Minimally-Supervised Relation Induction from Pre-trained Language Model

Lu Sun, Yongliang Shen, Weiming Lu *
College of Computer Science and Technology, Zhejiang University
{sunlu98,syl,luwm}@zju.edu.cn

Abstract

Relation Induction is a very practical task in Natural Language Processing (NLP) area. In practical application scenarios, people want to induce more entity pairs having the same relation from only a few seed entity pairs. Thus, instead of the laborious supervised setting, in this paper, we focus on the minimally-supervised setting where only a couple of seed entity pairs per relation are provided. Although the conventional relation induction methods have made some success, their performance depends heavily on the quality of word embeddings. The great success of Pre-trained Language Models, such as BERT, changes the NLP area a lot, and they are proven to be able to better capture relation knowledge. In this paper, we propose a novel method to induce relation with BERT under the minimally-supervised setting. Specifically, we firstly extract proper templates from the corpus by using the mask-prediction task in BERT to build pseudo-sentences as the context of entity pairs. Then we use BERT attention weights to better represent the pseudo-sentences. In addition, We also use the Integrated Gradient of entity pairs to iteratively select better templates further. Finally, with the high-quality pseudo-sentences, we can train a better classifier for relation induction. Experiments on Google Analogy Test Sets (GATS), Bigger Analogy Test Set (BATS) and DiffVec demonstrate that our proposed method achieves state-of-the-art performance.

1 Introduction

Relation induction is a task to judge whether two entities have a certain relation based on some given entity pairs of that relation, which was first proposed in (Vylomova et al., 2016). For instance, given $\{(Germany, Berlin), (France, Pairs), (Italy, Rome)\}$, relation induction is to predict whether new entity pairs such as $(China, Beijing)$ have the same relation as the given entity pairs. In practical scenarios, only a few seed entity pairs are available. It is challenging to judge the relation of the target entity pairs in this minimal supervision setting.

Word embedding, such as skip-gram (Mikolov et al., 2013a) and Glove (Pennington et al., 2014), are widely used in many natural language processing (NLP) tasks, and it was reported that word embeddings can capture the relational knowledge (Mikolov et al., 2013b). One intuitional method for relation induction task is using word embeddings to represent relations and induce relations based on vector translation or similarity (Vylomova et al., 2016; Drozd et al., 2016; Bouraoui et al., 2018; Vulić and Mrkšić, 2018; Camacho-Collados et al., 2019). However, the performance of these methods heavily depends on the pre-trained word embedding and these methods are rather noisy. According to the assumption that if two entities have a relationship in a known knowledge base, then all sentences that mention these two entities will express that relationship in some way (Mintz et al., 2009), many distant-supervised methods of relation extraction, such as PCNN(Zeng et al., 2015) and PCNN-BagATT (Ye and Ling, 2019) are proposed. Inspired by these methods, distant supervision might be another way to induce relation. To induce relation in the distant supervised way, we need a method to select proper sentences from corpus and extract relational knowledge from sentences. Luckily, many Pre-trained Language Models (PLMs), such as BERT(Devlin et al., 2019), GPT-2 (Radford et al., 2019) and XLNet(Yang et al., 2019), have been recently proposed and boost a great performance for many NLP tasks, such as question answering(Talmor et al., 2019; Feng et al., 2020), text summarization (Liu and Lapata, 2019; Lewis et al., 2020) and information extraction (Petroni et al., 2019; Alt et al., 2019). In order to better understand the PLMs, several works(Kim et al., 2020; Bouraoui et al., 2020; Ushio et al., 2021; Chen

[*] corresponding author

Findings of the Association for Computational Linguistics: NAACL 2022, pages 1776 - 1786
July 10-15, 2022 ©2022 Association for Computational Linguistics

et al., 2021) have proven that PLMs can capture syntactic and semantic knowledge. Bouraoui et al. (2020) have explored the possibility of inducing relation from BERT in a distant supervised way and got a good result. To take the advantage that BERT can capture context knowledge, they select templates from corpus and fill entities in them to let BERT predict the relation.

Existing methods are developed under the assumption of sufficient seed entity pairs. However, in practical scenarios, only a few entity pairs are available for a particular relation. These methods have difficulty in coping with the minimal supervision setting. The main reasons are: (1) Due to the lack of labeled entity pairs, the model tends to over-focus on the surface cues of the entity pairs and ignores the contextual semantics. By simply memorizing the seed entity pairs, it is difficult to generalize the model to other entity pairs. (2) The quality of templates is very important for relation induction.When the seed entity pairs of a certain relation are sparse, the number of candidate templates for this relation mined from the corpus will be reduced.

