Making LLMs as Fine-Grained Relation Extraction Data Augmentor

Yifan Zheng[1,2] , **Wenjun Ke**[1,3,*] , **Qi Liu**[2] , **Yuting Yang**[2]
Ruizhuo Zhao[2] , **Dacheng Feng**[2] , **Jianwei Zhang**[2,4] and **Zhi Fang**[2,4]
[1]School of Computer Science and Engineering, Southeast University
[2]Beijing Institute of Computer Technology and Application
[3]Key Laboratory of New Generation Artifcial Intelligence Technology and Its Interdisciplinary Applications (Southeast University)
[4]Laboratory for Big Data and Decision National University of Defense Technology
zhengyifan_ht@163.com, kewenjun@seu.edu.cn liuqi970811@163.com, yyttina@126.com,
{ruizhuozhao, fengdacheng_ht, zhangjianwei_ht, fangzhi_ht}@163.com

Abstract

Relation Extraction (RE) identifies relations between entities in text, typically relying on supervised models that demand abundant high-quality data. Various approaches, including Data Augmentation (DA), have been proposed as promising solutions for addressing low-resource challenges in RE. However, existing DA methods in RE often struggle to ensure consistency and contextual diversity in generated data due to the fine-grained nature of RE. Inspired by the extensive generative capabilities of large language models (LLMs), we introduce a novel framework named ConsistRE, aiming to maintain context consistency in RE. ConsistRE initiates by collecting a substantial corpus from external resources and employing statistical algorithms and semantics to identify keyword hints closely related to relation instances. These keyword hints are subsequently integrated as contextual constraints in sentence generation, ensuring the preservation of relation dependence and diversity with LLMs. Additionally, we implement syntactic dependency selection to enhance the syntactic structure of the generated sentences. Experimental results from the evaluation of SemEval, TACRED, and TACREV datasets unequivocally demonstrate that ConsistRE outperforms other baselines in F1 values by 1.76%, 3.92%, and 2.53%, respectively, particularly when operating under low-resource experimental conditions.

1 Introduction

Relation Extraction (RE) is pivotal in Information Extraction (IE), seeking to identify relations between entities within textual data. Its significance resonates in downstream applications like event extraction [Xiang and Wang, 2019], knowledge graph [Luan *et al.*, 2018], and intelligent question answering [Sun *et al.*, 2021]. Despite the commendable success of current methodologies, which predominantly follow a supervised paradigm, a notable reliance exists on extensive datasets with high-quality annotations. In practical scenarios, the primary hurdles confronting RE revolve around low-resource challenges. These include the relatively modest size of available datasets, restricted application field scopes, and the complexities associated with labeling special domains.

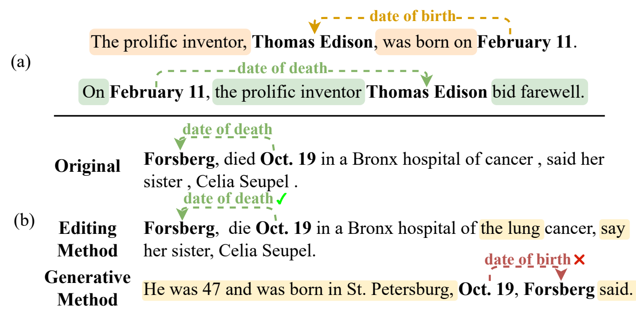

Figure 1: (a) Highlights the contextual variability in relation determination, emphasizing that an entity pair can manifest distinct relations within different context. (b) Compares mainstream DA methods in RE, revealing that the Editing Method preserves original relation dependencies at the expense of sentence diversity. At the same time, the Generative Method excels in contextual richness but may introduce deviations in dependencies of relation.

