electronics

MDPI

Article

Knowledge-Guided Prompt Learning for Few-Shot Text Classification

Liangguo Wang *, Ruoyu Chen and Li Li

School of Computer Science, Beijing Information Science & Technology University, Beijing 100101, China
* Correspondence:lgwang@bistu.edu.cn

Abstract: Recently, prompt-based learning has shown impressive performance on various natural language processing tasks in few-shot scenarios. The previous study of knowledge probing showed that the success of prompt learning contributes to the implicit knowledge stored in pre-trained language models. However, how this implicit knowledge helps solve downstream tasks remains unclear. In this work, we propose a knowledge-guided prompt learning method that can reveal relevant knowledge for text classification. Specifically, a knowledge prompting template and two multi-task frameworks were designed, respectively. The experiments demonstrated the superiority of combining knowledge and prompt learning in few-shot text classification.

Keywords: knowledge-guided; prompt learning; multi-task learning; text classification

1. Introduction

Text classification is one of the basic tasks in natural language processing (NLP). Many applications benefit from text classification, such as information extraction [1], task-oriented dialogue systems [2], and so on. Figure 1 shows an example of Chinese news headline classification. In this example, text-represented news headlines are classified into several pre-defined category labels.

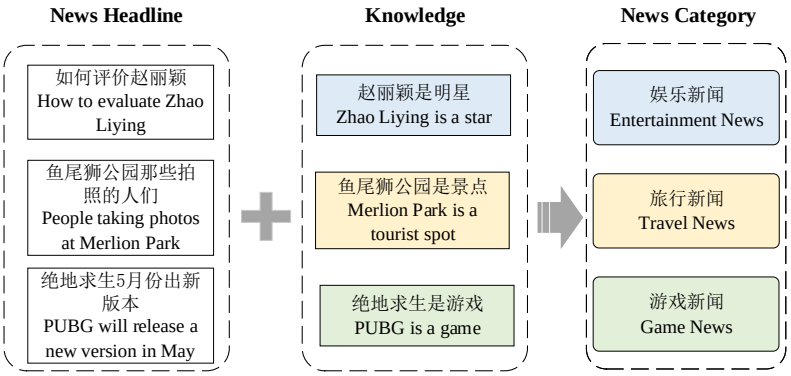


Figure 1. News headline classification samples. In these samples, news headlines are associated with certain conceptual knowledge, which has a positive impact on the text classification.

In the real world, text classification faces two challenges: the lack of labeled data and the emergence of new categories. Few-shot and zero-shot text classification are proposed to solve these two problems, respectively. Few-shot text classification aims to predict the labels of unknown texts with only a handful of samples, and zero-shot text classification aims to classify instances belonging to the classes that have no labeled instances. Both of them have become fast-developing fields in machine learning and have a wide range of applications in artificial intelligence.

check for updates

Citation: Wang, L.; Chen, R.; Li, L. Knowledge-Guided Prompt Learning for Few-Shot Text Classification. *Electronics* **2023**, *12*, 1486. https://doi.org/10.3390/electronics12061486

Academic Editors: Shangsong Liang and Zaiqiao Meng

Received: 22 February 2023
Revised: 19 March 2023
Accepted: 20 March 2023
Published: 21 March 2023

Existing text classification methods, in particular short text classification methods, always expand the feature space by utilizing an external open knowledge base [3,4]. However, most existing methods based on external knowledge rely on large-scale training instance data to formalize the model, which results in high costs for collecting eligible training data and poor performance in few-shot learning.

Recently, pre-trained language models (PLMs) have received increasing attention and shown remarkable improvements in a variety of downstream NLP tasks. Previous research has revealed that the parameters of PLMs appear to store vast amounts of linguistic and factual knowledge [5–7]. Current methods to adapt the versatile knowledge contained in PLMs to NLP tasks can roughly be categorized into two classes: fine-tuning and prompt learning. Fine-tuning methods [8,9] with additional classifiers have been widely used, but they are not suitable for the few-shot scenario because they rely heavily on task-specific labeled data to learn additional parameters. By formalizing the downstream task as a cloze-style prediction or text-generation task, prompt learning [10–12] eliminates the need for new parameters and is more appropriate for few-shot scenarios. For example, to classify the topic of a headline such as "how to evaluate Zhao Liying" into the "entertainment" category, a prompting function can be denoted as "x is a [mask] news", and prompt learning predicts the probability that the word "entertainment" is filled in the "[mask]".

Prompt learning has yielded significant results for few-shot learning through the implicit application of PLM knowledge. However, knowledge works as a black box in prompt learning, and it is unclear how this knowledge contributes to downstream NLP tasks. In this work, we aimed to analyze the relationship between knowledge stored in PLMs and text classification decisions in a few-shot setting by making explicit use of the knowledge. To this end, two approaches are proposed to utilize knowledge. The first approach is encoding knowledge into the prompting templates to infer the answer directly. We hope that the knowledge will guide PLMs to output correct text labels. Given that learning multiple related tasks can exploit task-generic and task-specific information simultaneously, multi-task learning can naturally improve the performance of few-shot text classification [13]. Therefore, the second approach is a multi-tasking learning model that combines knowledge exploration and text classification. In this method, the corresponding knowledge related to text classification decisions will be exported simultaneously with text labels, improving the interpretability of prompt-based models. Specifically, two kinds of multi-task models are established in the second method for cloze-style prompt-based text classification: (1) We used a dependent prompting template to predict the knowledge label and text category in sequence, so that the two tasks affect each other during training and prediction. (2) We built two independent prompting templates for these two tasks by sharing the text content. By this means, these two tasks have less association than the dependent model.

Overall, the contributions of this paper can be summarized as follows:

1. We made use of the conceptual knowledge explicitly by knowledge-prompting templates for few-shot text classification, significantly improving the performance of the original model.

2. We established two models in a multi-task prompting framework to explore the relation between knowledge probing and text classification. Comparing with the base model, the proposed method in our work outperforms it in most cases while retaining good interpretability for text classification.

We evaluated our methods on several publicly available text classification datasets in both few-shot and zero-shot settings. The experimental results showed that our knowledge-guided prompting models outperformed the baseline models by the accuracy and F1 score in both settings. We also conducted experiments with different training sizes and different prompting templates, and we found that the performance of the prompting models is associated with these factors. By manually checking the predicted labels of our multi-task models, we discovered that the knowledge probed could serve as an explanation for

both correct and incorrect predicted samples, which shows that our models improved the interpretability of prompt-based models to some extent.