Therefore, two major challenges should be addressed for the relation induction in the minimally-supervised setting. (1) How to obtain a good generalized relation induction model? (2) How to obtain high-quality templates? So we propose a novel approach called IST for minimally-supervised relation induction with Iteratively-Selected Templates from PLM. Specifically, for the first challenge, we use surface-agnostic features based on attention maps of BERT. Many works (Clark et al., 2019; Kovaleva et al., 2019; Michel et al., 2019; Wang et al., 2020) have revealed that the attention heads in BERT can capture much knowledge and some attention heads are related to certain relations, and some works use attention weights to predict relations (Gu et al., 2021). For the second challenge, we use Integrated Gradient (IG) (Sundararajan et al., 2017) to score the templates and iteratively select better templates. Intuitively, if a sentence can well express the relational knowledge between two entities, then the importance of these two entities must be high in the sentence. On the contrary, if a pair of entities do not play an important role in a sentence, this sentence certainly does not express the relationship between them. So IG might be used to select high-quality sentences to express relations.

We summarize our key contributions as follows:

- We propose a novel minimally-supervised relation induction approach IST. To the best of our knowledge, we are the first to address the minimally-supervised relation induction task.
- In order to overcome the minimally-supervised setting, we generate high-quality pseudo-sentences by iteratively selecting templates based on BERT and IG scores. Moreover, we use attention maps to train a more generalized model.
- We conduct extensive experiments on three standard benchmark datasets, and our proposed approach significantly outperforms the state-of-the-art approaches.

2 Our Approach

In this section, we first formulate the minimally-supervised relation induction task and give an overview of our approach. We then describe the details of each module in our approach.

2.1 Problem Formulation

Given a few seed entity pairs $P_r = \{(s_i, t_i)\}_{i=1}^{N}$ with a certain relation r, the task of relation induction is to judge whether a new entity pair (s, t) also has the relation r. In the minimally-supervised setting, the number of the seed pairs is small for each relation (in our experiments, no more than 5 per relation). To facilitate minimally-supervised relation induction, we generate high-quality pseudo sentences S_r for each relation r from a text corpus C according to the seed entity pairs P_r with the help of a pre-trained language model.

2.2 Overview

As illustrated in Figure 1, our approach consists of four main modules: template generation module, pseudo sentence generation module, relation classifier and template selection with IG.

In the template generation module, given seed entity pairs, some proper templates could be generated based on the mask-prediction results in BERT. For instance, considering the seed entity pairs $P_r = \{(Germany, Berlin), (France, Paris), (Japan, Tokyo)\}$, we can obtain a sentence set S_r where each sentence mentions both entities of a pair in P_r.Taking a sentence $The\ current\ capital\ of\ Japan\ is\ Tokyo.$ as an

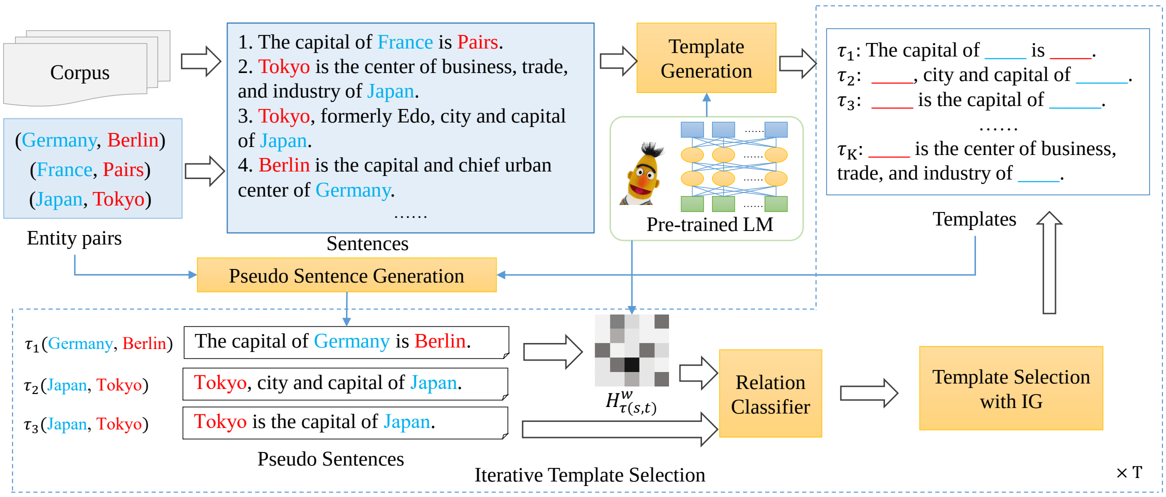
Corpus
(Germany, Berlin)
(France, Pairs)
(Japan, Tokyo)
Entity pairs
1. The capital of France is Pairs.
2. Tokyo is the center of business, trade, and industry of Japan.
3. Tokyo, formerly Edo, city and capital of Japan.
4. Berlin is the capital and chief urban center of Germany.
……
Sentences
Template Generation
Pre-trained LM
τ_1: The capital of ____ is ____.
τ_2: ____, city and capital of ____.
τ_3: ____ is the capital of ____.
……
τ_K: ____ is the center of business, trade, and industry of ____.
Templates
Pseudo Sentence Generation
τ_1(Germany, Berlin)
The capital of Germany is Berlin.
τ_2(Japan, Tokyo)
Tokyo, city and capital of Japan.
τ_3(Japan, Tokyo)
Tokyo is the capital of Japan.
Pseudo Sentences
$H^w_{\tau(s,t)}$
Relation Classifier
Template Selection with IG
Iterative Template Selection
× T