Numerous approaches have been proposed to address the challenges mentioned above, including meta-learning [Hu *et al.*, 2021; Liu *et al.*, 2022; Pouran Ben Veyseh *et al.*, 2023], transfer learning [Sarhan and Spruit, 2020; Gururaja *et al.*, 2023], data augmentation [Hu *et al.*, 2023; Zhao *et al.*, 2023; Xu *et al.*, 2023] and instruction prompting [Li *et al.*, 2023]. Among these, Data Augmentation (DA) stands out as a plug-and-play technology, offering direct applicability as a preprocessing method for a broad spectrum of tasks. While DA techniques have found success in tasks like Text Classification (TC) [Hsu *et al.*, 2021] and Named Entity Recognition (NER) [Ke *et al.*, 2023], their exploration in RE remains somewhat limited. This disparity arises due to the inherent fine-grained nature of RE compared to TC and NER. Modeling the intricate dependencies within RE proves challenging. As illustrated in Figure 1(a), the presence of the same en-

*Corresponding authors.

tity pair in a sentence may result in entirely different relation types due to variations in context.

Dominant methods frame fine-grained DA into controlled text generation paradigm [Ke *et al.*, 2023; Hu *et al.*, 2023]. Fine-grained DA is broadly categorized into two paradigms: editing and generative methods. Editing methods involve simple transform operations like random exchange, insertion, and deletion. However, the imposition of rule restrictions limits the diversity of samples, consequently diminishing the generalization capacity of the RE models. For instance, as illustrated in the first instance in Figure 1(b), merely substituting the *said* for *say* and adding *the lung* fails to introduce substantial contextual diversity. Generative methods offer the advantage of producing more fluid and diverse samples. However, current generative approaches exhibit two notable shortcomings. Firstly, compared to the original sentence, the generated counterpart may deviate semantically, failing to preserve the relation dependency between the original entity pairs. As exemplified in Figure 1(b), owing to variations in contextual semantics, the relation type between entities *Forsberg* and *Oct.19* transitions from *date_of_death* to *date_of_birth*. Secondly, existing methods lack specific hard constraints to ensure the inclusion of entity pairs during sentence generation. This oversight may introduce new entity pairs with unknown labels, leading to the generation of uncontrollable data. Consequently, when employing controlled text generation for RE DA, it becomes imperative to address the challenge of enhancing context diversity beyond entity pairs while preserving relation dependencies.

We argue that the crux of RE DA lies in preserving relation dependencies between pairs of entities through semantic consistency within the context. At the same time, to enhance the generalization ability of RE models, it is also necessary to ensure the diversity of contextual expressions during the generation process. Compared with existing pre-trained language models (PLMs) such as T5 [Raffel *et al.*, 2020] and BART [Lewis *et al.*, 2020], large language models (LLMs) such as GPT-3 [Brown *et al.*, 2020], LLaMA [Touvron *et al.*, 2023] and GPT-4 [OpenAI, 2024] show strong potential in generating diverse and contextually relevant texts, bringing new possibilities to RE DA. This paper proposes ConsistRE, an innovative RE DA method that maintains context consistency in RE. This method adds context constraints of keyword hints in the sentence synthesis process to ensure that the generated sentences maintain relation dependencies and semantic consistency while increasing the diversity of synthesized sentences with LLMs. Specifically, first, we apply statistical algorithms and semantic similarity to find the keyword hints most closely related to the relation instances based on a large amount of textual data. Following this, triples and keyword hints are included as controlled text as part of the prompt. During the sentence generation process, we filtered similar instances from both original and synthetic samples as demonstrations to enhance the performance of the LLMs. Finally, we select sentences that align more consistently with grammatical rules through syntactic dependency parsing to ensure that the generated sentences are more grammatically sound.

We assess the performance of our RE DA method on two RE models, ReDMP and SuRE, using three datasets: SemEval, TACRED, and TCAREV. The experimental results underscore the remarkable effectiveness of our approach in enhancing the diversity of generated sentences while preserving relation dependencies. When applied to ReDMP, ConsistRE exhibits superior performance, achieving F1 values of 1.48%, 5.48%, and 3.16% higher than other optimal methods on SemEval, TACRED, and TACREV, respectively. Similarly, under SuRE, ConsistRE outperforms other methods, yielding F1 values higher by 2.03%, 2.35%, and 1.9%. To sum up, the contributions of this paper are three-fold:

- We argue that the cornerstone of RE DA lies in maintaining the relation dependency of synthetic sentences through semantic consistency with context.
- We introduce ConsistRE, a framework that aims to simultaneously maintain the consistency of dependencies and diversity of synthetic sentences with LLMs.
- We conduct extensive experiments on three public datasets, demonstrating the importance of maintaining relation dependencies through contextual constraints.