2. Method

In this section, we present our knowledge-guided prompting text classification models.

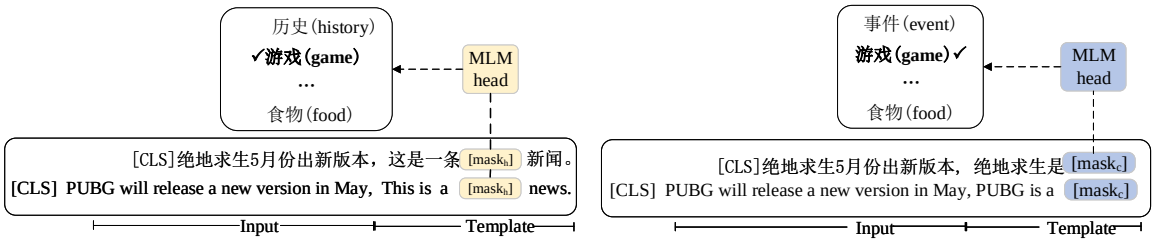
2.1. Problem Definition

As one of the basic NLP tasks, the goal of text classification is to categorize text into pre-defined categories. Let us use $x = (w_1, w_2, \ldots, w_n)$ to denote an input sentence, which consists of a sequence of words. Here, $w_i \in V$ and V is the vocabulary. We would like to classify the text into pre-defined category labels Y.

In the few-shot setting, the size of available annotated data is limited. We assumed that only a few samples of training sentences and corresponding labels can be used. The dataset can denoted as $D = \{(x_j, y_j)\}_{j=1}^{N}$. Our goal was to learn a text classification model from D so that we can predict the label of any unseen sentence y. In the zero-shot setting, we assumed that there are no samples for training. Our goal was to infer the text labels directly.

2.2. Prompt-Based Text Classification

We first present our base model: prompt-based text classification, which is shown in Figure 2. It was motivated by GPT-3 [10], which exploits the implicit knowledge stored in PLMs to predict text labels.



(a) prompt-based text classification (b) prompt-based knowledge probing

Figure 2. The prompt-based text classification and prompt-based knowledge-probing models.

Generally, there are three key steps in prompt-based learning [12]. The first step is to define a prompting function or template $f_p(\cdot)$ that converts the source input x into a prompt $x' = f_p(x)$ and then transforms it into vectors by PLMs. Usually, x' contains a placeholder that could be used to infer the answer by language models. Take the headline classification for example: "PUBG will release a new version in May" is a news headline, and one of the possible prompting templates could be defined as:

$$f_p(x) = [x], \text{this is a } [mask] \text{ news.}$$

where $[x]$ is the input slot for the headline text and $[mask]$ is the placeholder for the position where the masked language model (MLM) is used to predict words. Using this prompting function on our example, x' would become "PUBU will release a new version in May, this a [mask] news". Then, this raw sentence is encoded by the MLM and transformed into vectors.

Assuming the output of the MLM is $O \in R^{|f_p(x)|*d}$, where $|f_p(x)|$ is the sequence length and d is the MLM vector size:

$$O = MLM(f_p(x))$$

The label probability distribution of the [mask] token can be obtained by

$$P_{[mask]} = O_{[mask]}$$

The second step is to search for the words in the position of [mask] by maximizing $P_{[mask]}$. In the case of classification, the words should be semantically related to the category labels. Using Z to denote the potential word set, then we search $\overline{z}$ over Z by maximizing the probability:

$$\overline{z} = argmax_{z \in Z} P_{[mask]}$$

The third and last step is to map the highest-scoring answer words $\overline{z}$ to the predicted category labels Y. A mapping function $M(\cdot)$ should be defined as:

$$y = M(\overline{z})$$

In the simplest case, Z is equal to Y, then $M(\overline{z})$ is equal to $\overline{z}$, which means the word predicted by PLMs is the category label of the headlines. In the zero-shot setting, these three steps are executed step by step after choosing a suitable pre-trained language model (PLM). In the few-shot setting, the cross-entropy loss is calculated as follows:

$$L = -\frac{1}{N} \sum_{i=1}^{N} [l_i = M(\overline{z})] log(P_{i\overline{z}[mask]})$$

where N is the training size, l_i is the ground truth label of sample i, and $P_{i\overline{z}[mask]}$ is the probability of word $\overline{z}$ in sample i in the mask position. The PLM is fine-tuned by gradient descent after calculating L.

2.3. Prompt-Based Knowledge Probing

Previous work conducted an in-depth analysis of the knowledge stored in PLMs and found that PLMs contain much relational knowledge [6,14]. In this work, we followed this conclusion and assumed that PLMs contain much conceptual knowledge without fine-tuning. We considered "is-A" conceptual relational knowledge in this work, as it has proven helpful for short text classification [15]. Take the previous sample as an example: for the headline "PUBU will release a new version in May", we assumed the PLMs contain the conceptual knowledge: "PUBU is a game", then this knowledge is useful when determining the label of this headline to "game". For knowledge probing, previous work treated it as a text-generation task [6], where the search space is the whole vocabulary. When the target concepts lie in a fixed space, let us use C for representation; this kind of knowledge probing could also be defined as a text classification task and solved by prompt learning similarly as in Section 2.2. This method is shown in Figure 2. With the same input as the previous part, assume each headline contains one entity, and let us denote the prompt function for knowledge probing as $f'_p(\cdot)$. One of the possible prompt templates could be defined as

$$f'_p(x) = [x], [e] \text{ is a [mask].}$$

where $[e]$ represents the entity in the input text and $[x]$, [mask] have the same definition as Section 2.2. The rest of the steps are the same as the previous part, and finally, we expect to obtain the concept label c, where $c \in C$.

2.4. Knowledge-Guided Models

In this part, we introduce two methods that utilize the knowledge explicitly. The first method, called p prompt learning text classification model with knowledgeable prompting template (PTCKT), refers to the knowledge being encoded into the prompting template to decide the text labels. The second method, including two models: dependent prompting multi-task model (DPMT) and independent prompting multi-task model (IPMT), refers to integrating the text classification and knowledge probing together in a multi-task framework.