Figure 1: An overview of IST. First, we extract sentences that mention seed entity pairs as candidate sentences. Then, the template generation module uses BERT-prediction task to select proper templates from candidates. Templates and seed entity pairs are assembled to generate pseudo sentences to extract relational knowledge from BERT. The BERT attention weights between entities within the pseudo sentence are used as surface-agnostic features to better represent the relational knowledge. Then, the pseudo sentences and attention weights are combined to train a BERT-based relation classifier. Finally, we use the integrated gradient of entity pairs in pseudo sentences to evaluate the quality of templates and select better templates iteratively.

example, $\tau = (The\ current\ capital\ of\ \_\ is\ \_)$ is the generated template. Then, filling one entity into templates, the templates can be scored according to their ability to make BERT correctly predict another entity. This score is referred to as score$_{BERT}$.

After selecting proper templates based on score$_{BERT}$, we can generate pseudo sentences by assembling the templates and seed entity pairs. For example, a pseudo sentence $\tau(Germany, Berlin)$ = $The\ current\ capital\ of\ \underline{Germany}\ is\ \underline{Berlin}.$ is generated by assembling $(Germany, Berlin)$ and τ. We generate both positive and negative sentences in this process.

For each pseudo-sentence, we extract surface-agnostic features based on attention weight maps of BERT and use them to train a relation classifier.

Finally, we use integrated gradient (IG) together with score$_{BERT}$ to evaluate the quality of templates again, so we can refine templates iteratively.

We will describe each module in detail in the following sections.

2.3 Template Generation

To induce relation from masked pre-trained language models such as BERT, we need templates for relations. First, many template-based relation extraction methods(Agichtein and Gravano, 2000; Ravichandran and Hovy, 2002) have proved that words near to s and t in corpus may represent a certain relation. To extract templates for relation r, we traverse Wiki Corpus to find k_i sentences that contain both s_i and t_i(i≤N), and the distance between s_i and t_i in sentence $D_{st} \leq d$. Then we mask s_i and t_i in these senteces to generate templates $\tau_{i,1}, \tau_{i,2}...\tau_{i,k}$. We can extract all candidate templates for r: $\{\tau_{1,1}, \tau_{1,2}...\tau_{i,j}...\tau_{N,k_N}\}$ $(i \leq N, j \leq k_i)$, but not all of these templates are proper for inducing the relation r.

Then we need to select templates that are proper for BERT to induce relation r. Here we use BERT mask prediction as a template filter (Bouraoui et al., 2020). Specifically, for a template τ, insert s and t into τ respectively to get masked sentence $\tau(s, \_)$ and $\tau(\_, t)$. Then let BERT predict the masked token. If BERT can predict correctly, we consider the template τ is proper for relation r and $\tau(s, t)$ is "natural" for BERT.

$$score_{BERT}(\tau) = \sum_{i=0}^{N}(M(\tau(s_i, \_))+M(\tau(\_, t_i))) \tag{1}$$

where $M(\tau(s_i, \_))$ is 1 if the predicted token is t_i and 0 otherwise, and similar for $M(\tau(\_, t_i))$ = 1.

By ranking templates with $score_{BERT}$, K proper templates $T_r = \{\tau_1, \tau_2...\tau_K\}$ are selected from candidate templates.

2.4 Pseudo Sentence Generation

In order to train a relation classifier, we assemble templates and seed entity pairs to generate labeled pseudo sentences.

For positive sentences, we just assemble each entity pair $(s,t) \in P_r$ with each template $\tau \in T_r$ to generate a sentence $\tau(s,t)$.

While for the negative sentences, following(Vylomova et al., 2016), we have three strategy for each pair $(s_i,t_i) \in P_r$. First, we exchange s and t as (t_i,s_i)(suppose r is not symmetrical). Second, we change one entity to another entity in the same relation :(s_i,t_j) or $(s_j,t_i)(i \neq j, (s_i,t_i),(s_j,t_j) \in P_r)$. Third, we change one entity to an entity in other relations:(s_i,t_j)or $(s_j,t_i)(i \neq j,(s_j,t_j) \in P_{r'})$.

2.5 Relation Classifier

Under the minimally-supervised setting, the model should have good generalizability. We use surface-agnostic features based on attention weights of BERT to make model focus more on the relations rather than the surface information of training data.

As Clark et al. (2019) has pointed out, some heads of multi-head attention in BERT are related to certain relations, and attention weights of certain heads can be used to extract certain relation knowledge. Thus, for a proper template of relation r, the attention weights between s and t of certain heads related to r should be higher. But it is hard to specify each head is related to what relations. Thus we use attention weights of all heads as features to induce relation knowledge.

