applied sciences

MDPI

Article

Prompt Language Learner with Trigger Generation for Dialogue Relation Extraction

Jinsung Kim [1,†], Gyeongmin Kim [2,†], Junyoung Son [1,†] and Heuiseok Lim [1,2,*]

[1] Department of Computer Science and Engineering, Korea University, 145, Anam-ro, Seongbuk-gu, Seoul 02841, Republic of Korea; jin62304@korea.ac.kr (J.K.); s0ny@korea.ac.kr (J.S.)
[2] Human-Inspired AI Research, 145, Anam-ro, Seongbuk-gu, Seoul 02841, Republic of Korea; totoro4007@gmail.com
* Correspondence: limhseok@korea.ac.kr
† These authors contributed equally to this work.

Abstract: Dialogue relation extraction identifies semantic relations between entity pairs in dialogues. This research explores a methodology harnessing the potential of prompt-based fine-tuning paired with a trigger-generation approach. Capitalizing on the intrinsic knowledge of pre-trained language models, this strategy employs triggers that underline the relation between entities decisively. In particular, diverging from the conventional extractive methods seen in earlier research, our study leans towards a generative manner for trigger generation. The dialogue-based relation extraction (DialogeRE) benchmark dataset features multi-utterance environments of colloquial speech by multiple speakers, making it critical to capture meaningful clues for inferring relational facts. In the benchmark, empirical results reveal significant performance boosts in few-shot scenarios, where the availability of examples is notably limited. Nevertheless, the scarcity of ground-truth triggers for training hints at potential further refinements in the trigger-generation module, especially when ample examples are present. When evaluating the challenges of dialogue relation extraction, combining prompt-based learning with trigger generation offers pronounced improvements in both full-shot and few-shot scenarios. Specifically, integrating a meticulously crafted manual initialization method with the prompt-based model—considering prior distributional insights and relation class semantics—substantially surpasses the baseline. However, further advancements in trigger generation are warranted, especially in data-abundant contexts, to maximize performance enhancements.

Keywords: dialogue relation extraction; information extraction; trigger generation; prompt-based learning

Citation: Kim, J.; Kim, G.; Son, J.; Lim, H. Prompt Language Learner with Trigger Generation for Dialogue Relation Extraction. *Appl. Sci.* **2023**, *13*, 12414. https://doi.org/10.3390/app132212414

Academic Editor: Rafael Valencia-Garcia

Received: 10 October 2023
Revised: 7 November 2023
Accepted: 14 November 2023
Published: 16 November 2023

1. Introduction

Relation extraction (RE) aims to extract structured knowledge from unstructured text and is widely used in various downstream tasks such as knowledge base construction and question answering [1]. Although most existing RE systems focus on sentence-level RE and have achieved promising results on several benchmark datasets [2,3], they are limited in their representation ability to extract relational facts from multiple sentences [4]. The capability only to capture intra-sentence relational facts cannot cover numerous relational facts that appear across multiple sentences in a document or with more than one speaker in a dialogue, and understanding inter-sentence and intra-document information is more significant in practical scenarios [5,6]. Therefore, several studies have shifted their focus toward more challenging but practical RE tasks that require extracting relational information from more extended and complicated contexts, such as documents and dialogues [7,8].

The dialogue-based relation extraction (DialogRE) task aims to predict the relation(s) between two entities that appear in an entire dialogue and requires the cross-sentence RE technique in the conversational setting with multi-speakers and multi-utterances [9]. Due to the multi-occurrences of speakers and utterances in a dialogue, meaningful information

that supports the relational facts is spread over the entire dialogue, resulting in low relational information density. To effectively capture and understand the scattered relational information in a dialogue, it is essential to directly exploit the pre-trained language model (PLM) knowledge by appropriately guiding the model on which information is significant in the conversation [10]. Therefore, to leverage the knowledge inherent in PLM and guide it to identify important information in conversations, we adopt a prompt-based fine-tuning approach along with a trigger generation method in the DialogRE task.

Concerning the direct exploitation of the knowledge of PLM first, a prompt-based learning approach has been proposed and is advantageous in consistency in learning. Unlike the conventional fine-tuning approach, which utilizes the representation of a special classification token `[CLS]` from an additional classifier, the prompt-based learning approach directly exploits the learned knowledge of a pre-trained language model by alleviating discrepancy [11]. In particular, a prompt-based approach using PLMs such as Bidirectional Encoder Representations from Transformers (BERT) [12] solves downstream tasks by regarding them as a cloze task using the `[MASK]` token as a direct predictor, resulting in bridging the gap of learning objectives between pre-training phase and downstream task.

Moreover, providing appropriate guidance on which contextual representation is remarkable for the model in the DialogRE system helps to alleviate the challenge of dialogue relation extraction due to the low relational information density. The trigger, which can be described as a potential explanatory element, is defined as "the smallest span of continuous text that clearly indicates the existence of a given relation" and plays an essential role in understanding contextual features in the dialogue [9]. Table 1 shows a dialogue example that contains multiple entity pairs and triggers. For example, the first relation (R1) can be easily predicted when the word "mom" is accurately captured and predicted by the model, but there are no triggers aidful for the prediction in the cases of the relation types R3 and R4. As the amount of annotated triggers in the dataset is limited, this scarcity of triggers leads to the difficulty in providing guidance on which information is significant to the model for predicting relations between a given entity pair.

Table 1. This table demonstrates an example of DialogRE data. The triggers are bold and the entities are underlined in the given dialogue. They are scattered throughout the dialogue, resulting in low relational information density.

Dialogue			
1 Speaker 1, Speaker 2: Hi 2 Speaker 3: Hi! Hey **mom**. 3 Speaker 4: This is such a great party! 35 years. Very impressive, do you guys have any pearls of wisdom? 4 Speaker 2: Jack? 5 Speaker 1: Why would you serve food on such a sharp stick? 6 Speaker 3: That's a good question, **dad**. That's a good question ... 7 Speaker 4: Hmmm ...			
	Argument pair	Trigger	Relation Type
R1	(Speaker 3, Speaker 2)	mom	per:parents
R2	(Speaker 1, Speaker 3)	dad	per:children
R3	(Speaker 1, Speaker 2)	none	per:spouse
R4	(Speaker 1, Jack)	none	per:alternate_names

To these ends, we explore a prompt-based learner with trigger generation for dialogue relation extraction to take advantage of the inherent knowledge in PLMs and guide them to identify crucial information in dialogues. Specifically, the DialogRE downstream task is solved with the prompt-based masked-language modeling (MLM) objective, and also the effectiveness of utilization and manual initialization of prompt tokens is analyzed.

In addition, the potential of the generated triggers by utilizing the generative approach is explored.

The contributions of this study are summarized in three parts. (1) We present a prompt-based fine-tuning approach with a trigger generation method that alleviates the challenges of dialogue relation extraction. (2) We demonstrate that the prompt-based method, including the manual initialization method in our approach, significantly improves performance on the DialogRE task compared to the baseline model. (3) Moreover, we explore the effectiveness of extracted triggers by a generative approach and analyze their limitation with analytical experiments. By exploring these trigger-generation- and prompt-based approaches, our research aims to capture potential ways to directly leverage the model's implicit knowledge and guide the model to meaningful clues for dialogue relation extraction.

