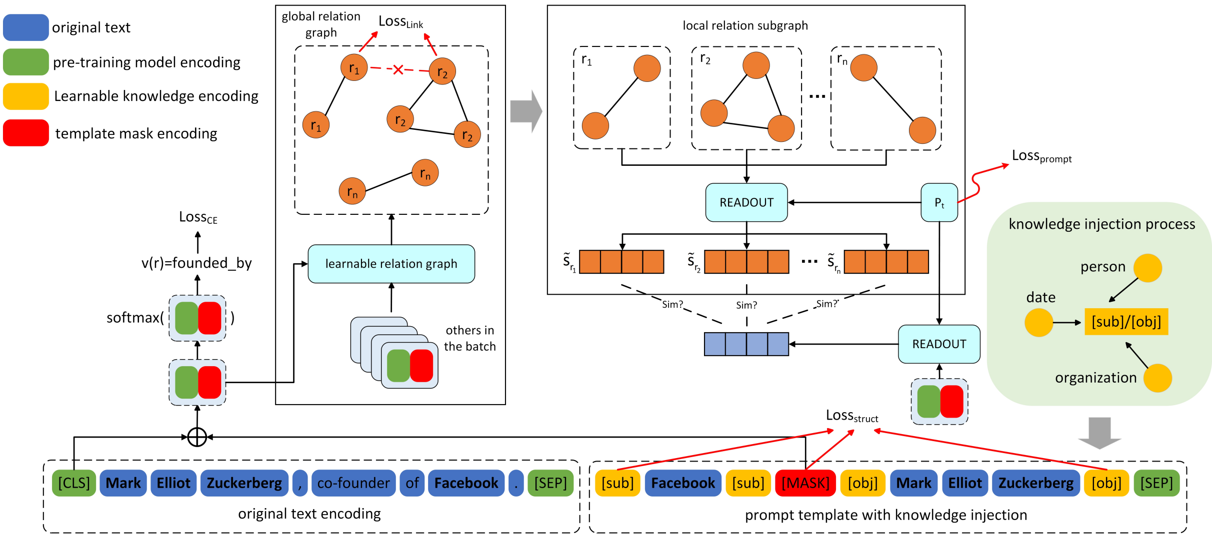



Figure 2: Model Architecture. This model consists of three main parts: the knowledge injection module, the global relation graph module and the local relation subgraph module, respectively.

tained by mapping $x$ to $x_{prompt}$ prompts via language model encoding. A probability distribution $p([MASK]|x_{prompt})$ describes which tokens are suitable for replacing $[MASK]$ words. The linguist $v(\cdot)$ maps $r_i$ to symbols representing the semantics of $r_i$. For example, "$positive/negative$" can be mapped to "$great/terrible$". Depending on whether the language model predicts "$great$" or "$terrible$", the labels of the sample instance $x$ can be identified as "$positive$" or "$negative$".

Because the language model predicts the correct labeling of the mask position, it is possible to formalize the mask position as $p(r|x)$, that is:

$$p(r|x) = p([MASK] = v(r)|x_{prompt}) \tag{1}$$

Given the probability distribution $p(r|x)$ on the masked location, the loss function is computed by the cross-entropy between $r$ and $p(r|x)$. The computation of $\mathcal{L}_{CE}$ is shown below:

$$\mathcal{L}_{CE} = -\frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \mathrm{r} \log p(r \mid x) \tag{2}$$

where $|\mathcal{X}|$ denotes the number of training data.

## 4. Methodology

This section presents the general framework of prompt tuning for few-shot relation extraction via modeling global and local graphs. The specific model architecture is shown in Fig. 2, and this section details how to construct and optimize this model.

### 4.1. Knowledge Injection Module

Traditional prompts are composed of two parts: a template and a set of labeled words, while the rich semantic knowledge in relation labels should not be ignored. Inspired by the KnowPrompt approach, this paper injects entity-related knowledge into constructing prompt templates.