Specifically, for a sentence $\tau(s,t)$, we calculate the attention weights between s and t of all heads as $\omega_{i,j,s\to t}$, where i denotes the i-th layer, j denotes the j-th head in layer i and $s \to t$ denotes that this is the attention s pays to t. We use the average between $s \to t$ and $t \to s$ to express the attention between them:

$$\omega_{i,j} = \frac{\omega_{i,j,s\to t} + \omega_{i,j,t\to s}}{2} \tag{2}$$

Then we construct attention weights embedding for the sentence $\tau(s,t)$:

$$H^{att}_{\tau(s,t)} = \{\omega_{1,1}, \omega_{1,2}...\omega_{i,j}...\omega_{nl,nh}\} \tag{3}$$

where nl denotes the layer number, nh denotes head number in a layer of BERT.

Besides $H^{att}_{\tau(s,t)}$, we also use BERT outputs to represent the sentence $\tau(s,t)$. Specifically, we input $\tau(s,t)$ into the BERT, and then use the output vector of the [CLS] token as the feature $H^{cls}_{\tau(s,t)}$. $H^{cls}_{\tau(s,t)}$ and $H^{att}_{\tau(s,t)}$ can compensate each other, since $H^{cls}_{\tau(s,t)}$ can capture the information whether $\tau(s,t)$ is "natural", and $H^{att}_{\tau(s,t)}$ contains the correlation between (s,t) and the relation r. Thus, we combine these two vectors through concatenation:

$$H_{\tau(s,t)} = H^{cls}_{\tau(s,t)} \oplus H^{att}_{\tau(s,t)} \tag{4}$$

Then, we feed $H_{\tau(s,t)}$ to a MLP classifier $\mathcal{F}$ and get the probability of (s,t) having relation r. We use a cross-entropy loss to optimize $\mathcal{F}$. In addition, we can also finetune BERT when training the classifier.

2.6 Iterative Template Selection

BERT can rank templates by measuring whether a sentence is natural. However, it can not capture the different attribution of each token in a sentence for expressing the relation.

Integrated Gradient is an attribution method proposed in (Sundararajan et al., 2017). As Cui et al. (2020) has described, the attribution score directly reflects how much changing tokens will change the model's outputs. A higher attribution score represents more importance of tokens. In our relation induction model, s and t obviously should be the most important two tokens in sentences. Intuitively, for a pseudo sentence $\tau(s,t)$, if the integrated gradient value for s and t to the relation prediction is higher, we are more confident that the relational knowledge of (s,t) can be extracted well by the model along with τ, so the template τ is much better. Thus, we can use the integrated gradient of (s,t) to the output of relation classifier to select templates once again. Here, $\mathcal{F}(\tau,s,t)$ denotes the relation classifier with $\tau(s,t)$ as the input.

According to Sundararajan et al. (2017), the integrated gradient value of s to $\mathcal{F}(\tau,s,t)$ is:

$$IG(\tau,s) = (s - s_0)\int_{x=0}^{1} \frac{\partial \mathcal{F}(\tau, s_0 + \alpha(s-s_0), t)}{\partial s} \mathrm{d}\alpha \tag{5}$$

where $\alpha \in [0,1]$, and it can be approximated as:.

$$IG(\tau,s) = (s - s_0)\sum_{i=1}^{m} \frac{1}{m} \times \frac{\partial \mathcal{F}(\tau, s_0 + \frac{i}{m}(s-s_0), t)}{\partial s} \tag{6}$$

where m is the number of approximate steps for computing integrate gradient and s_0 is generated by replacing the word embedding of s with zeros. For a template τ, we calculate the average integrated gradient value for all $(s,t) \in P_r$:

$$score_{IG}(\tau) = \sum_{(s,t)\in P_r} \frac{IG(\tau,s) + IG(\tau,t)}{2} \quad (7)$$

Then the templates are re-ranked according to the final score:

$$score = \alpha \cdot \frac{1}{rank_{BERT}} + (1-\alpha)\frac{1}{rank_{IG}} \quad (8)$$

where $rank_{BERT}$ denotes the rank of templates according to $score_{BERT}$, $rank_{IG}$ denotes the rank of templates according to $score_{IG}$, and $\alpha \in [0,1]$ is an coefficient to balance the two scores. Therefore, the templates could be selected iteratively for better relation induction.

2.7 Relation Induction

Given a new entity pair (x,y), we fill them into templates $\tau_i, (i \in K)$ and use the classifier to predict $p_i(x,y)$, which denotes how much $\tau_i(x,y)$ is "natural". Following Bouraoui et al. (2020), for all predictions from K templates $p_1(x,y),...,p_K(x,y)$, if $max_i p_i(x,y) > 1 - min_i p_i(x,y)$, then (x,y) is predicted to be positive.