The remaining parts of this manuscript are organized into the following sections. In Section 2, previous works related to dialogue relation extraction (Section 2.1), prompt-based learning (Section 2.2) and trigger generation (Section 2.3) are introduced. Section 3 first explains the overall structure of our approach (Section 3.1), and the following sections consist of the descriptions of the trigger generation method (Section 3.2), the construction process of inputs including prompts (Section 3.3) and the deliberate initialization method of inserted prompts (Section 3.4). Afterward, Section 4 covers experimental results and findings, and Section 5 presents various analyses on the effectiveness of trigger- and prompt-based approaches. Finally, Section 6 concludes by summarizing the purpose and findings of this study.

2. Related Works

2.1. Dialogue-Based Relation Extraction

Relation extraction (RE) is a task to extract appropriate relation types between two entities from the given text, and the extracted structured information plays a critical role in information extraction and knowledge base construction [13]. Although typical RE systems have achieved promising results on several sentence-level RE benchmark datasets [3,14], due to the limitations of extracting relational facts from a single sentence in practical life, several studies concentrate on RE tasks in a more complicated and lengthy context, such as document and dialogue. In line with this research trend, the inter-sentence RE ability to consider the relations of entities scattered across multiple sentences or utterances is essential. Additionally, cross-sentence RE, which aims to identify relations between an entity pair not mentioned in the same sentence or relations that any single sentence cannot support, is an essential step in building knowledge bases from large-scale corpora automatically [5,15,16].

Although dialogues readily exhibit cross-sentence relations, most existing RE studies put their attention on texts from formal genres, such as professionally written and edited news reports [17–19], while dialogues have been under-studied. To this end, to deal with a conversational environment based on English Friends transcripts, the dialogue-based relation extraction (DialogRE) dataset was proposed [9]. Through this corpus, it is possible to train the model to capture relational facts that appear in dialogue effectively.

2.2. Prompt-Based Learning

In the previous DialogRE benchmark studies, the fine-tuning method is prevalently employed on PLMs such as BERT [12] and RoBERTa [20], and promising performances have been shown [21]. For example, there is an approach that made embeddings of objects using a gate mechanism [22] and another approach that used multi-turn embeddings and meta-information, such as whether an entity exists in dialogue, then constructed a graph, and fed the graphs into graph convolutional neural networks [23]. Additionally, studies have been conducted to improve relation extraction performance while maintaining a complex model structure based on graph features such as bi-graph and document-level heterogeneous

graphs [24,25]. Liu et al. [26] attempt a hierarchical understanding of dialogue context by leveraging turn-level attention.

However, the general fine-tuning approach leads to discrepancies as the training objectives in the fine-tuning phase for the downstream task are different from those used in the pre-training phase, resulting in the degraded generalization capability [11]. For example, for the training of the BERT model, the learning through the `[MASK]` token is only employed in the pre-training step, and an extra classification layer is utilized in the fine-tuning step.

The prompt-based learning approach is proposed to alleviate the gap by increasing the consistency of learning objectives and effectively exploiting the learned knowledge of pre-trained language models (PLMs) in the downstream tasks. Unlike fine-tuning requiring adding extra layers on top of the PLMs, several prompt-based learning studies solve the downstream task with MLM objective by directly predicting the textual response to a given template. They employ the PLM directly as a predictor by completing a cloze task, thereby directly leveraging the knowledge of PLM learned in the pre-training step [10,11]. Specifically, the prompt-based learning approach updates the original input based on the template and predicts the label words with the `[MASK]` token. Afterwards, the model maps predicted label words to corresponding task-specific class sets. Several studies on the prompt-based learning approach have shown superior performance in low-resource setting [27–29]. Moreover, recent prompt-based approaches, combined with methods such as contrastive learning, are showing significant progress in various classification tasks, and their utilization not only in the field of classification but also for controlled text generation is encouraged [30–32].

2.3. Trigger Generation

Recent several DialogRE studies focus on utilizing additional explanations on the web and document text-based approach, and this tendency is also similar in natural language understanding tasks [33,34]. For example, the text in a study consisting of both weakly supervised examples that can be used for supervised pre-training and human-annotated examples comes from various publicly available sources on the internet that offer multiple domains and writing styles [18,35]. A common characteristic of these studies is that searching for additional explanatory information that can be conclusive evidence within the text is essential.

In a similar context, a study has also been devised that uses triggers, i.e., key evidence in DialogRE, as additional explanatory information [36]. In detail, the study tried to identify trigger spans in a given context using multi-tasking BERT and, accordingly, to leverage such signals to improve relation extraction. Similarly, An et al. [37] also attempt to exploit the trigger information by utilizing an extractive way. However, related studies are insufficient compared to the importance of utilizing triggers in dialogue relation extraction, and in particular, it is difficult to find a method using generative approaches.

3. Materials and Method

We explore an approach to enhance the capturing capability of pre-trained language models (PLMs) by exploiting a prompt-based learning approach and to guide on the crucial information, i.e., generated triggers for the dialogue relation extraction. In the DialogeRE task, each example X consists of a dialogue $D = \{s_1{:}u_1, s_2{:}u_2, \ldots, s_N{:}u_N\}$, subject entity e_1, and object entity e_2, where s_n is the n-th speaker and u_n is the corresponding utterance. Please note that the following parts denote the entity pair (e_1, e_2) as E. When $X = \{D, E\}$ is given, the dialogue-based relation extraction (DialogRE) task aims to predict an appropriate relation $r \in R$ from the set of pre-defined relations R between entities e_1 and e_2 by understanding D and capturing the scattered helpful information in it.

3.1. Prompt Language Learner with Trigger Generation

Figure 1 illustrates the model overview of our approach. Given an input text X, triggers regarded as informative in the dialogue are generated based on the dialogue D. Subsequently, these triggers are utilized to construct a prompt-based format for the input text using pre-defined prompt templates, which will be explained in detail in Section 3.3. The input with the prompt template is then fed into an appropriate model with a different learning objective, i.e., fine-tuning or prompt-based fine-tuning, depending on the type of the constructed input. When employing prompt-based fine-tuning, the model is trained to fill `[MASK]` token with a virtual relational token for each relation label. To that end, we add relational tokens to the model's vocabulary, which correspond to specific relation classes, such as `[per:friends]` for the '*per:friends*' relation label.