#### 4.1.1. Prompt Template Build with Knowledge Injection

The direct introduction of type information for entities requires additional annotation, which is not always available in datasets. Therefore, this paper obtains the scope of potential entity types through the a priori knowledge contained in the relation rather than annotations. For example, given the relation "$per : country\_of\_birth$", it is clear that the subject entity matching this relation belongs to "$person$", while the object entity matching this relation belongs to "$country$". Intuitively, prior distributions $\phi_{sub}$ and $\phi_{obj}$ can be estimated based on the relation classes on the sets of potential entity type candidates $C_{sub}$ and $C_{obj}$, respectively. Frequency statistics estimate the prior distributions. Some of the relations taken for $C_{sub}$ and $C_{obj}$ are shown in the labels listed in Table 1. For example, the prior distribution $C_{sub}$ is computed as follows: $\phi_{sub} = \{"organization" : 3/6, "person" : 3/6\}$. Thus, assigning type words around entities can be initialized with potential entity types to aggregate embeddings. Since the initialized type words are not the exact type of a particular entity, the learnable type words can be dynamically adjusted according to the context. The specific initialization method is shown below:

$$\hat{\mathbf{e}}_{[sub]} = \sum \phi_{sub} \cdot \mathbf{e}\left(I\left(\mathbf{C}_{sub}\right)\right) \tag{3}$$

$$\hat{\mathbf{e}}_{[obj]} = \sum \phi_{obj} \cdot \mathbf{e}\left(I\left(\mathbf{C}_{obj}\right)\right) \tag{4}$$

where $\hat{\mathrm{e}}_{[sub]}$ and $\hat{\mathrm{e}}_{[obj]}$ denote the embedding of

Table 1: Some examples of relations in the TACREV dataset

| **Relation Labels** | $C_{sub}$ | $C_{obj}$ | $C_r$ |
|---|---|---|---|
| per:country_of_birth | person | country | {“country”, “of”, “birth” } |
| per:date_of_death | person | date | {“date”, “of”, “death” } |
| per:schools_attended | person | organization | {“school”, “attended’} |
| org:alternate_names | organization | organization | {“alternate”, “names” } |
| org:city_of_headquarters | organization | city | {“city”, “of”, “headquarters” } |
| org:number_of_employees/members | organization | number | {“number”, “of”, “employees”, “members” } |

type words around subject and object entities. $I(\cdot)$ is a deletion operation on set duplicates. $\mathbf{e}$ is the word embedding encoded by the language model. Type words provide an initial sense of the range of entity types and express semantic information close to the actual entity types.

#### 4.1.2. Loss of Knowledge Injection Module

This paper employs additional structured constraints to optimize the prompts, integrating structured knowledge into the model. Specifically, we use the triple $(s, r, o)$ to describe relation facts. Here $s$ and $o$ denote the types of subject and object entities, respectively. $r$ is the relation label. In this model, the computation is involved by using the output embedding of the language model on the entity type words. In this paper, we define the loss of implicit structured constraints $\mathcal{L}_{struct}$ as follows:

$$\begin{aligned}\mathcal{L}_{struct} &= -\log \sigma(\gamma - d_r(\mathrm{s}, \mathrm{o})) \\ &-\sum_{i=1}^{n} \frac{1}{n} \log \sigma(d_r(\mathrm{s}_i', \mathrm{o}_i') - \gamma)\end{aligned} \tag{5}$$

$$d_r(\mathrm{s}, \mathrm{o}) = ||\mathrm{s} + \mathrm{r} - \mathrm{o}||_2 \tag{6}$$

where $(\mathrm{s}_i', \mathrm{r}, \mathrm{o}_i')$ is the negative sample and $\gamma$ is the margin. $\sigma$ is the $sigmoid$ function and $d_r$ is the scoring function. This paper assigns the correct type of relation for negative sampling at the $[MASK]$ position. Negative samples are constructed by randomly sampling subject or object entities to form unrelated triples, where entities are impossible types for the current relation.

### 4.2. Global Relation Graph Module

In this section, this paper proposes to construct a global relation graph to optimize the feature representation of samples between different relations. More supervised signals are mined from the training samples by constructing a global relation graph, which can improve the effectiveness of prompt-based tuning.