3 Experiment Setup

3.1 Datasets

We conduct the experiments on three standard benchmark datasets in English: Google Analogy Test Set (GATS) (Mikolov et al., 2013a), Bigger Analogy Test Set (BATS) (Gladkova et al., 2016) and DiffVec(Vylomova et al., 2016).

GATS contains 5 semantic relations and 9 syntactic relations, and each consists of a varying number of entity pairs. While **BATS** contains 40 relations which are divided into 20 morphology relations and 20 semantic relations, each relation has 50 instances. **DiffVec** contains 36 relations with a various number of entity pairs. 10 of them are lexical or morphology relations and the remaining 26 are semantic relations.

3.2 Implementation Details

The relation induction task can be modeled as a binary classification problem for each relation.We first split the dataset into 50% of training data and 50% of test data. Then, under the minimally-supervised setting, for each relation r, we randomly select N entity pairs from training data as the seed entity pairs P_r. We extract candidate templates from the English Wikipedia corpus[1] and $d = 15$. When generating K templates in T iterations, we initially select $K(T+1)$ templates according to BERT score, and then iteratively filter out K improper templates in each iteration according to the score defined in Formula 8 until K templates are reserved at last for the final iteration. Notice that when $T = 0$, we only select K templates according to the $score_{BERT}$ without considering the $score_{IG}$. In our experiments, we use BERT-base[2], and set $N = 5, K = 20, T = 3, \alpha = 0.5$ by default.

We generate the same number of negative examples as positive examples for the training data and 3 times as many negative examples as positive examples for the test data.

For each relation, we repeat the experiments for 10 times and calculate the average result. The seed entity pairs used in each trial is randomly selected. The results of all metrics are calculated with micro-average.

4 Baselines

We compare our approach with three kinds of baselines.

The first kind is using the combination of pre-trained word embeddings to present relations. Specifically, following Vu and Shwartz (2018), we use $s \oplus t \oplus (s \odot t)$ to represent the relation between (s,t) and use a MLP classifier to make predictions. Here, the pre-trained word embeddings we used are Glove(Pennington et al., 2014)[3] and Skip-Gram(Mikolov et al., 2013b)[4]. These two baselines are referred to as MLP$_{sg}$ and MLP$_{gl}$ respectively.We also use the Trans approach (Bouraoui et al., 2018) for relation induction by building subspaces for entities using word embeddings and modeling the relations with relative positions between subspaces.

The second kind is distant supervised methods. We use PCNN(Zeng et al., 2015) and PCNN-BagAtt(Ye and Ling, 2019) as two baselines. These distant supervised methods are proposed to solve the problem of noise in labeled data in relation extraction tasks. We also select the same number of

[1] We used the dump of May 2021

[2] We used the BERT implementation available at https://github.com/huggingface/transformers

[3] https://nlp.stanford.edu/projects/glove/

[4] https://code.google.com/archive/p/word2vec/

	N=3									N=5								
	GATS			BATS			DiffVec			GATS			BATS			DiffVec		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
MLPsg	41	54.4	39.6	40.3	45.8	40.6	40.1	49.8	41.5	43.3	56.8	45.3	43.5	47.1	43.8	43.5	51.3	43.9
MLPgl	42.5	54.8	43.1	41.2	45.7	51.3	40.5	50.2	41.9	43.9	56.5	45.9	43.8	47.0	43.9	43.8	51.6	44.2
PCNN	58.6	52.5	56.1	52.1	45.8	47.7	57.3	51.5	53.9	60.1	56.2	58.3	53.4	45.8	50.3	59.0	52.7	55.4
PCNN_BagATT	63.9	56.2	59.5	56.4	45.9	48.6	61.5	52.8	55.8	65.3	58.6	60.1	57.8	51.0	50.8	63.5	53.4	57.1
BERT_predict	34.1	48.5	39.6	32.5	45.3	36.1	34.5	46.8	38.5	35.0	48.9	40.1	33.1	45.5	36.8	35.2	46.7	38.9
Trans	35.5.	41.3	37.2	36.8	42.5	39.2	36.7	43.9	39.4	45.8	56.2	50.3	48.5	52.1	49.3	46.6	51.6	48.3
AutoPrompt	75.3	77.9	72.5	65.9	52.6	51.5	72.6	60.3	63.8	78.6	78.1	76.4	67.3	58.5	53.2	75.3	62.5	66.0
RI-BERT	79.3	80.5	75.8	70.1	53.0	53.2	77.4	65.8	68.3	80.7	80.1	79.5	70.1	55.7	55.9	79.5	67.2	70.4
IST	**84.2**	**82.9**	**80.1**	**72.5**	**54.8**	**58.9**	**81.3**	**67.8**	**71.0**	**85.3**	**84.2**	**82.6**	**73.8**	**58.5**	**60.8**	**82.5**	**70.1**	**73.2**

Table 1: Performance on three benchmarks when $N = 3$ and $N = 5$.

sentences that mention entity pairs from the English Wikipedia corpus to construct training data as in our approach. For an entity pair (s, t), K sentences are used to predict its relation. If the average prediction score $\overline{S}_K \geq \theta$, (s, t) is predicted to have relation r. θ is a threshold and set to 0.7 in our experiments for its best performance.