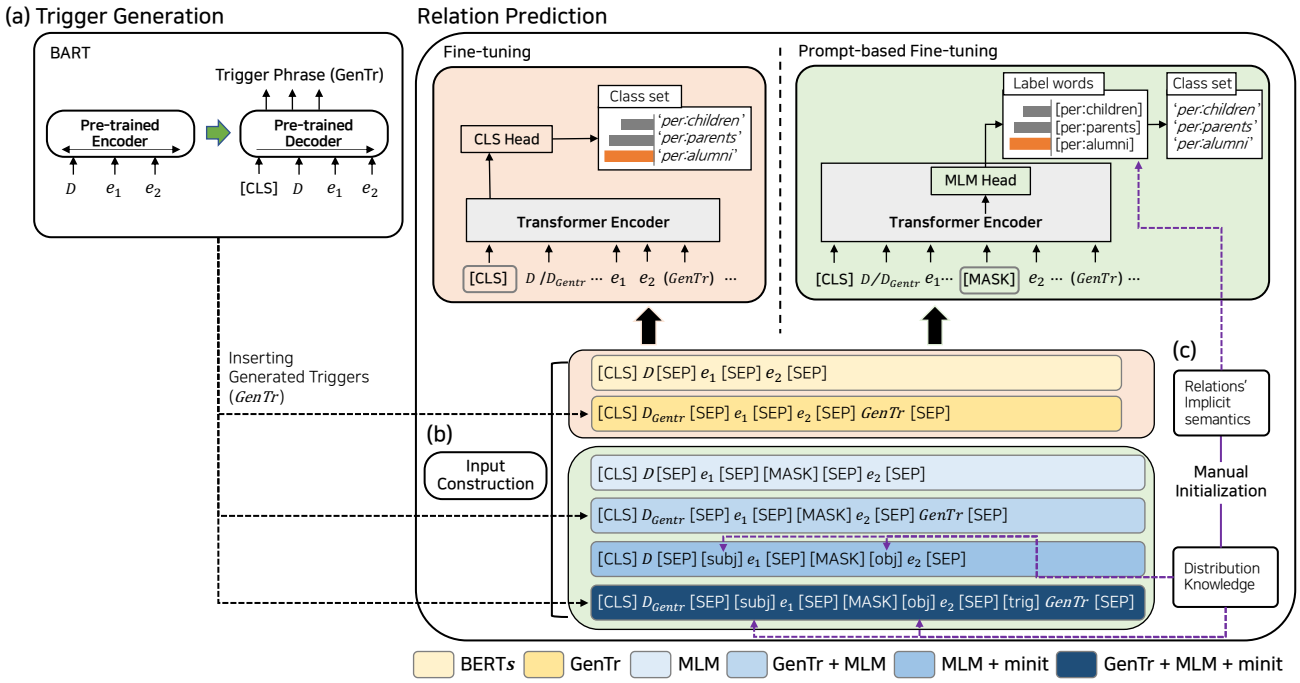

Figure 1. This figure demonstrates the overall model architecture. The model solved the task with the tokens from a model's vocabulary with the task's formalization as an MLM problem. For example, the label words in the prompt-based fine-tuning phase are from the model's vocabulary. Descriptions of each module (a), (b), and (c) are provided in each subsection.

Our approach is composed of the following three parts; (i) trigger generation, (ii) prompt-based fine-tuning method and (iii) the manual initialization of prompt tokens. The methods are applied to the basic dialogue relation extraction model BERT$_s$ [9]. With regard to the type of utilized methods, the relation prediction models are categorized into five types; (a) *GenTr*, (b) *MLM*, (c) $GenTr + MLM$, (d) $MLM + minit$, (e) $GenTr + MLM + minit$. BERT$_s$ and *GenTr* follow the previous fine-tuning approach, and $MLM, GenTr + MLM, MLM + minit$ and $GenTr + MLM + minit$ follow the prompt-based fine-tuning approach.

3.2. Trigger Generation

Since triggers are absent in a significant number of examples in the dataset, we intend to consider the critical information in predicting an appropriate relation by directly generating it. Unlike the explicit span extraction method, the generative approach is expected to produce implicit answers. The purpose of the generated triggers is to enhance the relation prediction capability by feeding them as one of the input features of the model. In contrast to previous trigger-related studies, our approach employs a generative model

with an encoder-decoder architecture, considering both the given entity pair (e_1, e_2) and the given dialogue D.

Our trigger generation module is illustrated in part (a) in Figure 1. This module generates *GenTr* (generated triggers) considering the given context and entity pair, thereby identifying the relation r using D and entity pair E, i.e., $f(D, E) \rightarrow R$. A single entity pair can include multiple relations in this process. As shown in the left part of Figure 1, the trigger generation module consists of encoder-decoder architecture by adopting the pre-trained BART [38] model. The input in fine-tuning step is constructed as "<s> D </s> E </s>". The module is taught to identify optimal triggers by defining the triggers of some examples where triggers exist as labels among the examples in the DialogRE dataset, and to generate triggers using the decoder of BART model. The number of generated triggers may be one or several, and the triggers generated in this way are also used as input features to the relation prediction module in the later step. Specifically, for given D, E, the trigger generator is trained as follows:

$$P(TR|D, E) = \prod_{i=1}^{n} p(r_i|D, E, tr_{<i}, \theta), \tag{1}$$

where TR is the ground-truth trigger sequence to generate and θ is the parameter set of the trigger generation module.

3.3. Input Construction

This phase is depicted in part (b) in Figure 1. To explore the effectiveness of the relation prediction based on how to construct the input structure, we distinguish the input structure construction approach into two objectives; fine-tuning and prompt-based fine-tuning. Following the widely used fine-tuning approach, we investigate the efficacy of the generated triggers. For the prompt-based fine-tuning approach, we explore the specific structure of a prompt template as it significantly impacts overall model performance. Thus, we systematically investigated the impact of different prompt design choices on the quality of extracted relations. In other words, the input design is constructed as six types depending on whether the generated triggers or prompt tokens are included, and the prompts are manually initialized. The set of input construction types is denoted as $TYPE$, i.e., $\{GenTr, MLM, GenTr + MLM, MLM + minit, GenTr + MLM + minit\}$, and a template function, $T(\cdot)$, is defined to map each example X to $T_{type \in TYPE}(X)$. Our input construction is employed to the input structure of BERT$_s$ [9] model, the baseline model for verifying the impact of our presented methods, and the input of BERT$_s$ is defined as "`[CLS]` D `[SEP]` e_1 `[SEP]` e_2 `[SEP]`"

To utilize the generated triggers when a dialogue D is given, we define D_{GenTr} as the dialogue where the phrases or words identical to the generated triggers are marked with trigger marker `[trig]`. Afterwards, the input, which consists of D_{GenTr} and an entity pair (e_1, e_2), is constructed by employing the template function T_{GenTr} as follows and is fed into the fine-tuning model:

$$T_{GenTr}(X) = \text{"}\texttt{[CLS]}\ D_{GenTr}\ \texttt{[SEP]}\ e_1\ \texttt{[SEP]}\ e_2\ \texttt{[SEP]}\ GenTr\ \texttt{[SEP]}\text{"}. \tag{2}$$

To leverage the parametric capability of PLM into the DialogRE task using the prompt-based fine-tuning approach, we regard the downstream task as an MLM problem. The MLM input type $T_{MLM}(X)$ is constructed by adding `[MASK]` token to the input structure of BERT$_s$ model, and the generated triggers are added on it to compose the MLM input type with the generated triggers $T_{GenTr+MLM}(X)$ as follows:

$$T_{MLM}(X) = \text{"}\texttt{[CLS]}\ D\ \texttt{[SEP]}\ e_1\ \texttt{[MASK]}\ e_2\ \texttt{[SEP]}\text{"}, \tag{3}$$

$$T_{GenTr+MLM}(X) = \text{"}\texttt{[CLS]}\ D_{GenTr}\ \texttt{[SEP]}\ e_1\ \texttt{[MASK]}\ e_2\ \texttt{[SEP]}\ GenTr\ \texttt{[SEP]}\text{"}. \tag{4}$$