#### 4.2.1. Construction of Global Relation Graph

Consider a batch $S = \{(x_i, r_i)\}_{i=1}^{N}$ containing $N$ randomly sampled pairs of labels. Its index is kept at $I = \{1, \cdots, N\}$. More information is developed by building a global relation graph of the sample information. Let $G = \{V, E\}$ denote the relation graph between $N$ training samples in $S$, where $V$ is the set of nodes. Each node $v_i \in V$ corresponds to a training sample $x_i$, and $E = \{e_{ij}\}$ is the set of edges between the $N$ training samples. If a node $v_i$ and another node $v_j$ belong to the same relation class, an edge $e_{ij}$ is created between them. Specifically, $e_{ij}$ is set as:

$$e_{ij} = \begin{cases} 1 & r_j = r_i \\ 0 & otherwise \end{cases} \tag{7}$$

#### 4.2.2. Loss of Global Relation Graph Module

On a global relation graph $G$ of a batch $S$, this paper converts this task into a link prediction problem. That is, connecting nodes with the same relation and disconnecting nodes with different relations. In this paper, we obtain the relation prediction $\hat{\mathbf{r}}_i$ for $v_i$ (corresponding to $x_i$) by using equation 1.

In this paper, $\hat{\mathbf{r}}_i$ and $\hat{\mathbf{r}}_j$ are established between $v_i$ and $v_j$ based on the correlation of $\hat{e}_{ij}$:

$$\hat{e}_{ij} = g\left(\hat{\mathbf{r}}_i, \hat{\mathbf{r}}_j\right) \tag{8}$$

where $\hat{\mathbf{r}}_i$ and $\hat{\mathbf{r}}_j$ are calculated from equation 1. $g(\cdot, \cdot)$ is the computed cosine similarity. In order to measure the loss of link prediction, the loss of $\mathcal{L}_{Link}$ is designed in this paper as:

$$\begin{aligned}\mathcal{L}_{Link} = -\sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{A}(i)} & e_{ij} \log\left(\hat{e}_{ij}\right) \\ &+ \left(1 - e_{ij}\right) \log\left(1 - \hat{e}_{ij}\right)\end{aligned} \tag{9}$$

where

$$\mathcal{A}(i) = \{j \in \mathcal{I} \; and \; i \neq j\} \tag{10}$$

### 4.3. Local Relation Subgraph Module

Based on constructing a global relation graph, this paper further optimizes the features of samples within the same relation by constructing a local relation subgraph.

#### 4.3.1. Construction of Local Relation Subgraph

In a global relation graph $G = (V, E)$ containing a set of relation classes $R$ and a set of labeled nodes $D = \{(v_1, l_1), (v_2, l_2), \cdots\}$, where $v_i \in V$ and the corresponding labels of $v_i$ are $l_i$. When the $k - shot$ setting is used in this paper, for each relation $r \in R$, there are exactly $k$ pairs $(v_i, l_i = r) \in D$. A local relation subgraph is further formed for each relation $r \in R$. The local relation subgraph is then converted to an average representation of the current relation class. The current relation is represented by the vector $\tilde{\mathbf{S}}_r$, which is computed as shown below:

$$\tilde{\mathbf{S}}_r = \frac{1}{k} \sum_{(v_i, \ell_i) \in D, \ell_i = r} \mathbf{p}_{t_i} \odot \mathbf{h}_{v_i} \tag{11}$$

where $\mathbf{h}_{v_i}$ denotes the feature vector of node $v_i$ and $\mathbf{p}_t$ denotes the learnable prompt vector. More recently, this paper abstracts this step into a $READOUT$ operation to aggregate node representations in subgraphs, and a prompt-assisted $READOUT$ operation on a local relation subgraph is shown below:

$$\mathbf{S}_x = READOUT(\{\mathbf{p}_t \odot \mathbf{h}_v : v \in V_x\}) \tag{12}$$

where $\mathbf{S}_x$ is the feature representation of the local relation subgraph $x$. $\odot$ denotes the element multiplication and $V_x$ denotes the set of nodes in the subgraph $x$. Specifically, in this paper, the node representations in the subgraph are feature-weighted and summed, where the prompt vector $\mathbf{p}_t$ is dimensionally re-weighted in order to extract the most relevant a priori knowledge for the relation extraction task.