2.4.1. Knowledge-Encoded Template

In Figure 3, we show the basic idea of PTCTK with an example. In the original prompting models, manually created are templates usually based on human introspection [12], and the knowledge related to the source text is ignored. In our proposed PTCTK method, we encode the conceptual labels of entities into the prompting template. In other words, the prompting template is associated with the text knowledge, so it is knowledgeable. Take the previous sample as example: for the headline "PUBU will release a new version in May", the entity "PUBU" is annotated as a game. Then, one of the possible prompting functions of this model is

$$f_p(x)=[x],\ [e] \text{ is a } [c],\ \text{this is a [mask] news.}$$

where c is filled by the concept during data preprocessing. For this sample, the prompt template converts the text into "PUBU will release a new version in May, PUBU is a game, this is a [mask] news". The other parts of this model are consistent with the previous part.

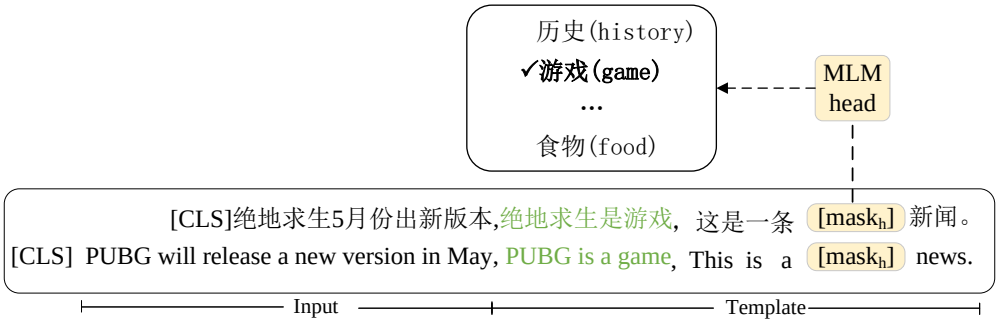


Figure 3. The prompt-based text classification model with knowledge template.

2.4.2. Multi-Task Framework

In this section, we introduce how to integrate text classification and knowledge probing together in a multi-task framework. Intuitively, just like the examples showed in Figure 1, conceptual knowledge is highly related to text labels as it could serve as the reason to decide the text categories. To learn the correlation between knowledge probing and text classification, two kinds of multi-task models were designed: the dependent prompt multi-task model (DPMT) and the independent prompt multi-task model (IPMT).

DPMT: In this model, the knowledge probing and headline classification are dependent and share one prompting template with two $[mask]$ placeholders, one $[mask]$ used to predict the conceptual label and the other used to predict the headline label. Take the previous samples as an example: one of the possible prompting templates could be defined as

$$f_p(x)= [x],\ [e] \text{ is a } [mask_c],\ \text{this is a } [mask_h] \text{ news.}$$

where $[mask_c]$ denotes the placeholder prepared for the conceptual category and $[mask_h]$ represents the placeholder used for headline labels. This model is shown in Figure 4. In this model, knowledge probing and headline classification are processed in sequence, and they share all the parameters.

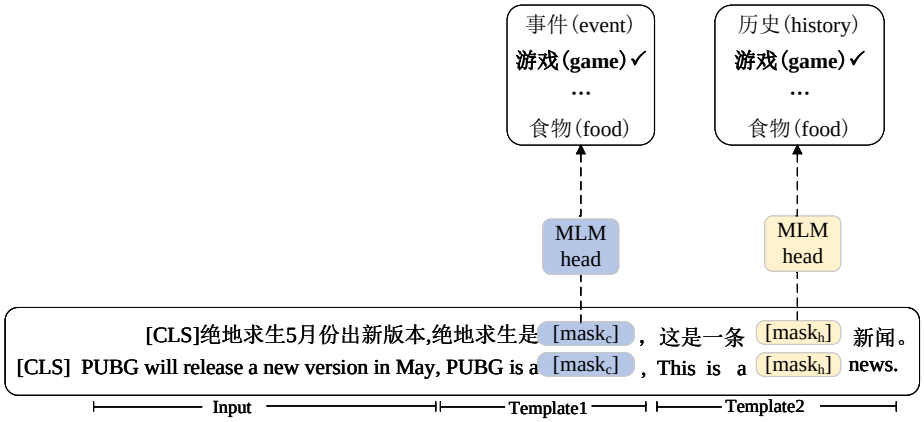
IPMT: In this model, knowledge probing and headline classification are processed independently, and they share the headline text with their own prompting templates. This model is shown in Figure 4. For our example, the prompting templates are defined as

$$f_p\ (x) = [x],\ \text{this is a } [mask_c] \text{ news.}$$
$$f'_p(x) = [x],\ [e] \text{ is a } [mask_h].$$

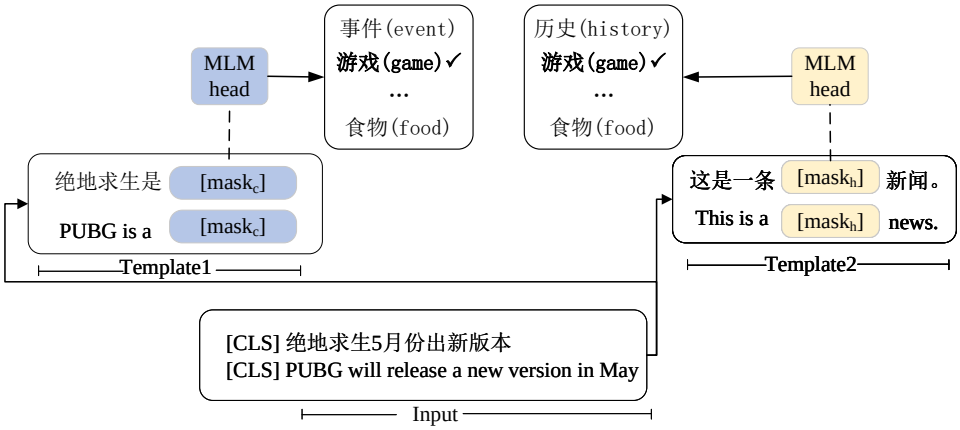
where $[x]$ refers to the shared headline.

The work process of our multi-task models follows the same steps described in Section 2.2. Given a masked pre-trained language model L and a headline x, first x is sent to the prompting template to generate x'. Next, x' is sent to L, and a sequence of hidden vectors $h_k \in R^k$ is obtained. After that, the hidden vector of the mask word is used to predict the target words. Finally, the target words are transformed into category labels via the mapping function. For the zero-shot setting, there is no training process, and the model is used to predict the label directly. For the few-shot setting, as the training data are available, the cross-entropy loss is calculated by the predicted probability distribution and the desired target words. Then, the parameters can be updated by a gradient-descent-based optimization algorithm.