2 Methods

Assuming that a relation instance (s, h, r, t) is given from the original annotated dataset X, where s, h, r, t represent the source sentence, head entity, relation type, and tail entity, respectively. ConsistRE aims to derive a substantially larger augmented dataset Y that maintains high consistency with X. For each instance $(\tilde{s}, h, r, t) \in Y$, $\tilde{s}$ is newly generated from s, while maintaining the original (h, r, t) unchanged.

The workflow of ConsistRE is illustrated in Figure 2. In the first stage, ConsistRE gathers a substantial amount of sentences related to triplet (h, r, t) from the Internet and acquires the keyword hints k most intricately associated with (s, h, r, t) utilizing statistical algorithms and semantic similarity. Moving on to the second stage, ConsistRE employs langchain[1] to select the most semantically similar instance as demonstrations d from the constructed example selector. Subsequently, d, (s, h, r, t), and k are integrated into a prompt template to generate prompts, and an LLM is employed to generate a set of sentence instances. Finally, in the third stage, syntactic dependency parsing is employed to select instances $\tilde{s}$ with superior syntax, forming the augmented dataset Y.

2.1 Keyword Hints Retrieval

The initial stage of our approach involves acquiring the most pertinent keyword hints k. Here, k represents the context most closely related to the relation instance (s, h, r, t) and will later be used as a hard constraint during the sentence generation, aiming to maintain the dependency consistency of the relation in the generated sentences.

Related Sentences Retrieval

Given the intricate nature of RE that demands fine-grained modeling, the identification of relations between specified entities necessitates comprehensive and contextually rich support. Relying solely on contextual information derived from the original sentence s might prove insufficient in capturing

[1] https://www.langchain.com/

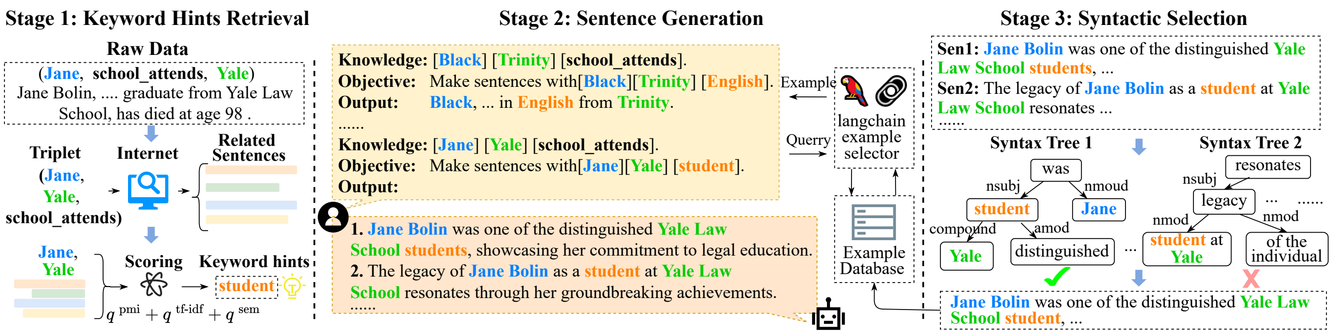

Figure 2: Overview of ConsistRE: 1) Applying statistical algorithms and semantic similarity to find keyword hints related to relation instances in textual data; 2) Incorporating triplet and keyword hints in prompts and selecting similar instances as demonstrations; 3) Ensuring grammatical correctness through syntactic dependency parsing.

the intricacies of the relations. Therefore, it becomes imperative to seek additional sentences with more extensive content to address this limitation. To mitigate this challenge, we augment our dataset by collecting substantial textual data from the Internet. Specifically, we utilize the search interface provided by Google[2] to gather a substantial set of sentences relevant to the triplet (h, r, t). The acquired sentences undergo preprocessing to extract pure text, forming the sentence set $\mathcal{C}$ for subsequent utilization in obtaining k.