There are two points to be made about this part of the paper. First, the choice of $READOUT$ aggregation scheme is flexible, including sum pooling and more advanced techniques. In the implementation of this paper, sum pooling is used. Second, this paper uses learnable prompt vectors $\mathbf{p}_t$ instead of language-based prompts. The main reason is that the present prompts are designed for graph structures. Thus, they are more abstract and cannot be language-based instructions. Instead, they should be topology-related and aligned with the core of graph learning. In particular, the critical properties may differ for different relation classes in the task.

After obtaining the average representation $\tilde{\mathbf{S}}_r$ of the local relation subgraphs, given a node $v_j$ that is not in the labeled set $D$, its class label $l_j$ can be represented as:

$$\ell_j = \arg\max_{r \in R} \operatorname{sim}\left(\mathbf{S}_{v_j}, \tilde{\mathbf{S}}_r\right) \tag{13}$$

Intuitively, a node should belong to the relation class most similar to the local relation subgraph.

#### 4.3.2. Loss of Local Relation Subgraph Modules

This paper formulates the loss based on subgraph similarity to optimize the learnable prompt $\mathbf{p}_t$. In the labeled training set for the relation extraction task $T = \{(x_1, y_1), \cdots, (x_i, y_i), \cdots, (x_n, y_n)\}$. $x_i$ is the training sample (i.e., node representation), and $y_i \in R$ is the labeling of $x_i$ in the set of relation classes $R$. The loss of prompt tuning $\mathcal{L}_{prompt}$ is defined as:

$$\mathcal{L}_{prompt}(\mathbf{p}_t) = - \sum_{(x_i, y_i) \in T} \ln \frac{exp(sim(\mathbf{S}_{x_i}, \tilde{\mathbf{S}}_{y_i})/k)}{\sum_{r \in R} exp(sim(\mathbf{S}_{x_i}, \tilde{\mathbf{S}}_r)/k)} \tag{14}$$

where the local relation subgraph aggregation feature for relation $r$ is represented by $\tilde{\mathbf{S}}_r$, which is also generated by the prompting aid $READOUT$. The prompt tuning loss is only generated by the learnable prompt vector $\mathbf{p}_t$. Since no fine-tuning is required, this greatly reduces the number of parameters that need to be updated. Not only does it improve the computational efficiency of learning and inference, but it also reduces the dependence on labeled data.

### 4.4. Joint Training

In this paper, a total of four types of losses are defined in the joint training: the cross-entropy loss $\mathcal{L}_{CE}$ in relation type prediction, the structuring loss $\mathcal{L}_{struct}$ in the knowledge injection module, the link prediction loss $\mathcal{L}_{Link}$ in the global relation graph module, and the learnable prompts loss $\mathcal{L}_{prompt}$ in the local relation subgraphs. For $\mathcal{L}_{CE}$ is obtained using equation 2. For $\mathcal{L}_{struct}$ calculated using equation 5. For $\mathcal{L}_{Link}$ calculated using equation 9. For $\mathcal{L}_{prompt}$ is calculated using equation 14. The total loss of the joint training is equal to the sum of the four types of losses. The overall loss $Loss$ is calculated as follows:

$$Loss = \mathcal{L}_{CE} + \alpha \mathcal{L}_{struct} + \beta \mathcal{L}_{Link} + \gamma \mathcal{L}_{prompt} \tag{15}$$

where $\alpha$, $\beta$ and $\gamma$ are the weight parameters between losses. This model is jointly trained by minimizing $Loss$.