Finally, as additional information such as entity type or distributional information can be employed as a guiding indicator for the model in the prompt-based approach, we

construct input designs $T_{MLM+minit}(X)$ and $T_{GenTr+MLM+minit}(X)$ by inserting additional prompt tokens based on the template function T_{MLM} and injecting the additional knowledge. Specifically, the prompt tokens `[subj]` and `[obj]` are inserted in front of each entity, and additional information (i.e., entity type information) is injected into the prompt tokens by initializing them deliberately. The detailed formulation of this prompt initialization method is described in the following Section 3.4. Therefore, the input structures for the manual initialization of prompt tokens are designed as follows:

$$T_{MLM+minit}(X) = \text{"}\texttt{[CLS]}\ D\ \texttt{[SEP]}\ \texttt{[subj]}\ e_1\ \texttt{[MASK]}\ \texttt{[obj]}\ e_2\ \texttt{[SEP]}\text{"}, \tag{5}$$

$$T_{GenTr+MLM+minit}(X) = \text{"}\texttt{[CLS]}\ D_{GenTr}\ \texttt{[SEP]}\ \texttt{[subj]}\ e_1\ \texttt{[MASK]}\ \texttt{[obj]}\ e_2\ \texttt{[SEP]}\ \texttt{[trig]}\ GenTr\ \texttt{[SEP]}\text{"}. \tag{6}$$

3.4. Prompt Manual Initialization

As described in the previous Section 3.3, the prompt tokens `[subj]` and `[obj]` inserted in the input are initialized with the prior distributions of entity types for the entities resulting in the learning of distributional knowledge for the model. In other words, the injection of distributional information is expected to help predict the relation(s) between an entity pair when they effectively learn the distribution of entity types. Specifically, inspired by previous studies on the manual initialization of prompts [39,40], we define the entity types as ET ={*"Person"*, *"Organization"*, *"Geographical entity"*, *"Value"*, *"String"*}, exploiting the pre-defined types in the DialogRE dataset. For a given prompt token $a \in \{\texttt{[subj]}, \texttt{[obj]}\}$ corresponding to each entity e_1 and e_2, we estimate the prior distributions of the entity types ϕ^a by calculating frequency in the dataset over ET as follows:

$$\tilde{e}(a) = \sum_{et \in ET} \phi^a_{et} \cdot e(et),\ a \in \{\texttt{[subj]}, \texttt{[obj]}\}, \tag{7}$$

where $e(\cdot)$ is the embedding from the PLM and $\tilde{e}(\cdot)$ is the initialized embedding of the prompt tokens.

Additionally, each relation representation is deliberately initialized by appending the set of virtual relational tokens V corresponding to the relation classes into the model's vocabulary and initializing them with the implicit semantics of the relations as aforementioned in Section 3.1. Suppose that the i-th semantic words set corresponding to the i-th component of V, i.e., the i-th virtual relational token v_i, is denoted as C_i. Specifically, v_i is computed by obtaining the average embedding of the semantic words set C_i. For instance, when the corresponding relational token of the relation label '*per:place_of_residence*' is `[per:place_of_residence]`, we initialize the token by aggregating the embeddings of the semantic words in the set, i.e., {"person", "place", "of", "residence"}. To be elaborated formally, the representation of v_i is calculated as follows:

$$\tilde{e}(v_i) = \frac{1}{|C_i|} \sum_{j}^{|C_i|} e(c_j), \tag{8}$$

where $\tilde{e}(\cdot)$ is the initialized embedding of the relation representation and c_j is the j-th component of C_i. These deliberate initialization processes are shown in part (c) in Figure 1.

4. Experiments

4.1. Experimental Setup

The experiments consist of a full-shot setting and a few-shot setting. In the few-shot setting, K, i.e., the number of examples given at once to train the model is set to three cases; 8, 16, 32. The dialogue-based relation extraction (DialogRE) dataset has two versions (v1 and v2), and the updated version fixed a few annotation errors, resulting in increased prediction difficulty for models; we used the second version.

The performance of each model was measured with F1 and $F1_c$ scores. The $F1_c$ score is the metric proposed in the DialogRE task for supplementing the F1 score in the conversational setting. Specifically, instead of being provided an entire dialogue, the model has to predict with only the utterances where an entity pair and phrases corresponding to the trigger first appear. The performance was measured as the average result of three different seeds.

The T5-large [41] model, a representative generative model with encoder-decoder architecture, was adopted for the trigger generation module. Also, $BERT_s$, the model proposed in the previous study [9], was adopted for the relation extraction baseline. The model adjusts the form of the input sequence by proposing a new template instead of inserting the special tokens to mark the start and end positions of entities. $BERT_s$ and our models were trained with the backbone of BERT-base [12] using the DialogRE train data. The hyperparameters for the training are as follows: 512 of sequence length, 8 of batch size, and training occurs during 30 epochs using AdamW [42] optimizer with weight decay 0.01.

4.2. Experimental Results

4.2.1. Full-Shot Setting

Table 2 demonstrates the performance change of the models to which our methods were applied compared to the baseline model $BERT_s$. All our models achieved performance improvement in F1 and $F1_c$ scores of the development set, and those models showed a similar overall tendency except for the model with the generated triggers (*GenTr*) in the development set. The F1 score of the *GenTr* model in the development set decreased by 0.14%p compared to the baseline model. In particular, the model with MLM and manual initialization methods ($MLM + minit$) showed the overall highest performance across the development and test set results. Compared to the $BERT_s$ model, the $MLM + minit$ model showed 4.32%p and 3.57%p improved performance at F1 and $F1_c$ score, respectively, in the development set and achieved 3.75%p and 3.31%p of performance gains in the test set.

Table 2. This table shows the main experimental results in a full-shot setting. The * mark indicates the re-implemented version. P and R indicate precision score and recall score, respectively. In addition, F1*c* score is the F1 score in the conversational setting. The best performance is bold and the second best is underlined.

Full-Shot Setting												
Method	**Dev**						**Test**					
	P	**R**	**F1**	**P_c**	**R_c**	**$F1_c$**	**P**	**R**	**F1**	**P_c**	**R_c**	**$F1_c$**
$BERT_s$ *	61.14	62.50	61.81	63.84	51.27	56.86	59.46	60.58	59.94	63.33	49.09	55.29
+*GenTr*	62.91	63.24	63.07	65.00	51.72	57.60	59.00	60.68	59.80	63.20	49.45	55.48
+*MLM*	62.88	64.20	63.52	65.59	52.94	58.58	<u>62.27</u>	<u>62.29</u>	<u>62.27</u>	65.43	50.69	57.12
+*GenTr* +*MLM*	<u>65.09</u>	64.76	64.92	67.07	53.38	59.44	61.13	60.95	61.03	65.42	50.13	56.76
+*MLM* +minit	**65.47**	**66.81**	**66.13**	<u>67.94</u>	**54.43**	**60.43**	**64.00**	**63.44**	**63.69**	<u>67.04</u>	**52.05**	**58.60**
+*GenTr* +*MLM* +minit	64.92	<u>66.35</u>	65.62	**67.98**	<u>53.99</u>	<u>60.18</u>	61.18	62.67	61.90	**67.19**	<u>50.86</u>	<u>57.87</u>

Although the *GenTr* model showed 1.26%p, 0.74%p of performance gains in the development set at F1 and $F1_c$ scores, it showed only a slight increase or decrease with the amount of approximately 0.1%p in the test set. Moreover, when comparing *GenTr*, *MLM* and $GenTr + MLM$ models, in the development set, the model in which the MLM method and the generated triggers were used together ($GenTr + MLM$) showed the best performance with 64.92 of F1 and 59.44 of $F1_c$ scores, whereas in the test set, the model in which only the masked-language modeling (MLM) method was used (*MLM*) performed the highest with 62.27 and 57.12.