Keyword Hints Obtain

We formulate a relevance score to discern keyword hints k that most effectively encapsulate entity relations within the retrieved sentence set $\mathcal{C}$. Specifically, the relevance score q assigned to each occurring word w is defined as follows:

$$q = q^{\text{pmi}} + q^{\text{tf-idf}} + q^{\text{sem}} \tag{1}$$

$$q^{\text{pmi}} = \log\left(\frac{P(w,h,t)}{P(w)\cdot P(h)\cdot P(t)}\right) \tag{2}$$

$$q^{\text{tf-idf}} = \frac{\sum_c \text{TF}(w,c) \times \text{IDF}(w,c,\mathcal{C})}{|\mathcal{C}|} \tag{3}$$

$$q^{\text{sem}} = \cos\left(\text{EMB}(w), \text{EMB}(s)\right) \tag{4}$$

q^{pmi} represents the score computed by the Pointwise Mutual Information (PMI) [Church and Hanks, 1989], which is a widely used linguistic statistical method to gauge word correlation. $P(w)$, $P(h)$, and $P(t)$ respectively represent the probability of the calculated word w, head entity h, and tail entity t appearing in the sentences set $\mathcal{C}$. $P(w,h,t)$ represents the probability of all three appearing simultaneously.

$q^{\text{tf-idf}}$ represents the score calculated by the TF-IDF. The integration of TF-IDF aims to eliminate frequently occurring but semantically insignificant words. TF(w,c) represents the frequency of w in sentence $c \in \mathcal{C}$, while IDF$(w,c,\mathcal{C})$ represents the rarity of w in the sentence set $\mathcal{C}$.

q^{sem} represents the semantic similarity between s and w, which is employed to ensure that the k aligns closely with the semantics of the original sentence. EMB(s) and EMB(w) are encoded by Sentence-BERT [Reimers and Gurevych, 2019]. q^{pmi}, $q^{\text{tf-idf}}$ and q^{sem} are adjusted to range 0 to 1. By computing the relevance score q for each word, we select the w with the highest score as keyword hints k.

2.2 Sentence Generation

LLMs exhibit robust contextual learning capabilities and can be significantly augmented through few-shot in-context demonstrations. In the second stage, we aim to generate a set of high-quality sentences $\hat{s}$. We break down prompt acquisition into the following two steps: demonstration selection and prompt formulation.

Demonstration Selection

To better stimulate and leverage the In-Context Learning (ICL) capabilities of LLMs, choosing similar relation instances from the example database to form the demonstration d in the few-shot prompt is essential. We employ the example selector in langchain to execute these steps, utilizing Sentence-BERT as the encoding model and FAISS[3] as the embedding database. Example database is initialized with original dataset X, and subsequent augmented data $(\tilde{s}, h, r, t)$ is added during the execution process. Iteratively increasing the number of examples in the example database can expand the optional range of demonstrations. We select three examples from the example database that are semantically closest to (s, h, r, t) as demonstrations. The format of the demonstration is as follows:

Knowledge: The relation between [head entity] and [tail entity] is [relation type]

Objective: Make sentences with given entities [head entity], [tail entity] and keyword [keyword hint]

Output: [source sentence]

Deserving a special mention, *[keyword hint]* in the demonstration is extracted from the source sentence using TopicRank [Bougouin *et al.*, 2013].

Prompt Formulation

To enhance the context-learning accuracy of LLMs, we incorporate semi-formatted structural constraints into our prompt.

[2]https://developers.google.com/custom-search

[3]https://github.com/facebookresearch/faiss

Specifically, we input the relation instance (h, r, t) and keyword hints k into the task-prompt p. We combine the demonstration d selected in the preceding step and p sequentially in two steps to construct the prompt provided to the LLM to obtain the desired sentences $\hat{s}$ for each instance. ICL can be conceptualized as LLMs implicitly conducting Bayesian inference [Xie *et al.*, 2022]:

$$p(\hat{s}) = \int_d p(\hat{s}|d, p)p(d|p)d(d) \tag{5}$$

Given the prompt p and multiple demonstrations, LLMs learn via marginalization by "selecting" the demonstration.