Table 2: Statistical results of dataset information

| Dataset | Train | Val | Test | Rel |
|---|---|---|---|---|
| SemEval | 6,507 | 1,493 | 2,717 | 19 |
| TACRED | 68,124 | 22,631 | 15,509 | 42 |
| TACREV | 68,124 | 22,631 | 15,509 | 42 |
| Re-TACRED | 58,465 | 19,584 | 13,418 | 40 |

Table 3: Comparative results of models

| | Model | TACRED | | | TACREV | | | Re-TACRED | | | SemEval | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | K=8 | K=16 | K=32 | K=8 | K=16 | K=32 | K=8 | K=16 | K=32 | K=8 | K=16 | K=32 |
| Fine-tuning | SpanBERT (Joshi et al., 2020) | 8.4 | 17.5 | 17.9 | 5.2 | 5.7 | 18.6 | 14.2 | 29.3 | 43.9 | - | - | - |
| | LUKE (Yamada et al., 2020) | 9.5 | 21.5 | 28.7 | 9.8 | 22.0 | 29.3 | 14.1 | 37.5 | 52.3 | - | - | - |
| | GDPNet (Xue et al., 2021) | 11.8 | 22.5 | 28.8 | 8.3 | 20.8 | 28.1 | 18.8 | 48.0 | 54.8 | 42.0 | 67.5 | 81.2 |
| | TANL (Paolini et al., 2021) | 18.1 | 27.6 | 32.1 | 18.6 | 28.8 | 32.2 | 26.7 | 50.4 | 59.2 | - | - | - |
| Prompt-tuning | TYP Marker (Zhou and Chen, 2021) | 28.9 | 32.0 | 32.4 | 27.6 | 31.2 | 32.0 | 44.8 | 54.1 | 60.0 | - | - | - |
| | PTR (Han et al., 2022) | 28.1 | 30.7 | 32.1 | 28.7 | 31.4 | 32.4 | 51.5 | 56.2 | 62.1 | 70.5 | 81.3 | 84.2 |
| | KnowPrompt (Chen et al., 2022) | 32.0 | 35.4 | 36.5 | 32.1 | 33.1 | 34.7 | 55.3 | 63.3 | 65.0 | 74.3 | 82.9 | **84.8** |
| | **ours** | **32.7** | **36.5** | **39.2** | **33.9** | **36.3** | **38.1** | **58.8** | **64.4** | **66.3** | **79.0** | **83.0** | 84.6 |

Table 4: Ablation study results

| Model | TACRED | | | TACREV | | |
|---|---|---|---|---|---|---|
| | K=8 | K=16 | K=32 | K=8 | K=16 | K=32 |
| ours | 32.7 | 36.5 | 39.2 | 33.9 | 36.3 | 38.1 |
| -global relation graph | 31.9 | 36.0 | 36.2 | 32.8 | 35.1 | 36.6 |
| -local relation subgraph | 32.2 | 36.1 | 36.4 | 32.2 | 35.4 | 36.7 |
| -knowledge injection | 32.4 | 35.8 | 35.3 | 32.4 | 35.5 | 37.9 |

## 5. Experiment

This paper evaluates the validity of the methodology on four datasets. The method is compared with state-of-the-art baseline models, and further experiments and analyses are conducted on the essential components of the model.

### 5.1. Dataset

In this paper, experiments are conducted on four widely used public datasets, SemEval 2010 Task 8 (SemEval)(Hendrickx et al., 2019), TACRED(Zhang et al., 2017), TACREV(Alt et al., 2020) and Re-TACRED(Stoica et al., 2021). Table 2 lists the details of each dataset.