The third kind is using the relational knowledge from PLMs, such as RI-BERT (Bouraoui et al., 2020), AutoPrompt(Shin et al., 2020)[5] and BERT$_{predict}$.

RI-BERT induces relational knowledge from BERT, and our approach would degenerate to it when not using attention maps as the surface-agnostic features and not using $score_{IG}$ to refine the templates. We implement the method by ourselves since there is no open source.

AutoPrompt tries to elicit knowledge from PLM using automatically-constructed prompts. Here, We generate templates with AutoPrompt for each relation. Since there is only one template can be generated for each relation, we use a threshold-based method to determine whether a new entity pair (x, y) has a relation. When $p(x, y) > \delta$, the prediction would be positive. Here, δ = 0.8 is the best threshold in our experiments.

BERT$_{predict}$ is a simple baseline proposed by ourselves. After K templates are selected with $score_{BERT}$, we directly use BERT mask-prediction task to judge relation. Specifically, for an entity pair (s, t) and a template τ, if BERT can predict $\tau(s, \_)$ or $\tau(\_, t)$, the score of (s, t) will be increased by 1. The max score is $2K$, so if the score of $(s, t) \geq \epsilon \cdot 2K$, (s, t) is predicted to have the relation r. ϵ is a threshold and set to 0.7 for its best performance.

5 Experimental Results

5.1 Main Results

The main experimental results on the three aforementioned benchmarks are shown in Table 1, which reports the micro-average of precision, recall and F1 of our approach **IST** and other state-of-the-art methods when $N = 3$ and $N = 5$.

From the table, there are several observations drawn from different aspects. (1) Our approach **IST** achieves the best performance against all other kinds of methods. (2) Pre-trained word embedding-based approaches such as MLP$_{sg}$ and MLP$_{gl}$ performance poorly, which proves that only few labeled entity pairs will degrade these approaches greatly. And Translation does not turn out well because of the lack of entities to construct representative subspaces. (3) The relational knowledge directly drawn from BERT also contains much noise according to the results of BERT$_{predict}$. (4) Traditional distant-supervised approaches which don't resort to PLM suffer from the noisy and sparse bag issues, although PCNN-BagATT uses intra-bag and inter-bag attention to handle sentence and bag-level noise, and get better performance, they are still not suitable for the minimally-supervised relation induction task. (5) AutoPrompt and RI-BERT use proper prompts or templates from BERT, so they can obtain a better performance. However, they did not consider the generalization problem in the minimally-supervised setting. In addition, they ignored the contribution of each token in a sentence for expressing the relation, especially for the entity pairs, but only considered whether a sentence is natural or not according to BERT. (6) More labeled entity pairs can achieve better performance by comparing the results of N = 3 and N = 5. This

[5] https://github.com/ucinlp/autoprompt

phenomenon is reflected by all methods in both three datasets.

5.2 Ablation Study and Analysis

Performance of Different Relations To further explore the performance of different relations, we show the detailed results of each relation in GATS in Table 2.

From the table, we can see that our approach achieves better performance for both semantic and morphology relations. Moreover, the iteratively template selection can bring a significant improvement, especially for semantic relations. As to morphological relations, the improvement is not so evident. This is because the entities in morphological relations are always adverbs or adjectives to which little attention is paid, so $H^{att}_{\tau(s,t)}$ plays a limited role.

	GATS	RI-BERT	T=0	T=1	T=2
Semantic	currency	56.7	58.8	58.6	**59.5**
	family	76.9	78.8	78.4	**79.9**
	capital-common	88.4	87.3	85.7	**91.6**
	city-in-state	68.2	71.0	73.1	**75.2**
	capital-world	77.3	76.8	78.0	**78.2**
	Average	73.5	74.5	74.7	**76.9**
Morphology	adj-to-adv	39.1	38.8	42.3	**44.8**
	opposite	55.3	**59.7**	54.0	56.6
	comparative	**90.9**	87.5	88.2	89.0
	superlative	78.1	79.8	**80.6**	77.7
	presen-participle	98.4	96.2	98.1	**98.9**
	nationality-adj	91.5	**92.4**	91.7	92.1
	past-tense	96.9	**97.8**	97.2	97.0
	plural	93.8	91.6	**96.6**	95.8
	plural-verb	**100**	99.0	99.7	99.7
	Average	82.6	82.6	83.2	**83.5**

Table 2: Detailed experimental results (F1) for each relation on GATS. T denotes the iteration number

Performance of attention weights and IG To investigate the effectiveness of BERT attention weights and IG, we compare the performance of several variants of our approach on GATS.