Additionally, we do not include the original sentence s in the prompt to maintain the diversity of synthetic sentences. Task-prompt p is defined as follows:

Knowledge: The relation between [h] and [t] is [r]

Objective: Make sentences with given entities [h], [t] and keyword [k]

Output:

2.3 Syntactic Selection

For the sentence $\hat{s}$ generated in the preceding stage, we posit that when the syntactic structure of the generated sentence closely aligns with the sentence s, the generated result is more consistent with the original one. In pursuit of this, we introduce a similarity calculation method based on syntactic dependency structure to aid in selecting instances with superior syntax for the final augmented sentences.

In particular, for the original sentence s and each sentence $\hat{s}$ within the corresponding candidate set, we utilize Stanford Parse[4] to conduct syntactic analysis, resulting in the generation of the respective syntactic dependency trees, denoted as T_1 and T_2. The structure of syntactic dependency trees can encapsulate the inter-word dependency relations and convey syntactic structural information.

Following this, we employ the Tree Edit Distance (TDS) to gauge the similarity between two syntactic dependency trees. TDS is a method employed for measuring the similarity between two tree structures, quantifying the disparity between one tree and another by calculating the minimum number of edit operations necessary to transform one tree into the other. These edit operations encompass inserting, deleting, and replacing nodes. The formula for calculating TDS can be expressed as follows:

$$\begin{aligned} d\left(T_1, T_2\right) = \min\{ & d\left({T_1}', {T_2}'\right) + \delta\left(\text{sub}, n_1, n_2\right), \\ & d\left({T_1}', T_2\right) + \delta\left(\text{del}, n_1\right), d\left(T_1, {T_2}'\right) + \delta\left(\text{ins}, n_2\right)\} \end{aligned} \tag{6}$$

Among them, ${T_1}'$ and ${T_2}'$ represent the subtrees of T_1 and T_2, respectively, after the removal of the root node. n_1 and n_2 denote the root nodes of T_1 and T_2. In this case, the cost function $\delta(\cdot)$ for the three operations is uniformly defined as 1. The outlined issues can be efficiently addressed using dynamic programming [Zhang and Shasha, 1989]. Through the computation of TDS, we choose several sentences with the most favorable syntactic structure as the final augmented $\tilde{s}$, ensuring that the generated sentences exhibit sound syntactic structure and grammatical legitimacy.

[4]https://stanfordnlp.github.io/CoreNLP

3 Experiments

In this section, we describe the datasets used, outline the experimental settings, present the baselines, and provide the results of the experiments.

3.1 Datasets and Experimental Settings

We conduct our experiments on three public RE datasets: SemEval 2010 Task 8 (**SemEval**) [Hendrickx *et al.*, 2009], the TAC Relation Extraction Dataset (**TACRED**) [Zhang *et al.*, 2017], and the revisited TAC Relation Extraction Dataset (**TACREV**) [Alt *et al.*, 2020]. The statistics of datasets are presented in Table 1. **SemEval** is a traditional dataset widely employed in RE. It undergoes manual precision labeling and is devoid of noise. The SemEval dataset encompasses 19 relation types: Cause-Effect, Component-Whole, and others. **TACRED** is a more extensive dataset designed for RE. Its content primarily originates from news and online texts within the TAC KBP newswire and web forum corpus. Annotated through crowdsourcing, TACRED comprises 42 relation types. **TACREV** is a dataset derived from the original TACRED dataset. It addresses and rectifies some errors found in the annotated data within TACRED.

Dataset	#Rel	#Train						#Val	#Test
		Shot-5	Shot-10	Shot-20	Shot-50	Shot-100	All		
SemEval	19	91	181	361	876	1570	6507	1439	2717
TACRED	42	210	412	822	1904	3426	68124	22631	15509
TACREV	42	210	418	828	2309	3956	68124	22631	15509

Table 1: Statistics of our experimental datasets. *Shot-n* means sampling n instances from each relation type. For relation types with fewer than n instances, we sample all available data. *All* refers to the complete training dataset.

In our experimental setup, we sample 5, 10, 20, 50, and 100 instances for each relation type to simulate low-resource scenarios. Both ConsistRE and other baseline models augment the sampled data **3x** to ensure a fair experiment comparison. The augmented data, along with the initial sampled data, is then fed into the RE model for training. The remainder of the data remains unseen by all DA methods and RE models. In this study, Micro-F1 is chosen as a critical metric to assess and compare all DA methods. We adopt gpt-3.5-turbo[5] as the backbone model of ConsistRE, and each result is averaged over three runs for reporting.