### 5.2. Baseline Comparison

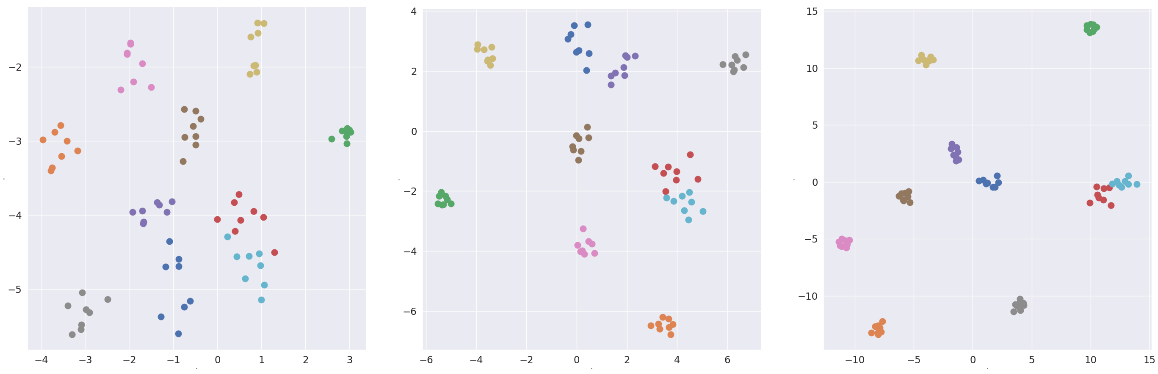
Table 3 reports the experimental results of the different models on the four datasets. In this paper, all models are classified into two categories: one is fine-tuning-based, and the other is prompt-based. This paper sets each dataset up as three different low-resource scenarios for separate experiments. Specifically, it includes the scenario with sample size K=8, the scenario with sample size K=16 and the scenario with sample size K=32. The results show that the proposed method performs well on all four datasets, which indicates that the proposed method is effective.

This paper finds that the present model performs better by comparing it with recent work. Specifically, the present model outperforms the baseline model in both the fine-tuning-based and prompt-based approaches. Furthermore, the present model performs well in very low-resource scenarios with four datasets.

### 5.3. Ablation Study

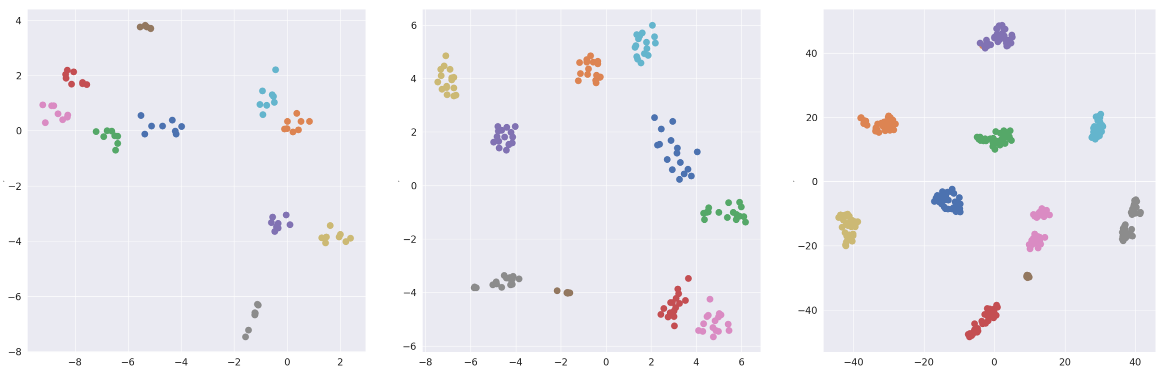
In order to validate the effectiveness of the main components in the model, this paper conducts ablation studies on two publicly available datasets. The effects of different components on the model performance are compared by removing one component at a time, and the results of the ablation studies are shown in Table 4.

In this paper, a total of three types of ablation studies are conducted on two public datasets, namely: removing the global relation graph module (-global relation graph), removing the local relation subgraph module (-local relation subgraph) and removing the knowledge injection module (-