Compared with traditional text classification methods and the base model described in Section 2.2, our multi-task models could improve the interpretability of neural network models. For example, in Figure 1, traditional methods only output the headline labels without any reason why they chose these labels. In other words, they lack interpretability. In our multi-task framework, the related conceptual labels are also output as an explanation of the label decision for the headlines.



(a) DPMT model



(b) MPMT model

Figure 4. The proposed multi-task prompting models. In DPMT, knowledge probing and headline classification are processed in sequence and they share all the parameters. In MPMT, knowledge probing and headline classification are processed indepedently, they share the headline text with their own prompting templates.

3. Experiments

We evaluated our models by conducting experiments on three Chinese text classification datasets. Our models were implemented with the OpenPrompt [16] library https://github.com/thunlp/OpenPrompt (accessed on 1 February 2023)—an open-source framework for prompt learning. We ran all the experiments on a 16 GB Tesla P100 GPU.

3.1. Datasets and Experiment Settings

Here are the datasets we used in our experiments for evaluation:

NLPCC : This is a Chinese news headline dataset provided by the NLPCC shared task http://tcci.ccf.org.cn/conference/2017/taskdata.php (accessed on 1 February 2023). The data are collected from several Chinese news websites such as toutiao, sina, and so on.
THUCNews: This dataset is collected by Tsinghua University http://thuctc.thunlp.org/ (accessed on 1 February 2023); it contains complete news content and corresponding labels. In our experiment, we only used the headlines for classification.
AIStudio: This is another news headline dataset used for the AIStudio data competition https://aistudio.baidu.com/aistudio/datasetdetail/103654/0 (accessed on 1 February 2023); each headline has an associated category label.

We are interested in a multi-task setting where both entity concept labels and headline category labels are available for the training process. However, these three datasets only have headline labels, but lack conceptual labels. Therefore, to obtain knowledge information, we refer to CN-Probase [17] as an external knowledge base to automatically annotate the entities. Specifically, we processed the raw datasets as follows: First, for each sample in the dataset, the headline was sent to an entity-linking API service http://kw.fudan.edu.cn/apis/qa/ (accessed on 1 February 2023) to recognize the entities. Then, we reserved samples that contained one and only one entity. Following that, the entities of the remaining samples were fed into the entity concept API service http://kw.fudan.edu.cn/apis/cnprobase/ (accessed on 1 February 2023) to obtain their conceptual labels. Following this procedure, we created three datasets, each with two labels: one for the entity conceptual label and the other for the headline category label.

In this study, our work mainly focused on the text classification task in few-shot and zero-shot environments, where the number of samples was not very large. The detail of the datasets is shown in Table 1. The headline category number of these three datasets is 7 and the concept number is 14, 17, and 17, respectively. There are 1813, 2354, and 1441 samples in NLPCC, THUNews, and AIStuido. In the few-shot setting, following previous work [18], we primarily used a training set size K = 2 (each category with K samples), but explored K = 1,2,4,8 in the experiments. For each data, to form the training/validation/test data, we sampled 20 samples for each category to make up a subset of the data, and the rest were used for testing. Then, for each K, we sampled 2*K samples of each category from the data subset for training and validation, respectively.

Table 1. Detail of the experimental datasets.

Datasets	Text Class Size	Concept Class Size	Data Size
NLPCC	7	14	1813
THUCNews	7	17	2354
AIStudio	7	17	1441

We evaluated the following models in our experiments:

PTC: This is the prompt-learning-based text classification model we presented in Section 2.2.
PTCKT: This is our prompt-learning-based text classification model with the knowledge prompting template introduced in Section 2.4.

DPMT: This is our dependent prompting multi-task model described in Section 2.4.2. In this model, the conceptual knowledge and text label are encoded in one prompting template; thus, their mask slots will be predicted sequentially.
IPMT: This is our independent prompting multi-task model presented in Section 2.4.2. In this model, the conceptual knowledge and headline label are encoded independently with their own prompting template by sharing the headline text.

The prompting templates for these models were the same as described previously in each section. For the answer space, they were simply set as the conceptual label words and headline category label words. We conducted experiments in the few-shot and zero-shot settings. In the few-shot setting, our model was trained using the Adam algorithm [19] with a learning rate initialized at 0.001. The batch size was set to 64. We ran all the experiments 4 times with different data seeds and report the average performance to reduce variance.

We also compared our methods with a few prompt learning and fine-tuning baselines:

KPT : This is a knowledgeable prompt-tuning method that incorporates knowledge into Prompt Verbalizer for Text Classification [20]. This expands the label word space using external knowledge bases.
WARP: Instead of manually designed discrete prompting functions, this method utilizes trainable parameters as prompting templates [21]. This kind of prompting is also called continuous prompting [12].
BERT: This is the BERT [22] model for classification. A headline was encoded by BERT, and then, the vector of the first token was used for classification.
BERT-CNN: This model builds a convolutional neural network (CNN) on top of BERT. It predicts the label after the CNN layer.
BERT-RNN: Instead of a CNN, a recurrent neural network (RNN) was used to encode the vector on top of BERT.
BERT-RCNN: Instead of a CNN and an RNN, the combination of a CNN and an RNN was used to encode the vector on top of BERT. After the BERT encoding, the vectors were processed by an RNN and then sent to a CNN before classifying.

For the pre-trained language model, we used bert-base-chinese https://huggingface.co/BERT-base-chinese (accessed on 1 February 2023) for our models, as well as these baselines. With the evaluation, we report the accuracy and F1 scores on the test data.

3.2. Experimental Results

The experimental results for the few-shot (K=2) and zero-shot settings are shown in Tables 2 and 3, respectively. For the supervised baselines, we report the performance of these models without training in the zero-shot setting. From the table, we have the following observations:

(1) In all of our methods, there is no doubt that PTCKT had the best performance in all the experiments, which means the related knowledge is helpful for the model to classify text. It outperformed the base model by 20% to 40% across the datasets for the few-shot setting and 7% to 12% for the zero-shot setting. For our multi-task models, the DPMT model beat the PTC model in both the few-shot and zero-shot settings; the performance improved upon PTC by 8.9% in NLPCC, 2.7% in THUNews, 11.5% in AIStudio for the few-shot setting and 2.4%, 1.7%, and 0.7% for the few-shot samples in the F1 value. For IDPT, in the zero-shot setting, it obtained identical scores as PTC since there was no difference in their models considering the headline part. In a few-shot setting, it performed better than the PTC model in NLPCC while worse in THUNews and AIStuido. These results suggest that the explicit use of knowledge could improve the accuracy and F1 score for prompting-based text classification. Considering the different performance of our two multi-task models, when classifying text and probing knowledge in a multi-task framework, the selection of multi-task models plays a key role in the performance.