3.2 Baselines

We choose the following two types of DA methods as baselines for comparison:

Editing methods: **WordNet Synonym Substitution (WSS)** [Mueller and Thyagarajan, 2016] introduces lexical variations by replacing selected tokens with synonyms from WordNet [Fellbaum, 1998]. **EDA** [Wei and Zou, 2019] proposes a set of token-level word operations. **Word Embedding Substitution (WES)** [Jiao *et al.*, 2020] enhances data diversity by substituting tokens with contextual word embeddings from BERT [Devlin *et al.*, 2019].

[5]https://openai.com/product

	Methods	SemEval						TACRED						TACREV					
		5	10	20	50	100	Avg.	5	10	20	50	100	Avg.	5	10	20	50	100	Avg.
ReDMP	Base	20.60	30.46	50.32	79.17	83.29	52.78	12.92	28.21	53.01	72.00	75.04	48.24	10.01	20.14	57.28	70.73	75.54	46.74
	WSS	25.07	36.13	61.92	78.14	83.19	56.89	16.84	47.76	62.42	71.35	71.83	54.04	18.14	46.49	**67.48**	70.03	72.84	50.00
	EDA	22.92	32.30	62.16	82.31	84.77	55.95	19.02	43.81	63.71	61.40	69.11	51.41	21.12	50.90	62.15	71.54	74.54	56.05
	WES	26.24	39.49	67.69	82.67	84.62	60.14	22.28	45.96	62.35	66.44	69.44	53.29	14.64	47.77	63.75	68.32	71.01	53.10
	REMix	25.22	33.14	61.35	81.91	84.47	57.22	17.37	48.82	64.51	72.23	74.01	55.39	19.09	56.08	65.37	**72.49**	74.63	57.53
	LAMBADA	24.99	31.11	43.93	64.23	71.85	47.22	26.28	48.96	55.81	61.94	60.01	50.60	16.60	51.95	58.92	63.34	64.89	51.14
	GDA	24.91	40.41	66.10	74.31	80.91	57.33	11.12	24.82	39.95	62.49	70.43	41.76	25.70	48.35	60.57	68.36	74.79	55.55
	ConsistRE	**28.79**	**43.08**	**68.02**	**82.84**	**85.36**	**61.62**	**32.06**	**52.49**	**68.32**	**74.11**	**77.37**	**60.87**	**28.76**	**59.68**	66.86	72.11	**76.06**	**60.69**
SuRE	Base	17.03	18.53	31.47	64.97	79.63	42.33	65.70	70.99	74.23	74.64	84.17	73.95	68.43	71.88	75.02	82.63	86.12	77.02
	WSS	32.51	48.67	65.24	76.70	82.42	61.11	70.95	73.25	74.47	81.42	83.54	76.73	71.46	76.01	77.86	84.70	85.87	79.18
	EDA	25.29	45.70	64.17	80.67	84.60	60.01	69.84	72.76	75.79	81.40	82.29	76.42	67.61	75.49	77.12	85.31	86.18	78.34
	WES	30.99	49.16	60.73	77.43	**83.88**	60.44	69.40	71.52	75.17	78.17	82.38	75.33	70.00	74.62	76.87	81.28	83.94	77.34
	REMix	24.88	43.70	58.59	**81.77**	83.79	58.55	66.84	70.98	74.25	77.00	83.17	74.45	70.55	75.42	78.42	83.56	86.84	78.96
	LAMBADA	23.51	49.95	58.93	68.54	76.09	55.40	69.89	72.17	74.23	76.29	78.81	74.28	68.92	73.60	76.87	78.70	81.74	75.97
	GDA	26.98	38.73	55.72	74.46	77.48	54.67	56.55	60.92	72.66	78.59	82.24	70.19	70.30	75.12	78.81	85.27	85.28	78.96
	ConsistRE	**35.99**	**51.84**	**66.93**	77.42	83.51	**63.14**	**74.60**	**75.05**	**77.77**	**83.06**	**84.90**	**79.08**	**72.95**	**77.47**	**80.27**	**86.56**	**88.16**	**81.08**

Table 2: Performance (Micro-F1 %) of different methods under Shot-{5, 10, 20, 50, 100} settings. The best results are in **bold** while second-best ones are underlined. Avg. denotes the average score.