Figure 2 illustrates the difference between the F1 score and F1*c* score for each model. This can be interpreted as the smaller the deviation, the more consistent the performance

is in conversation settings. First, in the validation set on the left, the results of all models except the *MLM* model are located above that of the baseline (BERT*s*). In contrast, in the test set on the right, the difference between the F1 score and F1*c* score of the *MLM* model and the *MLM* + *minit* model is larger than that of the baseline, and the differences of the other models are observed to be located under the baseline. According to these results, the results of the validation set may make it seem as if the impact of the provision of triggers and prompt-based approaches is inconsistent in conversation settings. However, regarding the unseen data of the test set, the model that provided the trigger (*GenTr*) and the *GenTr* + *MLM* and *GenTr* + *MLM* + *minit* models that also applied the prompt-based method show less difference between F1 and F1*c* scores, implying the effectiveness of the simultaneous application of trigger provision and prompt-based methods in practical inference situations.

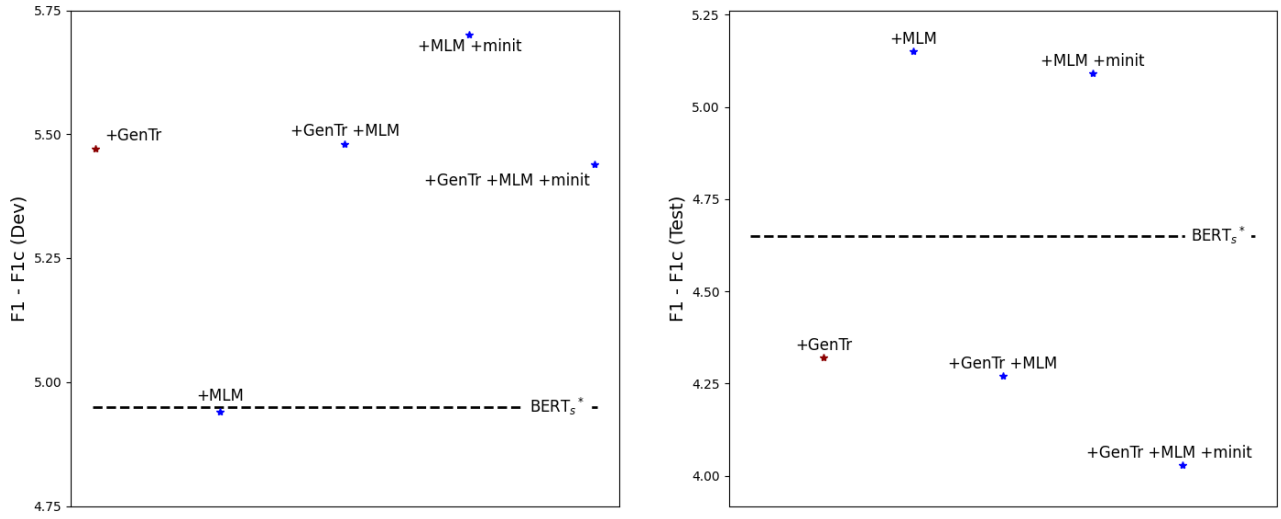

Figure 2. This figure illustrates the difference between F1 and F1$_c$ performance of each model. The figure on the left is the result from the validation set, and the figure on the right is the result from the test set. The dotted lines indicate BERT*s*, the baseline model, and each dot (*) represents the results of models to which each methodology was applied. Red dots represent the results of the fine-tuned model, and blue dots represent those of the prompt-based model.

4.2.2. Few-Shot Setting

Table 3 shows the few-shot results according to the number of shots *K*. In the development set, the model where MLM and manual initialization were used (*MLM* + *minit*) obtained the overall highest scores both at F1 and F1$_c$ scores regardless of *K*, demonstrating significant performance improvements compared to the baseline model. For example, when *K* is 8, the *MLM* + *minit* model outperformed the BERT$_s$ model by 12.04%p and 10.93%p at F1 and F1$_c$ scores, respectively.

In the test set, except for the F1 score of *K* = 8, the *MLM* + *minit* also showed the overall highest performances at F1 and F1$_c$ scores, achieving 10.23%p, 4.27%p, 4.51%p of performance improvements at F1 and 8.88%p, 3.7%p and 2.43%p at F1$_c$, when *K* is 8, 16 and 32, respectively, compared to the baseline BERT$_s$. When *K* is 8, the model that demonstrated the highest test F1 score is the model with the generated triggers, MLM method and manual initialization method (*GenTr* + *MLM* + *minit*), showing 37.95.

In particular, when the generated triggers were supplemented to the baseline model (*GenTr*), the model demonstrated the most significant performance increase when *K* is 8, showing a performance gain of 8 points: 6.15%p and 5.92%p at F1 and F1$_c$ scores in the development set and 4.04%p and 3.57%p in the test set. In the case of *K* = 32 of the test set, the F1 and F1$_c$ scores of the *GenTr* model marginally dropped by 0.84%p and 1.56%p, respectively. Additionally, unlike with the full-shot setting, among the *GenTr*, *MLM* and

GenTr + *MLM* models, the model with both the MLM method and the generated triggers (*GenTr* + *MLM*) did not guarantee the best overall performance in the development set, and the *MLM* model did not achieve the most significant overall performance increase in the test set. According to the change in *K*, the model with the most performance among the three has changed.

Table 3. This table shows the relation extraction results in the few-shot setting. K indicates the number of shots, i.e., the number of samples given, and consists of 8, 16 and 32. The * mark indicates the re-implemented version. The best performance is bold and the second best is underlined.

Few-Shot Setting												
Method	**Dev**						**Test**					
	K = 8		**K = 16**		**K = 32**		**K = 8**		**K = 16**		**K = 32**	
	F1	**F1$_c$**	**F1**	**F1$_c$**	**F1**	**F1$_c$**	**F1**	**F1$_c$**	**F1**	**F1$_c$**	**F1**	**F1$_c$**
BERT$_s$ *	27.56	26.30	41.49	38.61	47.87	44.17	27.67	26.49	42.84	40.11	48.71	45.55
+ *GenTr*	33.71	32.22	46.26	42.47	49.86	45.25	31.71	30.47	44.83	41.46	47.87	43.99
+ *MLM*	28.41	25.67	39.48	35.79	48.25	43.49	28.52	25.88	43.17	38.72	50.35	45.15
+ *GenTr* + MLM	32.95	30.43	45.76	41.27	51.03	46.46	30.86	28.78	43.92	39.21	48.01	44.08
+ *MLM* + minit	**39.60**	**37.23**	**48.27**	**44.59**	**53.20**	**48.20**	37.90	**35.37**	**47.11**	**43.81**	**53.22**	**47.98**
+ *GenTr* + *MLM* + minit	38.97	36.43	46.48	43.25	51.16	46.65	**37.95**	35.36	45.63	42.29	49.66	45.51

Figure 3 demonstrates the performance change according to the K value of each model in the few-shot settings of the test set. The chart on the left shows the change in the F1 score according to the K value, and the chart on the right shows the change in the F1*c* score. Regardless of the change in K value, the *MLM* + *minit* and *GenTr* + *MLM* + *minit* models consistently show better or similar performance than the baseline model (BERT*s*) in both F1 and F1*c*. On the other hand, the *GenTr* and *GenTr* + *MLM* models outperformed the baseline performance when the K value was low in both F1 and F1*c*, but as the K value increased, the increase was observed to be smaller than that of the baseline's performance. Based on these results, it can be recognized that leveraging the prompt-based technique along with deliberate initialization of inserted prompts ensures consistent performance across few-shot environments.