Table 2. Average (Avg) and standard deviation scores (Std) of the accuracy and F1 for all the methods in the few-shot setting (K = 2) on different datasets. Bold stands for the best.

	NLPCC		**THUNews**		**AIStudio**	
	ACC (%) Avg/Std	**F1 (%) Avg/Std**	**ACC (%) Avg/Std**	**F1 (%) Avg/Std**	**ACC (%) Avg/Std**	**F1 (%) Avg/Std**
PTC	49.9/9.1	48.2/9.8	54.8/8.7	53.5/8.5	46.0/**5.7**	41.7/**7.4**
PTCKT	**79.0**/9.3	**78.2**/9.6	**77.9**/**3.7**	**75.5**/**4.1**	**86.4**/9.7	**81.3**/10.3
DPMT	58.9/7.3	57.1/7.7	60.9/13.8	56.2/14.7	59.3/8.2	53.2/10.8
IPMT	54.5/**6.1**	54.0/6.7	50.6/6.7	43.3/10.7	43.2/11.4	38.8/10.4
KTP	68.8/6.5	67.6/6.8	73.5/4.6	72.9/8.7	79.7/8.6	77.3/10.5
WARP	67.6/8.1	65.8/8.9	69.5/7.6	66.8/7.5	74.9/10.2	72.7/9.5
BERT	42.7/25.0	34.1/29.1	58.4/32.1	51.7/32.9	58.4/32.1	51.7/32.9
BERT-CNN	60.2/8.8	56.0/12.0	73.8/8.5	72.7/9.5	73.8/8.5	72.7/9.5
BERT-RNN	60.4/8.3	56.2/7.2	75.2/11.6	73.0/14.9	75.2/11.6	73.0/14.9
BERT-RCNN	58.0/7.7	56.4/**4.6**	70.79/18.7	67.5/24.3	70.8/18.7	67.5/24.3

Table 3. Accuracy and F1 scores for all the methods in the zero-shot setting on different datasets. Bold stands for the best.

	NLPCC		**THUNews**		**AIStudio**	
	ACC (%)	**F1 (%)**	**ACC (%)**	**F1 (%)**	**ACC (%)**	**F1 (%)**
PTC	17.6	14.6	11.4	9.2	13.9	9.3
PTCKT	**25.6**	**26.4**	18.5	16.1	24.9	**17.5**
DPMT	19.7	17.0	12.8	10.9	14.3	10.0
IPMT	17.6	14.6	11.4	9.2	13.9	9.3
KTP	18.3	17.5	**19.5**	**17.3**	18.4	16.3
WARP	8.4	6.7	7.5	5.4	9.5	7.2
BERT	17.7	4.4	15.1	3.7	**30.4**	7.8
BERT-CNN	12.5	3.6	11.2	3.4	5.0	2.2
BERT-RNN	12.8	6.1	7.1	2.9	17.0	12.1
BERT-RCNN	14.5	5.5	8.4	4.4	23.3	10.3

(2) Comparing our methods with the prompting and fine-tuning baselines, for the zero-shot setting, in most cases, our PTCKT model performed the best. KTP obtained the best accuracy and F1 score on THUNews and had relatively high performance on the other two dataset since it utilizes knowledge from an external knowledge base. WARP had relatively low performance mainly because the randomly initialized template did not adapted to our task. In most cases, all the prompt-based methods did much better than the fine-tuning methods in the accuracy and F1 value. This result indicates that the prompt-based models could exploit PLMs more efficiently without training data. However, it is worth noting that the fine-tuned BERT model had the highest accuracy score of 30% on the AIStudio data. After checking the result, we found that the models predicted the same label in most cases, and this could also be inferred by the low F1 value of 9.3%, which was worse than all the prompting methods. For the few-shot setting, we can see that the accuracy and F1 value improved significantly for all the models. However, there was a

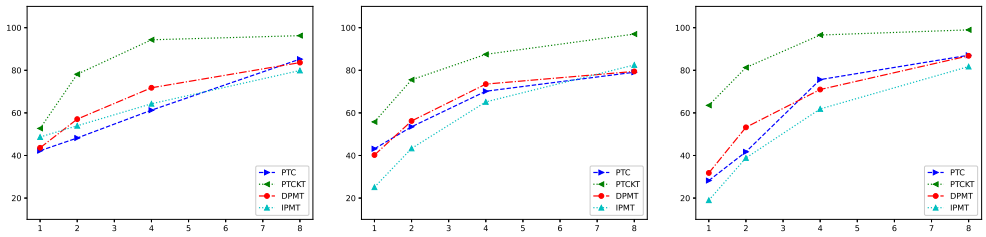
wide gap in the performance of all the prompt-based models. This result demonstrates the importance of a well-designed prompting strategy for prompt learning.

(3) Comparing the performance between the few-shot and zero-shot settings, all the models obtained remarkably increased performance after training on a few samples. Compared with the prompt-based models, the fine-tuned models had more improvement after training. These results show that a small amount of data annotation is cost-effective both for prompting and fine-tuning and fine-tuning is more sensitive to the training data.

Overall, Tables 2 and 3 show that knowledge is useful when inferring text labels and either encoding knowledge into the prompting template or probing knowledge together with classifying headlines could improve the performance of text classification.

3.3. Impact of Training Data Size

We conducted additional experiments to see the effect of the training data size on our methods. We used different sizes of training data and kept the test data the same in all experiments. Figure 5 shows the F1 scores on the datasets. We set K to 1, 2, 4, and 8 to form different training data. From this result, we can see that there was a similar performance trend when the data size grew for all three datasets, and normally, more training data could achieve better performance, while the growth rate decreased as the data increased. With different data sizes, the PTCKT model achieved the best performance without doubt. However, the relative performance of PTC, DPMT, and IPMT varied with different K. On average, DPMT ranked first, PTC second, and IPMT last. This is consistent with our previous findings from Table 2.