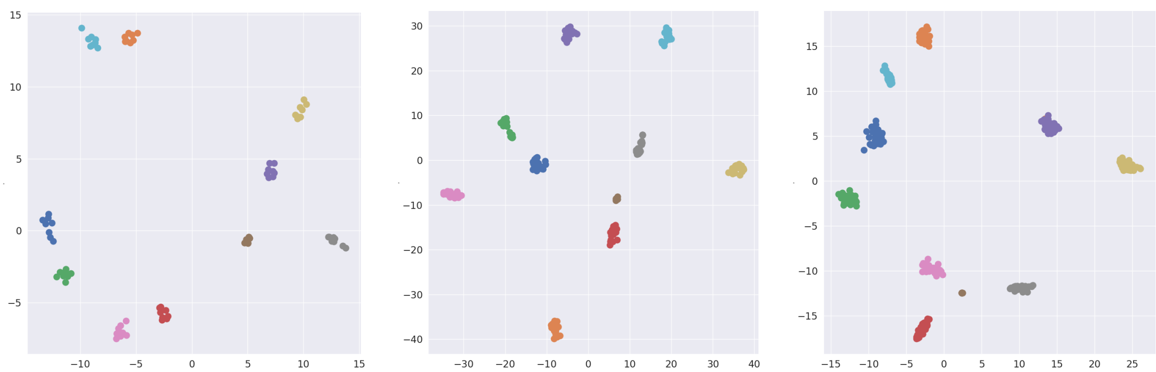


(a) Results of feature visualization after adding knowledge injection (b) Results of feature visualization after adding global relation graph (c) Results of feature visualization after adding local relation subgraphs

Figure 3: Results of the visualization of the features at each stage



(a) Feature visualization results for KnowPrompt with K=8 (b) Feature visualization results for KnowPrompt with K=16 (c) Feature visualization results for KnowPrompt with K=32



(d) Feature visualization results of our model for K=8 (e) Feature visualization results of our model for K=16 (f) Feature visualization results of our model for K=32

Figure 4: Results of feature visualization comparison between our model and KnowPrompt

knowledge injection). The experimental results show that the model's performance is degraded to varying degrees when the essential components are removed. This shows that the critical components in this paper are effective.

## 5.4. Feature Visualization and Analysis

In order to visualize the changes in feature distribution more intuitively, this paper uses t-SNE on the Re-TACRED dataset to map the feature distribution in two-dimensional space. Firstly, this paper performs feature visualization at various stages of the model to verify the validity of the components.

Second, this paper compares with a similar method, KnowPrompt.

#### 5.4.1. Visualization of Features at Each Stage of the Model

In order to verify the validity of each component in the model, this paper visualizes the features at each stage of the model. The distribution of features at each stage is shown in Fig. 3. Among them, Fig. 3(a) represents the feature visualization results after adding knowledge injection, Fig. 3(b) represents the feature visualization results after adding a global relation graph based on knowledge injection, and Fig. 3(c) represents the feature visualization results after adding local relation subgraph based on global relation graph.

As shown in Fig. 3(a), the features have produced a more obvious aggregation phenomenon after introducing knowledge injection based on prompt-tuning. However, the distance between different relations is close, and the distribution of features within the same relation is loose. This may lead to misclassification when the relations are finally judged. From Fig. 3(b), which introduces the global relation graph, it can be seen that the distance between different relations becomes farther. As can be seen in Fig. 3(c), which introduces the local relation subgraph, the distribution of features within the same relation also becomes more clustered. However, when the relations are more similar (e.g., blue relation and red relation), the model still cannot distinguish them well. t-SNE visualization also indirectly proves the validity of the present model.

#### 5.4.2. Comparison of Model Visualizations Under Different Conditions

Since the KnowPrompt method is more similar to the one in this paper, a detailed comparison is made. In this paper, the feature visualization results of the two methods are set for K=8, K=16 and K=32, respectively. The visualization results of the two methods under different settings are shown in Fig. 4. Among them, Fig. 4(a), Fig. 4(b) and Fig. 4(c) represent the feature visualization results of the KnowPrompt method under three settings. Fig. 4(d), Fig. 4(e) and Fig. 4(f) represent the feature visualization results of our method under three settings.

The results show that this method exhibits better aggregation than the KnowPrompt method for K=8 and K=16. This is due to the constraints of the global relation graph and local relation subgraph optimizing the representation of features. Whereas at K=32, both methods show better performance with little difference in results. This is because the number of samples for each type of relation has reached 32, and both methods have enough sample size to learn a better feature representation.