Generative methods: **REMix** [Teru, 2023] applies lexically constrained decoding to back-translation. **LAMBADA** [Anaby-Tavor *et al.*, 2020] fine-tune GPT-2 and generate candidate examples. **GDA** [Hu *et al.*, 2023] employs two modules for model training: one ensures semantic coherence through reordering, while the other maintains grammatical structure with a unified pattern.

To ensure a fair comparison of each DA method's performance, we employ the following two RE models as evaluation benchmarks: **ReDMP** [Tian *et al.*, 2022] enhances performance by incorporating syntactic information through a syntax-induced encoder trained on auto-parsed data with dependency masking. **SuRE** [Lu *et al.*, 2022] transforms relation extraction into a summarization task, improving precision and efficiency through indirect supervision, sentence and relation conversion techniques, and constraint decoding for robust inference.

3.3 Main Results

The experimental results on the three datasets are presented in Table 2. **Base** uses only the sampled original data from the training dataset without additional operations.

In general, most baselines outperform the non-augmentation method (**Base**), highlighting the effectiveness of DA methods. With fewer sampled data (Shot-5, 10, and 20), DA methods consistently exhibit more significant performance improvements. However, under the experimental settings of Shot-50 and 100, the performance improvement is limited, and there is even a decline in performance.

Intuitively, generative methods are expected to outperform editing methods. However, in the context of our experimental setup, generative methods (LAMBADA on SemEval/TACREV, GDA on TACRED) exhibit noticeably poorer performance, even falling below **Base**. This can be attributed to generative methods needing to be adequately trained when the sample size is minimal (<15% on SemEval and <5% on TACRED/TACREV). In contrast, editing methods, being more straightforward and not reliant on extensive training data, achieve more promising results.

Across the three datasets and two evaluation models, our method consistently outperforms all other baseline methods on average without negative improvement in all sampling settings. Specifically, when tested with ReDMP, ConsistRE demonstrates F1 values that are 1.48%, 5.48%, and 3.16% higher than those of other optimal methods on SemEval, TACRED, and TACREV, respectively. Testing with ReDMP, F1 values of ConsistRE are higher by 2.03%, 2.35%, and 1.9% in three datasets, respectively. These results unequivocally showcase the superior adaptability of our method in generating a more significant number of new samples. This underscores the importance of emphasizing consistency and diversity of expression in the context.

3.4 Ablation Study

Our approach aims to generate augmented samples with consistent relation dependencies and diverse expressions by utilizing keyword hints. To assess the effectiveness of the components, we conduct ablation experiments on SemEval focusing on three aspects. Table 3 presents the results, where *w/o keywords* signifies that no keyword hints are added as restricted context during the sentence generation, *w/o langchain* refers to using a fixed example for demonstration, and *w/o syntactic* indicates the absence of syntactic selection.

The results reveal the positive significance of all three components for performance. Specifically, removing keyword hints leads to a significant performance decline on both ReDMP and SuRE, reaching 5.07% and 6.14%, respectively. Similarly, the removal of langchain and syntactic selection also caused a notable decline, with drops of 3.92% and 2.24% on ReDMP and 4.53% and 3.99% on SuRE. Notably, keyword hints have a pronounced impact on performance loss. This is because, without keyword hints, LLMs are prone to synthesizing sentences that deviate from semantics or fail to convey relation dependencies explicitly.

	Methods	SemEval						
		5	10	20	50	100	Avg.	↓%
ReDMP	ConsistRE	28.79	43.08	68.02	82.84	85.36	61.62	-
	w/o keywords	**26.52**	**38.19**	**59.71**	**77.94**	**80.39**	**56.55**	**5.07**
	w/o langchain	27.23	38.66	62.35	78.92	81.34	57.70	3.92
	w/o syntactic	27.86	42.50	65.79	78.92	81.81	59.38	2.24
SuRE	ConsistRE	35.99	51.84	66.93	77.42	83.51	63.14	-
	w/o keywords	**29.89**	**40.35**	**57.86**	76.32	80.56	**57.00**	**6.14**
	w/o langchain	30.90	41.15	65.62	**75.28**	**80.08**	58.61	4.53
	w/o syntactic	33.93	44.47	59.54	75.74	82.06	59.15	3.99

Table 3: Results of the main components ablation experiment, where ↓ represents the model's performance decline. Results with the most significant reduction are marked in **bold**.