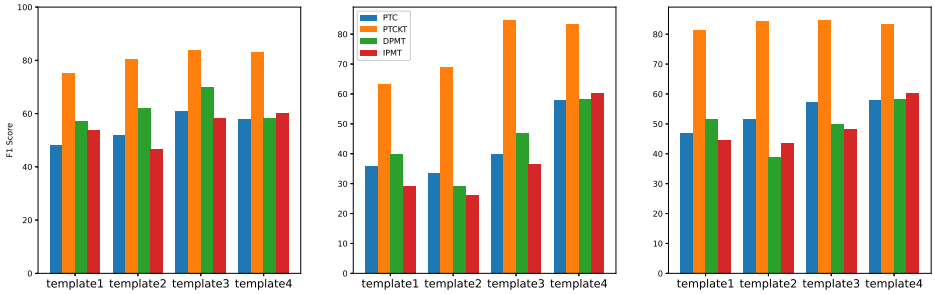


(**a**) NLPCC (**b**) THUNews (**c**) AIStudio

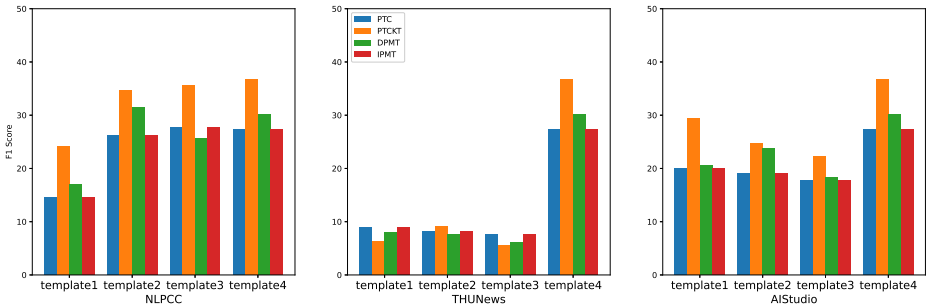
Figure 5. F1 score with different values of K for the few-shot setting.

3.4. Impact of Prompting Templates

The prompting function plays a key role in the performance of prompt learning. We investigate several prompting templates in Table 4 for the prompt-based methods on the three datasets. The experimental results are shown in Figure 6. For the same dataset and the same method, the performance varied remarkably across different templates, meaning the knowledge patterns stored in the PLM with different distributions. Except for Template 1 and Template 3 in the zero-shot setting with THUNews, the PTCKT model supplied with the extra knowledge obtained the best performance for all the templates and both settings across the datasets. This result indicated that PTCKT could make use of different knowledge patterns more efficiently than the other methods.



(**a**) Few-Shot Performance



(**b**) Zero-Shot Performance

Figure 6. F1 score with different prompting templates.

3.5. Interpretability of Our Models

We show two headlines from the data and the predicted labels by the models in Table 5. In the first sample, the PTC model mistakenly predicted the label as "finance". PTCKT and our multi-task models correctly inferred the label as they have the knowledge "minecraft is a game". In the second example, our multi-task models improperly classified this news to entertainment because there is another "Li Na" who is a singer. These two examples showed that inferring conceptual knowledge together with the headline label could offer some reasons why the model chose the category. In other words, our multi-task models offer knowledge interpretability for text classification.

Table 4. Different templates for the prompting models. [x] denotes the headline text; [e] represents the entity in [x]; [mask] refers to the words to be predicted.

Models	Prompting Templates
PTC/PTCKT	1. [x]， 这是一条[mask]新闻。(This is a [mask] news.) 2. [x]，这条新闻的类型是[mask]。(The type of this news is [mask]) 3. [x]， 类别:[mask]。(type:[mask]) 4. [x]， 这条新闻的主题是 [mask]。(The topic of this news is [mask])
DPMT	1. [x]，[e]是[mask$_c$]，这是一条[mask$_h$]新闻。([e] is [mask$_c$], This is a [mask$_h$] news) 2. [x]，[e]的类型是[mask$_c$],新闻的类型是[mask$_h$]。 The type of [e] is [mask$_c$], the type of this news is [mask$_h$] 3. [x]，[e]：[mask$_c$]，类别：[mask$_h$]。([e]:[mask$_c$],type:[mask$_h$]) 4. [x]，[e]是[mask$_c$]的实体，新闻的主题是[mask$_h$]。 [e] is a entity of [mask$_c$], the topic of this news is [mask$_h$]
IPMT	1. [x]， [e] 是 [mask$_c$]。([e] is [mask$_c$]) [x]， 这是一条 [mask$_h$] 新闻。(This is a [mask$_h$] news) 2. [x]， [e] 的类型是 [mask$_c$]。(The type of [e] is [mask$_c$]) [x]， 新闻的类型是 [mask$_h$]。(The type of the news is [mask$_h$]) 3. [x]， [e]： [mask$_c$]。([e] : [mask$_c$]) [x]， 类别： [mask$_h$] 。(type: [mask$_h$]) 4. [x]， [e] 是 [mask$_c$] 的实体。([e] is the entity of [mask$_c$]) [x]， 新闻的主题是 [mask$_h$] 。(the topic of the news is [mask$_h$]

Table 5. Two examples from the data. Underlined phrases are the entities, and the class labels of these examples are game and sports, respectively. [x] represents the headline, and other parenthetical text is the words predicted by the models.

Headline:	我的世界1.10最新生物召唤指令详解 Detailed explanation of the latest creature summoning instructions in minecraft1.10
PTC:	[x]，这是一条[金融]新闻。 This is a [financial] news.
PTCKT:	[x]， 我的世界是游戏，这是一条[游戏]新闻。 Minecraft is a game, this is a [game] news
DPMT:	[x]， 我的世界是[游戏]，这是一条[游戏]新闻。 Minecraft is a game, this is a [game] news.
IPMT:	[x]，我的世界是[游戏](Minecraft is a game) [x]，这是一条[游戏]新闻。 (this is a [game] news.)
Headline:	李娜在法向全球发出来汉邀约 Li Na sent an invitation to Wuhan to the world in France
PTC:	[x]，这是一条[历史]新闻。 this is a [history] news
PTCKT:	[x]，李娜是体育人物，这是一条[体育]新闻。 Li Nais a sports figure, this is a [sports] news
DPMT:	[x]，李娜是[娱乐人物]，这是一条[娱乐]新闻。 Li Na is an [entertainment figure], this is an [entertainment] news.
IPMT:	[x]，李娜是[娱乐人物]。(Li Na is [entertainment figure].) [x]，这是一条[娱乐]新闻。(this is an [entertainment] news.)