## 6. Conclusion

This paper proposes prompt tuning for few-shot relation extraction via modeling global and local graphs. The method constructs a global relation graph based on the labeling consistency between samples, thus optimizing the feature representation of samples between different relations. The global relation graph is then further divided to form local relation subgraphs, thus optimizing the feature representations of samples within the same relation. This fully uses the limited supervised information and improves the tuning efficiency. This paper validates the method's effectiveness on four publicly available datasets. The results show that the proposed model performs better in low-resource scenarios than existing methods.

## 7. References


Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. 2020. Tacred revisited: A thorough evaluation of the tacred relation extraction task. *arXiv preprint arXiv:2004.14855*.

Eyal Ben-David, Nadav Oved, and Roi Reichart. 2021. Pada: A prompt-based autoregressive approach for adaptation to unseen domains. *arXiv preprint arXiv:2102.12206*, 3.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web conference 2022*, pages 2778–2788.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Tianyu Gao, Xu Han, Hao Zhu, Zhiyuan Liu, Peng Li, Maosong Sun, and Jie Zhou. 2019. Fewrel

2.0: Towards more challenging few-shot relation classification. *arXiv preprint arXiv:1910.07124*.

Zhijiang Guo, Guoshun Nan, Wei Lu, and Shay B Cohen. 2021. Learning latent forests for medical relation extraction. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3651–3657.

Xu Han, Weilin Zhao, Ning Ding, Zhiyuan Liu, and Maosong Sun. 2022. Ptr: Prompt tuning with rules for text classification. *AI Open*, 3:182–192.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid O Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2019. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. *arXiv preprint arXiv:1911.10422*.

Scott B Huffman. 1995. Learning information extraction patterns from examples. In *International Joint Conference on Artificial Intelligence*, pages 246–260. Springer.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the association for computational linguistics*, 8:64–77.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Juan Li, Ruoxu Wang, Ningyu Zhang, Wen Zhang, Fan Yang, and Huajun Chen. 2020. Logic-guided semantic representation learning for zero-shot relation classification. *arXiv preprint arXiv:2010.16068*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

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*.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2021. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. *arXiv preprint arXiv:2101.05779*.

Hao Peng, Tianyu Gao, Xu Han, Yankai Lin, Peng Li, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2020. Learning from context or names? an empirical study on neural relation extraction. *arXiv preprint arXiv:2010.01923*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Laria Reynolds and Kyle McDonell. 2021. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–7.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? *arXiv preprint arXiv:2103.08493*.

Timo Schick and Hinrich Schütze. 2020a. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Timo Schick and Hinrich Schütze. 2020b. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30.

George Stoica, Emmanouil Antonios Platanios, and Barnabás Póczos. 2021. Re-tacred: Addressing shortcomings of the tacred dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13843–13850.

Derek Tam, Rakesh R Menon, Mohit Bansal, Shashank Srivastava, and Colin Raffel. 2021. Improving and simplifying pattern exploiting training. *arXiv preprint arXiv:2103.11955*.

Zifeng Wang, Rui Wen, Xi Chen, Shao-Lun Huang, Ningyu Zhang, and Yefeng Zheng. 2020. Finding influential instances for distantly supervised relation extraction. *arXiv preprint arXiv:2009.09841*.

Fuzhao Xue, Aixin Sun, Hao Zhang, and Eng Siong Chng. 2021. Gdpnet: Refining latent multi-view graph for relation extraction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14194–14202.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: deep contextualized entity representations with entity-aware self-attention. *arXiv preprint arXiv:2010.01057*.

Hongbin Ye, Ningyu Zhang, Shumin Deng, Mosha Chen, Chuanqi Tan, Fei Huang, and Huajun Chen. 2021. Contrastive triple extraction with generative transformer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 14257–14265.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Multi-level matching and aggregation network for few-shot relation classification. *arXiv preprint arXiv:1906.06678*.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Conference on Empirical Methods in Natural Language Processing*.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 207–212.

Wenxuan Zhou and Muhao Chen. 2021. An improved baseline for sentence-level relation extraction. *arXiv preprint arXiv:2102.01373*.