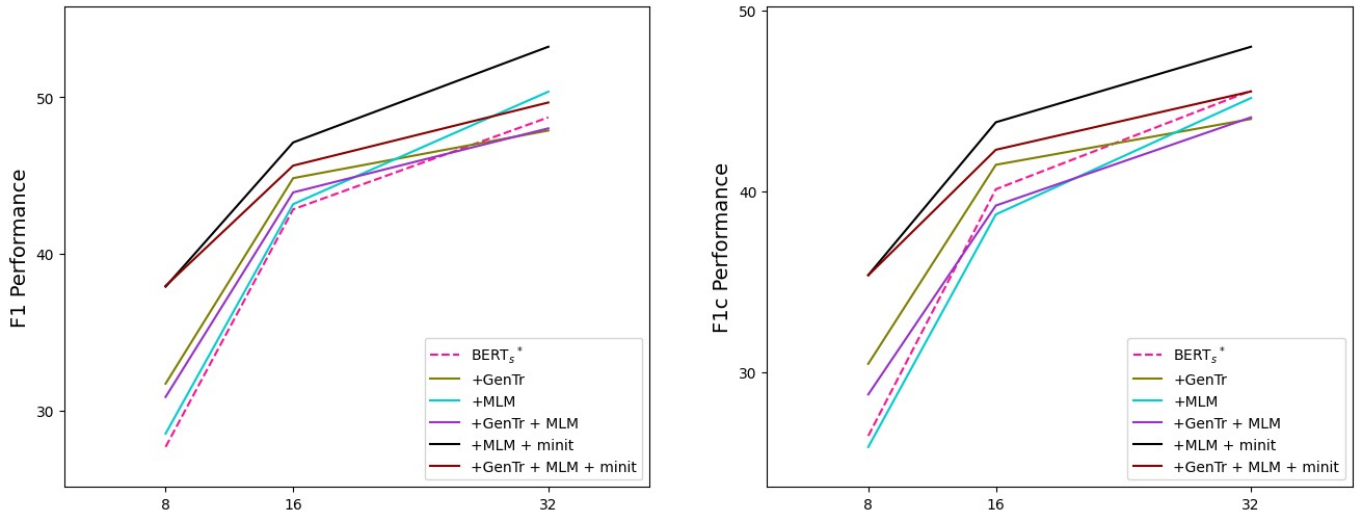

Figure 3. This figure illustrates performance changes depending on the value of K (number of shots). The *x*-axis represents K values, and the *y*-axis represents F1 (**left**) and F1*c* (**right**) performances of each model. The * mark indicates the re-implemented version.

5. Discussion

In this section, we analyze several findings based on the main results presented above (Tables 2 and 3) and additional experimental results.

5.1. Learning Distributional Knowledge with Prompt Manual Initialization Is Advantageous

According to Tables 2 and 3, the model with the overall highest performances in both full-shot and few-shot settings is the *MLM* + *minit* model. Moreover, the performance gap between the *MLM* + *minit* model and the *MLM* model is considerable, showing at least 1.48%p of improvement both at F1 and $F1_c$ in the full-shot setting. We assume that as the *MLM* model does not contain prompt tokens for an entity pair, i.e., `[subj] and [obj]`, and its relation representation is randomly initialized, the model has difficulty learning the distribution of the training dataset, compared to the *MLM* + *minit* model. Therefore, it is confirmed that injecting the knowledge on the entity type distribution and the semantic information from relation classes is effective, as presented in the previous study [40].

5.2. Generated Triggers Are Apt to Be Practical When Given a Small Number of Examples

As shown in Tables 2 and 3, providing triggers was more effective when only a few examples were given regardless of the learning objectives, i.e., the fine-tuning and prompt-based fine-tuning approaches.

First, compared with the performance of the baseline model, the fine-tuning model with the generated triggers (*GenTr*) showed that the full-shot performance slightly dropped by 0.14%p at F1 score in the test set, and the performance in the 32-shot setting also decreased by 0.84%p, implying inconsistent effectiveness. However, compared to the full-shot or 32-shot setting K = 32), the *GenTr* model in the 8- or 16-shot setting showed more significant performance gains, achieving 4.04%p and 1.99%p of improvements at the F1 score and 3.98%p and 1.35%p of increase at $F1_c$ score in the test set, respectively. This tendency of performance changes was similarly shown at the $F1_c$ score, achieving improvements of at least 1.62%p in the 8- and 16-shot settings, whereas only a minor performance increase or decrease is shown in the full-shot and 32-shot settings.

In addition, the *GenTr* + *MLM* model corresponding to the prompt-based fine-tuning approach with the generated triggers also demonstrated similar results. In the 8- and 16-shot settings of the test set, it showed higher performances at F1 and $F1_c$ scores by 2.34%p and 2.9%p (K = 8) and by 0.75%p and 0.49%p (K = 16) than the prompt-based model without the generated triggers (*MLM*). In contrast, the *MLM* model achieved higher performances in the 32-shot and the full-shot settings. These results are to be interpreted that the provided triggers serve as helpful clues in a setting with little data to train on, regardless of the learning objectives.

5.3. A Critical Point Is How Appropriate Triggers Are Generated

The generative approach for the triggers did not demonstrate significant performance improvement, particularly in the 32-shot and the full-shot settings. We attribute this minor performance gain to the insufficient quality of the generated triggers, as the annotated ground-truth triggers in the dataset for training are highly scarce. Therefore, we conducted additional comparison experiments to analyze this assumption in detail by changing the type of PLM and inserting an additional input feature.

Tables 4 and 5 show the efficacy comparison of the generated triggers according to the type of trigger generation models. We compared two typical generative pre-trained language models (PLMs) with encoder-decoder architecture, T5 and BART [38]. Specifically, T5-large and BART-large models are adopted for the trigger generation module, and the fine-tuned models with the generated triggers by each model are shown as *GenTr* (T5) and *GenTr* (BART), respectively. In the full-shot setting, *GenTr* (T5) outperformed *GenTr* (BART) by approximately 0.5%p both at F1 and $F1_c$ scores in the development set, but in the test set, *GenTr* (BART) demonstrated approximately 0.3%p higher scores. Moreover, the *GenTr* (T5) model showed higher overall performance improvements than the *GenTr*

(BART) in the few-shot setting except for the $F1_c$ score when K is 32. In regard to these results, we assume that the T5 model more effectively handled the problem of lack of triggers to learn due to its large parameter size than the BART model, but we found that this was not a determinant factor.

Table 4. This table shows the comparison in performance based on the generative model type of generated triggers. BART and T5 models were used for the trigger generation task. *GenTr* (w/rel) model refers to a case where relation class information is provided as an input feature when generating a trigger. The values in parentheses indicate the change in performance compared to the baseline model. The * mark indicates the re-implemented version.