4. Related Work

This work focused on knowledge-guided prompt learning for text classification. As mentioned above, we propose to learn knowledge probing and text classification in a multi-task prompt learning framework. There are three groups of research that are related: knowledge probing, prompt-based text classification, and multi-task learning:

Knowledge probing: As implicit knowledge is learned by pre-training language models when training on a large scale of datasets, knowledge probing [6,23] is proposed to analyze the factual and commonsense knowledge stored in PLMs. Reference [6] utilized a cloze-style prompting template to probe the knowledge that PLMs obtain during pretraining and released a public available dataset for evaluation. After that, various work was performed aimed at extracting such knowledge more effectively [7,24,25]. Different from previous work, Reference [14] proposed to probe domain-specific knowledge instead of general domain knowledge, and they created a biomedical benchmark. The experiments showed that the biomedical-specific PLM contained more biomedical factual information than the general PLM.

Prompt-based text classification: With the emergence of GPT-3 [10], PLM-based prompt learning has received considerable attention, especially in a few-shot setting. For text classification, Reference [11] proposed the PET model, which utilizes prompt learning to annotate a large unlabeled dataset for future training. After that, prompting research can mainly be divided into two kinds of work: prompt engineering [21,26] and answer engineering [27,28]. Prompt engineering is aimed at choosing a proper prompting template for downstream tasks. For example, instead of setting the prompting template manually, Reference [26] utilized the seq2seq pre-trained model to generate the templates automatically. Answer engineering aims at constructing a suitable map between the prediction

space of PLMs and the actual labels. For instance, Reference [20] proposed to make use of a knowledge base to enrich the output space of the certain class.

Multi-task learning: Multi-task learning, which learns multiple tasks simultaneously, has been widely used in natural language processing. It exploits multiple task-generic and task-specific information, making it suitable for few-shot learning [13]. For example, Reference [29] exploited a multi-task framework to solve two text classification tasks together in the few-shot scenario, and their experimental results demonstrated the effectiveness of the multi-task model. Normally, the related tasks should share parameters in some strategies [29,30]; thus, they are able to reinforce each other by updating the shared parameter during the training process. Recently. Reference [31] developed a system that maps various NLP tasks into human-readable prompted form, combining multi-task learning with prompt learning for zero-shot learning and leading to zero-shot task generalization.

5. Conclusions

In this paper, we studied how to solve the problem of text classification based on prompt learning guided by knowledge. We assumed that knowledge and text labels are highly correlated and argued that exploiting knowledge explicitly can improve the efficiency of text classification. In particular, this paper proposed two approaches to exploiting knowledge. One is to directly encode the knowledge in the prompt template, and the other is to solve text classification and knowledge detection in a multi-task prompt model. The experiments showed that our proposed model worked well for the few-shot and zero-shot settings in most cases. With the established multi-task model, we also manually examined the predicted labels and found that the retrieved conceptual knowledge can semantically improve the interpretability of the predicted category labels.

Limitations: The model proposed in this paper is only effective when text classification is related to specific knowledge, which limits its applicability. In addition, the model is incapable of handling the classification of long and complex texts requiring much knowledge.

Future work: In the future, we intend to apply the method to alphabetic languages such as English to demonstrate its applicability in different languages. We also intend to study the classification of long text containing multiple entities.

Author Contributions: Conceptualization, L.W.; methodology, L.W.; software, L.W.; validation, R.C. and L.L.; formal analysis, L.W.; investigation, L.W.; resources, R.C.; data curation, L.W.; writing—original draft preparation, L.W.; writing—review and editing, L.W., R.C. and L.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by: Beijing Information Science and Technology University, Grant Number 2022XJJ22, and Guangxi Key Lab of Multi-source Information Mining & Security (MIMS21-M-04).

Data Availability Statement: The adopted datasets came from the following public domain resources: http://tcci.ccf.org.cn/conference/2017/taskdata.php (accessed on 1 February 2023); http://thuctc.thunlp.org/ (accessed on 1 February 2023); https://aistudio.baidu.com/aistudio/datasetdetail/103654/0 (accessed on 1 February 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Leeuwenberg, A.; Moens, M.F. Temporal Information Extraction by Predicting Relative Time-lines. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics, Brussels, Belgium, 31 October–4 November 2018; pp. 1237–1246. https://doi.org/10.18653/v1/D18-1155.
2. Zhang, Z.; Takanobu, R.; Zhu, Q.; Huang, M.; Zhu, X. Recent Advances and Challenges in Task-Oriented Dialog Systems. *Sci. China Technol. Sci.* **2020**, *63*, 2011–2027. https://doi.org/10.1007/s11431-020-1692-3.
3. Flisar, J.; Podgorelec, V. Improving short text classification using information from DBpedia ontology. *Fundam. Informaticae* **2020**, *172*, 261–297.
4. Zhan, Z.; Hou, Z.; Yang, Q.; Zhao, J.; Zhang, Y.; Hu, C. Knowledge attention sandwich neural network for text classification. *Neurocomputing* **2020**, *406*, 1–11.