To reduce the effect of BERT attention weights, the representation of sentence $\tau(s,t)$ is simplified from $H^{cls}_{\tau(s,t)} \oplus H^{att}_{\tau(s,t)}$ to $H^{cls}_{\tau(s,t)}$. In addition, without IG, there would be no iterative template selection procedure. The results are shown in Table 3, and the performance drops in all variants, which proves that both attention maps and integrated gradient are useful in our approach.

Different Number of Templates To analyze the impacts of the number of templates (K), we conduct experiments with different numbers of templates, and the results are shown in Table 4. From the table, we find that more templates can bring better performance in all iterations. However, if K is too large, the time consumption will be greater and some unsuitable templates will be retained, leading to worse results.

GATS	T=0	T=1	T=2	T=3
IST	79.7	80.2	81.1	82.6
w/o att	79.5	79.7	80.6	80.9
w/o IG	78.4	78.4	78.4	78.4

Table 3: The F1 scores of IST and other variants on GATS with different iterations.

	K=5	K=10	K=20
T=0	72.5	75.3	79.7
T=1	73.8	78.5	80.2
T=2	75.6	78.9	81.1

Table 4: F1-score with different number of templates ($K = 5, 10, 20$) and different iterations ($T = 0, 1, 2$) on GATS.

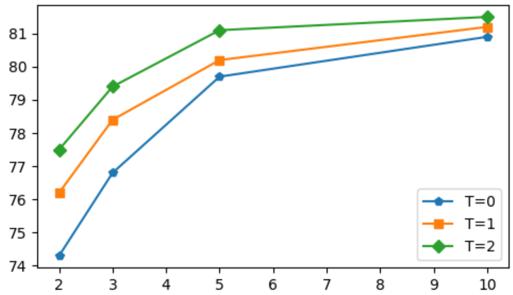

Figure 2: F1 scores with different numbers of seed entity pairs(N = {2, 3, 5, 10}) of our approach on GATS

Different Number of Seeds We evaluate our approach with different numbers of seed entity pairs (N), and the results are shown in Figure 2. From the figure, we can see that F1 score increases gradually until convergence for all iterations. Our approach already achieves a satisfactory result when N = 5.

Effect of Balance Coefficient The parameter $\alpha \in [0, 1]$ is a balance coefficient between $score_{BERT}$ and $score_{IG}$ for template scoring. Larger α will consider $score_{IG}$ more in the scoring. We conduct the experiments with different α on GATS, and the results are shown in Figure 3. From the figure, we find that our approach achieves the best performance when α = 0.5.

T=0	T=3
The Government of _ denoted 300 million _ to finance the school's construction in 1975.	The _ (, plural: / ,) is the currency of _ .
Currently, _ uses the _ as its national currency.	This was one of the reasons for naming the current currency of the Republic of _ the _.
Following the introduction of the euro, the _ was linked to the euro, until January 1, 2015, when _ officially adopted the euro as its currency.	The _ (; ; sign: ; code: KHR) is the currency of _.
AutoPrompt: _ cial largest greenwich _.	

Table 5: Case study for relation *currency*, where top 3 templates are exhibited with different approaches.

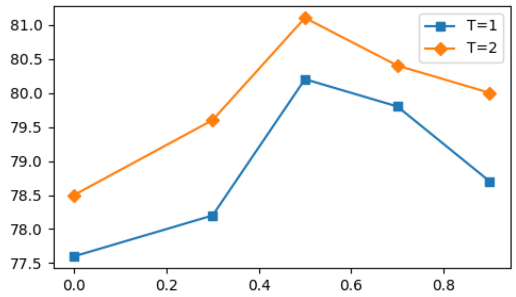

Figure 3: F1 scores with α={0, 0.3, 0.5, 0.7, 0.9]} of our approach on GATS

Case Study Table 5 compares the selected templates of relation currency between $T = 0$ and $T = 3$. From human's intuition, we find that comparing to (T = 3), the templates filtered out only with $score_{BERT}$(T = 0) are more ambiguous that they might indicate a co-occurrence relationship rather than relation *currency*. For example, for the first template "The Government of _ denoted 300 million _ to finance the school's construction in 1975.", it is natural for the government of a country to denote their own currency or just use dollar to evaluate how much they have denoted. So $\tau(s, dollar)$ is natural when s denotes any country. This is due to the way of selecting templates that only requires the templates is proper for all $(s, t) \in P_r$ without explicitly declaring what the relation is. In fact, the model can distinguish $co-occurrence$ and *currency* only after the BERT is fine-tuned with negative examples. As to the template generated by AutoPrompt, it is a combination of some tokens rather than a human-readable sentence. Although AutoPrompt got good results on some tasks(Shin et al., 2020), the template is totally not interpretable from human's perspective.