Full-Shot Setting				
Method	**Dev**		**Test**	
	F1	**$F1_c$**	**F1**	**$F1_c$**
$BERT_s$ *	61.81	56.86	59.94	55.29
+ *GenTr* (T5)	63.07 (+1.26)	57.60 (+0.74)	59.80 (−0.14)	55.48 (+0.19)
+ *GenTr* (BART)	62.58 (+0.77)	57.07 (+0.21)	60.16 (+0.22)	55.83 (+0.54)
+ *GenTr* (w/rel)	75.21 (+13.4)	67.47 (+10.61)	73.36 (+13.42)	65.63 (+10.34)

Table 5. This table demonstrates the performance comparison between T5 and BART models in the few-shot setting. K indicates the number of shots, and results from the test set are provided. The values in parentheses indicate the change in performance compared to the baseline model. The * mark indicates the re-implemented version.

Few-Shot Setting						
Method	**Shot**					
	K = 8		**K = 16**		**K = 32**	
	F1	**$F1_c$**	**F1**	**$F1_c$**	**F1**	**$F1_c$**
$BERT_s$*	27.67	26.49	42.84	40.11	48.71	45.55
+ *GenTr* (T5)	31.71 (+4.04)	30.47 (+3.98)	44.83 (+1.99)	41.46 (+1.35)	47.87 (+0.84)	43.99 (−1.56)
+ *GenTr* (BART)	28.69 (+1.02)	29.54 (+3.05)	43.64 (+0.80)	40.94 (+0.83)	48.73 (+0.02)	45.07 (−0.48)

In addition, *GenTr* (w/rel) in the full-shot setting indicates a model where the triggers were generated by providing the relation class r as an additional training input feature, and the input for this model is constructed as "<s> D </s> E </s> r </s>". Utilizing the triggers generated in this way led to an exponential increase in the relation prediction performances, achieving performance improvements of more than 10%p at F1 and $F1_c$ scores in both data splits. This result confirms the significance of providing appropriately generated triggers to the model as demonstrated in the previous paper [9]. Table 6 shows an example of the comparison between the generated triggers by the three model types, i.e., *GenTr* (T5), *GenTr* (BART) and *GenTr* (w/rel). In the case of the first relation (R1), all three models generated "boyfriend" as a trigger, whereas only the *GenTr* (w/rel) model correctly generated a trigger "love", for the second relation (R2). In addition to the given examples in the table, there are several cases in which only the model with the triggers generated using relation classes as an additional input feature correctly predicted triggers, such as a trigger "husband" for the relation "per:spouse". In other words, redundant phrases.

Thus, with regard to the trigger generation method, we conclude that simply providing a dialogue and an entity pair as input features for generation is insufficient to guide the generative model effectively on the critical contextual information due to the scarcity of the annotated triggers. Moreover, to play a decisive role, triggers should be generated by supplementing the input features with another complement with informational importance comparable to the relation class r. Based on this perspective, discovering the additional

significant features for improving the trigger generation procedure even without relation classes will be our future work.

Table 6. This table shows an example of the generated triggers by three model types (*GenTr* (T5), *GenTr* (BART) and *GenTr* (w/rel)) in the DialogRE development set. *GenTr* (w/rel) is a model where relation class information is provided as an input feature when generating a trigger. The ground-truth triggers (GT Trigger) and the generated triggers (*GenTr*) by the models are bolded. R1 (R2) indicates the given relational information.

Dialogue						
1	Speaker 1: So, um . . . I'm proposing to Phoebe tonight.					
2	Speaker 2: Tonight?! Isn't an engagement ring supposed to have a diamond? Oh, there it is!					
3	Speaker 1: Yeah, well, being a failed scientist doesn't pay quite as well as you might think. That's um . . . one seventieth of a karat. And the clarity is um . . . is quite poor.					
	(· · · · ·)					
14	Speaker 3: Ok, my husband just gave your **boyfriend** some very bad advice. Look, David is going to propose to you tonight.					
15	Speaker 4: Wow? Really? That's fantastic!					
16	Speaker 3: What are you serious? You wanna marry him? Wha . . . What about Mike?					
17	Speaker 4: Oh, ok, you want me to marry Mike? Alright, well, let's just gag him and handcuff him and force him down the aisle. I can just see it: "Mike, do you take Phoebe . . . " You know, it's every girl's dream!					
18	Speaker 3: Do you really think marrying someone else is the right answer?					
19	Speaker 4: Sure! Look, ok, bottom line: I **love** Mike . . . David! David. I love David. Don't look at me that way, Roseanne Roseannadanna!					
	Argument pair	**GT Trigger**	**Generated Triggers**			**Relation Type**
			GenTr (T5)	*GenTr* (BART)	*GenTr* (w/rel)	
R1	(Speaker 1, Speaker 4)	boyfriend	boyfriend	boyfriend	boyfriend	per:girl/boyfriend
R2	(Speaker 4, Mike)	love	boyfriend	boyfriend	love	per:positive_impression

6. Conclusions

This paper explored simple yet effective methods in dialogue relation extraction by introducing prompt-based fine-tuning and the trigger generation approach. Also, their effectiveness is analyzed with additional experiments. In particular, unlike the previous extractive approach, we adopted the generative approach for the trigger generation module and compared the efficacy of the generated triggers between representative generative pre-trained language models (PLMs), i.e., BART and T5. The generated triggers have shown more significant effects in the few-shot setting compared to the full-shot setting, specifically when the shot K is 8. However, due to the insufficiency of ground-truth triggers for training, there still are points to improve the trigger generation module in the future. In addition, the prompt-based approach, including the prompt manual initialization method, which considers the prior distributional knowledge, demonstrated its effectiveness, showing significant performance improvements compared to the baseline model.

To summarize, this study aimed to directly exploit the model's implicit knowledge in the dialogue relation extraction task through the utilization of a trigger-generation method and a prompt-based approach and guide the model to clues about relational facts. To this end, attempts were conducted to utilize generative models, add soft prompts, and deliberately initialize inserted prompts. Furthermore, motivated by the observations, it is expected to improve task performance by utilizing more diverse generative models for enriching the quality of generated triggers in future work.

Author Contributions: Conceptualization, J.K. and J.S.; methodology, J.K.; software, J.K., G.K. and J.S.; validation, J.K.; formal analysis, J.K.; investigation, J.K. and G.K.; resources, J.K. and J.S.; data curation, J.S.; writing—original draft preparation/review and editing, J.K., G.K. and J.S.; visualization, J.K. and G.K.; supervision/project administration/funding acquisition, H.L.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ICT Creative Consilience program (IITP-2023-2020-0-01819) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation) and under the ITRC (Information Technology Research Center) support program (IITP-2022-2018-0-01405) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation). Additionally, it was also supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00368, A Neural-Symbolic Model for Knowledge Acquisition and Inference Techniques).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: A publicly available dataset was utilized in this study. These data can be found here: "https://github.com/nlpdata/dialogre" (accessed on 23 June 2023).