5. Tenney, I.; Xia, P.; Chen, B.; Wang, A.; Poliak, A.; McCoy, R.T.; Kim, N.; Durme, B.V.; Bowman, S.; Das, D.; et al. What do you learn from context? Probing for sentence structure in contextualized word representations. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
6. Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; Miller, A. 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), Association for Computational Linguistics, Hong Kong, China, 3–7 November 2019; pp. 2463–2473. https://doi.org/10.18653/v1/D19-1250.
7. Zhong, Z.; Friedman, D.; Chen, D. Factual Probing Is [MASK]: Learning vs. Learning to Recall. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; Association for Computational Linguistics, Online, 30 August–3 September 2021; pp. 5017–5033. https://doi.org/10.18653/v1/2021.naacl-main.398.
8. Howard, J.; Ruder, S. Universal Language Model Fine-tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018; Long Papers; Gurevych, I., Miyao, Y., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; Volume 1, pp. 328–339. https://doi.org/10.18653/v1/P18-1031.
9. Han, X.; Zhang, Z.; Ding, N.; Gu, Y.; Liu, X.; Huo, Y.; Qiu, J.; Yao, Y.; Zhang, A.; Zhang, L.; et al. Pre-trained models: Past, present and future. *AI Open* **2021**, *2*, 225–250.
10. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. In Proceedings of the Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, virtual, 6–12 December 2020.
11. Schick, T.; Schütze, H. Exploiting Cloze-Questions for Few-Shot Text Classification and Natural Language Inference. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, 19–23 April 2021; Merlo, P., Tiedemann, J., Tsarfaty, R., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 255–269. https://doi.org/10.18653/v1/2021.eacl-main.20.
12. Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; Neubig, G. Pre-Train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Comput. Surv.* 2022, *Just Accepted*. https://doi.org/10.1145/3560815.
13. Wang, Y.; Yao, Q.; Kwok, J.T.; Ni, L.M. Generalizing from a Few Examples: A Survey on Few-shot Learning. *Acm Comput. Surv.* **2020**, *53*, 1–34. https://doi.org/10.1145/3386252.
14. Sung, M.; Lee, J.; Yi, S.; Jeon, M.; Kim, S.; Kang, J. Can Language Models Be Biomedical Knowledge Bases? In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing; Association for Computational Linguistics: Online and Punta Cana, Dominican Republic, Virtual, 7–11 November 2021; pp. 4723–4734. https://doi.org/10.18653/v1/2021.emnlp-main.388.
15. Chen, J.; Hu, Y.; Liu, J.; Xiao, Y.; Jiang, H. Deep Short Text Classification with Knowledge Powered Attention. In Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, HW, USA, 27 January–1 February 2019; AAAI Press: Menlo Park, CA, 2019; pp. 6252–6259. https://doi.org/10.1609/aaai.v33i01.33016252.
16. Ding, N.; Hu, S.; Zhao, W.; Chen, Y.; Liu, Z.; Zheng, H.; Sun, M. OpenPrompt: An Open-source Framework for Prompt-learning. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL 2022—System Demonstrations, Dublin, Ireland, 22–27 May 2022; Basile, V., Kozareva, Z., Stajner, S., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 105–113. https://doi.org/10.18653/v1/2022.acl-demo.10.
17. Chen, J.; Wang, A.; Chen, J.; Xiao, Y.; Chu, Z.; Liu, J.; Liang, J.; Wang, W. CN-Probase: A Data-Driven Approach for Large-Scale Chinese Taxonomy Construction. In Proceedings of the 2019 IEEE 35th International Conference on Data Engineering (ICDE), Macao, China, 8–11 April 2019; pp. 1706–1709. https://doi.org/10.1109/ICDE.2019.00178.
18. Le Scao, T.; Rush, A. How many data points is a prompt worth? In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; Association for Computational Linguistics, Online, 6–11 June 2021; pp. 2627–2636. https://doi.org/10.18653/v1/2021.naacl-main.208.
19. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015.
20. Hu, S.; Ding, N.; Wang, H.; Liu, Z.; Wang, J.; Li, J.; Wu, W.; Sun, M. Knowledgeable Prompt-tuning: Incorporating Knowledge into Prompt Verbalizer for Text Classification. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, 22–27 May 2022; Muresan, S., Nakov, P., Villavicencio, A., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2022; pp. 2225–2240. https://doi.org/10.18653/v1/2022.acl-long.158.
21. Hambardzumyan, K.; Khachatrian, H.; May, J. WARP: Word-level Adversarial ReProgramming. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Trento, Italy, 20–23 May 2019; Long Papers; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; Volume 1, pp. 4921–4933. https://doi.org/10.18653/v1/2021.acl-long.381.

22. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. 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, Minneapolis, MN, USA, 2–7 June 2019; Long and Short Papers; Association for Computational Linguistics: Stroudsburg, PA, USA, 2019; Volume 1, pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423.
23. Petroni, F.; Lewis, P.S.H.; Piktus, A.; Rocktäschel, T.; Wu, Y.; Miller, A.H.; Riedel, S. How Context Affects Language Models' Factual Predictions. In Proceedings of the Conference on Automated Knowledge Base Construction, AKBC 2020, Virtual, 22–24 June 2020. https://doi.org/10.24432/C5201W.
24. Jiang, Z.; Xu, F.F.; Araki, J.; Neubig, G. How Can We Know What Language Models Know? *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 423–438. https://doi.org/10.1162/tacl_a_00324.
25. Shin, T.; Razeghi, Y.; Logan, I.; Wallace, E.; Singh, S. AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. *arXiv* **2020**, 4222–4235. arXiv:2010.15980. . https://doi.org/10.18653/v1/2020.emnlp-main.346.
26. Gao, T.; Fisch, A.; Chen, D. Making Pre-trained Language Models Better Few-shot Learners. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Online, 1–6 August 2021; Long Papers; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; Volume 1, pp. 3816–3830. https://doi.org/10.18653/v1/2021.acl-long.295.
27. Schick, T.; Schmid, H.; Schütze, H. Automatically Identifying Words That Can Serve as Labels for Few-Shot Text Classification. In Proceedings of the 28th International Conference on Computational Linguistics; International Committee on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; pp. 5569–5578. https://doi.org/10.18653/v1/2020.coling-main.488.
28. Chen, X.; Zhang, N.; Xie, X.; Deng, S.; Yao, Y.; Tan, C.; Huang, F.; Si, L.; Chen, H. KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction. In Proceedings of the WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, 25–29 April 2022; Laforest, F., Troncy, R., Simperl, E., Agarwal, D., Gionis, A., Herman, I., Médini, L., Eds.; ACM: New York, NY, 2022; pp. 2778–2788. https://doi.org/10.1145/3485447.3511998.
29. Hu, Z.; Li, X.; Tu, C.; Liu, Z.; Sun, M. Few-Shot Charge Prediction with Discriminative Legal Attributes. In Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, NM, USA, 20–26 August 2018; Bender, E.M., Derczynski, L., Isabelle, P., Eds.; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 487–498.
30. Ma, Y.; Cambria, E.; Gao, S. Label Embedding for Zero-shot Fine-grained Named Entity Typing. In Proceedings of the COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, Osaka, Japan, 11–16 December 2016; Calzolari, N., Matsumoto, Y., Prasad, R., Eds.; ACL: Stroudsburg, PA, USA, 2016; pp. 171–180.
31. Sanh, V.; Webson, A.; Raffel, C.; Bach, S.; Sutawika, L.; Alyafeai, Z.; Chaffin, A.; Stiegler, A.; Raja, A.; Dey, M.; et al. Multitask Prompted Training Enables Zero-Shot Task Generalization. In Proceedings of the Tenth International Conference on Learning Representations, ICLR 2022, Virtual, 25–29 April 2022. Available online: https://openreview.net/forum?id=9Vrb9D0WI4 (accessed on 1 February 2023).