6 Related Work

6.1 Relation Induction

Relation induction was first proposed in (Vylomova et al., 2016). They used the vector difference between two entities to represent the relation between them. More researches on the relation induction with word embeddings were proposed in(Drozd et al., 2016; Bouraoui et al., 2018; Vu and Shwartz, 2018). They pointed out that the difference is not the best way to express the relationship and proposed more complicated methods to better extract relational knowledge between word embeddings.

6.2 Knowledge Induction from BERT

BERT was proven to be able to capture relational knowledge(Kim et al., 2020; Bouraoui et al., 2020; Ushio et al., 2021; Chen et al., 2021). Inspired by this, some works tried to use BERT on the relation induction task (Shin et al., 2020; Bouraoui et al., 2020; Jiang et al., 2020). The key point of these methods is to fill entities in the proper templates.

Recently, many efforts focus on the generation of templates. Jiang et al. (2020) proposed a template generation strategy based on paraphrasing aiming to improve lexical diversity while remaining relatively faithful to the original prompt. Shin et al. (2020) proposed AutoPrompt method to generate templates, or as they called, prompts, from nothing instead of from corpus. They automated create prompts based on gradient-guided search.

7 Conclusion

In this paper, we propose a novel minimally-supervised relation induction approach. Our proposed approach can iteratively select proper templates using $score_{IG}$ and $socre_{BERT}$, and obtain a good generalized ability with surface-agnoistic features based on attention maps of BERT. Experiments illustrate that our approach achieves state-of-the-art performance on three standard benchmarks.

Acknowledgements

This work is supported by the Key Research and Development Program of Zhejiang Province, China (No. 2021C01013), the National Key Research and Development Project of China (No. 2018AAA0101900), CKCEST, and MOE Engineering Research Center of Digital Library.

References

Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the Fifth ACM Conference on Digital Libraries*, DL '00, page 85–94, New York, NY, USA. Association for Computing Machinery.

Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1388–1398, Florence, Italy. Association for Computational Linguistics.

Zied Bouraoui, Jose Camacho-Collados, and Steven Schockaert. 2020. Inducing relational knowledge from bert. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7456–7463.

Zied Bouraoui, Shoaib Jameel, and Steven Schockaert. 2018. Relation induction in word embeddings revisited. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1627–1637, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jose Camacho-Collados, Luis Espinosa Anke, and Steven Schockaert. 2019. Relational word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3286–3296, Florence, Italy. Association for Computational Linguistics.

Catherine Chen, Kevin Lin, and Dan Klein. 2021. Constructing taxonomies from pretrained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4687–4700, Online. Association for Computational Linguistics.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Leyang Cui, Sijie Cheng, Yu Wu, and Yue Zhang. 2020. Does bert solve commonsense task via commonsense knowledge? *ArXiv*, abs/2008.03945.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuoka. 2016. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3519–3530, Osaka, Japan. The COLING 2016 Organizing Committee.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. 2020. Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1295–1309, Online. Association for Computational Linguistics.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pages 8–15, San Diego, California. Association for Computational Linguistics.

Xiaotao Gu, Zihan Wang, Zhenyu Bi, Yu Meng, Liyuan Liu, Jiawei Han, and Jingbo Shang. 2021. Ucphrase: Unsupervised context-aware quality phrase tagging. *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Taeuk Kim, Jihun Choi, Daniel Edmiston, and Sang goo Lee. 2020. Are pre-trained language models aware of phrases? simple but strong baselines for grammar induction. *ArXiv*, abs/2002.00737.

Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. 2019. Revealing the dark secrets of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4365–4374, Hong Kong, China. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *NeurIPS*.

Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *ICLR*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Deepak Ravichandran and Eduard Hovy. 2002. Learning surface text patterns for a question answering system. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 41–47, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4222–4235, Online. Association for Computational Linguistics.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. *ArXiv*, abs/1703.01365.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. 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 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Asahi Ushio, Luis Espinosa Anke, Steven Schockaert, and Jose Camacho-Collados. 2021. BERT is to NLP what AlexNet is to CV: Can pre-trained language models identify analogies? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3609–3624, Online. Association for Computational Linguistics.

Tu Vu and Vered Shwartz. 2018. Integrating multiplicative features into supervised distributional methods for lexical entailment. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 160–166, New Orleans, Louisiana. Association for Computational Linguistics.

Ivan Vulić and Nikola Mrkšić. 2018. Specialising word vectors for lexical entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1134–1145, New Orleans, Louisiana. Association for Computational Linguistics.

Ekaterina Vylomova, Laura Rimell, Trevor Cohn, and Timothy Baldwin. 2016. Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1671–1682, Berlin, Germany. Association for Computational Linguistics.

Yile Wang, Leyang Cui, and Yue Zhang. 2020. Does chinese bert encode word structure? In *COLING*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*.

Zhi-Xiu Ye and Zhen-Hua Ling. 2019. Distant supervision relation extraction with intra-bag and inter-bag attentions. 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

2810–2819, Minneapolis, Minnesota. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal. Association for Computational Linguistics.