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

References

1. Ji, H.; Grishman, R.; Dang, H.T.; Griffitt, K.; Ellis, J. Overview of the TAC 2010 knowledge base population track. In Proceedings of the Third Text Analysis Conference (TAC 2010), Gaithersburg, MD, USA, 15–16 November 2010; Volume 3, p. 3.
2. Socher, R.; Huval, B.; Manning, C.D.; Ng, A.Y. Semantic Compositionality through Recursive Matrix-Vector Spaces. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Jeju Island, Republic of Korea, 12–14 July 2012; pp. 1201–1211.
3. Lin, Y.; Shen, S.; Liu, Z.; Luan, H.; Sun, M. Neural relation extraction with selective attention over instances. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 7–12 August 2016; pp. 2124–2133.
4. Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J. Relation classification via convolutional deep neural network. In Proceedings of the COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 23–29 August 2014; pp. 2335–2344.
5. Swampillai, K.; Stevenson, M. Inter-sentential relations in information extraction corpora. In Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10), Valletta, Malta, 17–23 May 2010.
6. Peng, N.; Poon, H.; Quirk, C.; Toutanova, K.; Yih, W.t. Cross-Sentence N-ary Relation Extraction with Graph LSTMs. *Trans. Assoc. Comput. Linguist.* **2017**, *5*, 101–115. [CrossRef]
7. Han, X.; Wang, L. A Novel Document-Level Relation Extraction Method Based on BERT and Entity Information. *IEEE Access* **2020**, *8*, 96912–96919. [CrossRef]
8. Jia, Q.; Huang, H.; Zhu, K.Q. DDRel: A New Dataset for Interpersonal Relation Classification in Dyadic Dialogues. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 13125–13133. [CrossRef]
9. Yu, D.; Sun, K.; Cardie, C.; Yu, D. Dialogue-Based Relation Extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 4927–4940. [CrossRef]
10. Han, X.; Zhao, W.; Ding, N.; Liu, Z.; Sun, M. Ptr: Prompt tuning with rules for text classification. *AI Open* **2022**, *3*, 182–192. [CrossRef]
11. 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 (Volume 1: Long Papers), Online, 1–6 August 2021; pp. 3816–3830. [CrossRef]
12. 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, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]
13. Hur, Y.; Son, S.; Shim, M.; Lim, J.; Lim, H. K-EPIC: Entity-Perceived Context Representation in Korean Relation Extraction. *Appl. Sci.* **2021**, *11*, 11472. [CrossRef]
14. Qin, P.; Xu, W.; Wang, W.Y. Dsgan: Generative adversarial training for distant supervision relation extraction. *arXiv* **2018**, arXiv:1805.09929.
15. Ji, F.; Qiu, X.; Huang, X.J. Detecting hedge cues and their scopes with average perceptron. In Proceedings of the Fourteenth Conference on Computational Natural Language Learning–Shared Task, Uppsala, Sweden, 15–16 July 2010; pp. 32–39.
16. Zapirain, B.; Agirre, E.; Marquez, L.; Surdeanu, M. Selectional preferences for semantic role classification. *Comput. Linguist.* **2013**, *39*, 631–663. [CrossRef]
17. Elsahar, H.; Vougiouklis, P.; Remaci, A.; Gravier, C.; Hare, J.; Laforest, F.; Simperl, E. T-rex: A large scale alignment of natural language with knowledge base triples. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan, 7–12 May 2018.

18. Yao, Y.; Ye, D.; Li, P.; Han, X.; Lin, Y.; Liu, Z.; Liu, Z.; Huang, L.; Zhou, J.; Sun, M. DocRED: A large-scale document-level relation extraction dataset. *arXiv* **2019**, arXiv:1906.06127.
19. Mesquita, F.; Cannaviccio, M.; Schmidek, J.; Mirza, P.; Barbosa, D. Knowledgenet: A benchmark dataset for knowledge base population. 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), Hong Kong, China, 3–7 November 2019; pp. 749–758.
20. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
21. Xue, F.; Sun, A.; Zhang, H.; Chng, E.S. Gdpnet: Refining latent multi-view graph for relation extraction. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 14194–14202.
22. Long, X.; Niu, S.; Li, Y. Consistent Inference for Dialogue Relation Extraction. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, Montreal, Canada, 19–27 August 2021; Zhou, Z.H., Ed.; pp. 3885–3891. [CrossRef]
23. Lee, B.; Choi, Y.S. Graph Based Network with Contextualized Representations of Turns in Dialogue. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, Online and Punta Cana, Dominican Republic, 7–11 November 2021; pp. 443–455. [CrossRef]
24. Chen, H.; Hong, P.; Han, W.; Majumder, N.; Poria, S. Dialogue relation extraction with document-level heterogeneous graph attention networks. *Cogn. Comput.* **2023**, *15*, 793–802. [CrossRef]
25. Duan, G.; Dong, Y.; Miao, J.; Huang, T. Position-Aware Attention Mechanism–Based Bi-graph for Dialogue Relation Extraction. *Cogn. Comput.* **2023**, *15*, 359–372. [CrossRef]
26. Liu, X.; Zhang, J.; Zhang, H.; Xue, F.; You, Y. Hierarchical Dialogue Understanding with Special Tokens and Turn-level Attention. *arXiv* **2023**, arXiv:2305.00262.
27. Schick, T.; Schütze, H. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv* **2020**, arXiv:2001.07676.
28. Li, X.L.; Liang, P. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv* **2021**, arXiv:2101.00190.
29. Liu, X.; Ji, K.; Fu, Y.; Tam, W.; Du, Z.; Yang, Z.; Tang, J. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Dublin, Ireland, 22–27 May 2022; pp. 61–68.
30. Zhang, S.; Khan, S.; Shen, Z.; Naseer, M.; Chen, G.; Khan, F.S. Promptcal: Contrastive affinity learning via auxiliary prompts for generalized novel category discovery. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, Canada, 8–22 June 2023; pp. 3479–3488.
31. He, K.; Mao, R.; Huang, Y.; Gong, T.; Li, C.; Cambria, E. Template-Free Prompting for Few-Shot Named Entity Recognition via Semantic-Enhanced Contrastive Learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, 1–13. . [CrossRef] [PubMed]
32. Yang, K.; Liu, D.; Lei, W.; Yang, B.; Xue, M.; Chen, B.; Xie, J. Tailor: A soft-prompt-based approach to attribute-based controlled text generation. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Toronto, Canada, 9–14 July 2023; pp. 410–427.
33. Kumar, S.; Talukdar, P. NILE: Natural Language Inference with Faithful Natural Language Explanations. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, 5–10 July 2020; pp. 8730–8742. [CrossRef]
34. Liu, H.; Yin, Q.; Wang, W.Y. Towards Explainable NLP: A Generative Explanation Framework for Text Classification. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 5570–5581. [CrossRef]
35. Ormandi, R.; Saleh, M.; Winter, E.; Rao, V. Webred: Effective pretraining and finetuning for relation extraction on the web. *arXiv* **2021**, arXiv:2102.09681.
36. Lin, P.W.; Su, S.Y.; Chen, Y.N. TREND: Trigger-Enhanced Relation-Extraction Network for Dialogues. *arXiv* **2021**, arXiv:2108.13811.
37. An, H.; Chen, D.; Xu, W.; Zhu, Z.; Zou, Y. TLAG: An Informative Trigger and Label-Aware Knowledge Guided Model for Dialogue-based Relation Extraction. In Proceedings of the 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD), Rio de Janeiro, Brazil, 24–26 May 2023; IEEE: New York, NY, USA, 2023; pp. 59–64.
38. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. 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, Online, 5–10 July 2020; pp. 7871–7880. [CrossRef]
39. Son, J.; Kim, J.; Lim, J.; Lim, H. GRASP: Guiding model with RelAtional Semantics using Prompt. *arXiv* **2022**, arXiv:2208.12494.
40. 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: ACM Web Conference 2022, Lyon, France, 25–29 April 2022; Association for Computing Machinery: New York, NY, USA, 2022; pp. 2778–2788. [CrossRef]
41. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 5485–5551.
42. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. *arXiv* **2017**, arXiv:1711.05101.