3.5 Analysis Experiments

In this section, we perform experiments to assess the influence of keyword hints and the size of the generated data on the performance. Additionally, we evaluate the diversity of the generated samples.

Keyword Hints Selection Strategy

The ablation experiment effectively demonstrates the impact of adding keyword hints closely related to the relation instance during sentence generation. Separate experiments are conducted on the SemEval dataset to assess the contributions of three keyword hints selection strategies, with results presented in Table 4. Firstly, it can be observed that using each of the three strategies individually yields better results than not using keyword hints. When using PMI alone, there is a performance decrease of 1.68% and 2.93%, respectively. This is due to the introduction of partially semantically irrelevant keyword hints leading to a deviation in relation dependencies. Using TF-IDF and semantic similarity alone resulted in performance drops of 4.33% and 4.27% on ReDMP and 3.25% and 4.08% on SuRE. This is because these two strategies cannot identify the most representative keyword hints. In comparison, PMI contributes the most to our method.

	Methods	SemEval						
		5	10	20	50	100	Avg.	↓%
ReDMP	ConsistRE	28.79	43.08	68.02	82.84	85.36	61.62	-
	w/o keywords	**26.52**	**38.19**	**59.71**	77.94	**80.39**	**56.55**	**5.07**
	PMI only	27.23	41.92	65.77	81.18	83.60	59.94	1.68
	TF-IDF only	26.94	40.37	60.72	**76.98**	81.46	57.29	4.33
	semantic only	27.46	41.72	64.20	77.69	80.80	58.37	3.25
SuRE	ConsistRE	35.99	51.84	66.93	77.42	83.51	63.14	-
	w/o keywords	29.89	**40.35**	**57.86**	76.32	80.56	**57.00**	**6.14**
	PMI only	33.43	45.12	65.47	75.88	81.17	60.21	2.93
	TF-IDF only	32.53	46.25	62.97	**72.49**	80.14	58.87	4.27
	semantic only	**28.55**	44.65	65.62	76.39	**80.08**	59.06	4.08

Table 4: Evaluating the influence of keyword hints selection strategy via modifying the relevance score. Results with the most significant reduction are marked in **bold**.

Number of Keyword Hints

In this experiment, we investigate how the quantity of keyword hints influences on SemEval. The results, depicted in Figure 3, reveal surprisingly consistent trends across all sampling settings on both ReDMP and SuRE. Using only one keyword hints suffices to achieve optimal results in all cases. Increasing the number of keyword hints does not lead to performance improvement; instead, there is a varying degree of decline across all sampling settings, sometimes even lower than when no keyword hints are used. This is because in SemEval, short sentences are predominant, and an excessive number of keyword hints as hard constraints can limit the diversity of expressions.

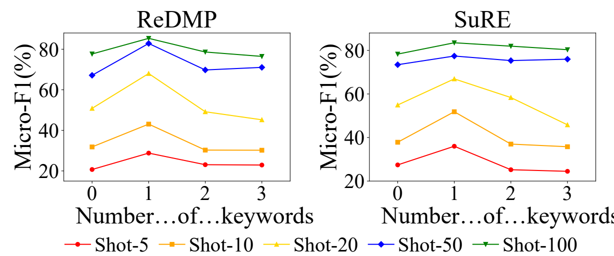

Figure 3: Performance under different keyword hints number.

Generated Data Size

In this experiment, we report the performance of the RE model by combining the sampled original sentences and generated sentences. How to determine the optimal expansion ratio of generated sentences is of great significance in data augmentation. Less generated sentences may not fulfill the purpose of data augmentation, while too many sentences can alter the distribution of the original sentences, resulting in performance degradation. So we investigate the impact of different sizes of generated data on model performance. We conduct experiments on two RE models with expansion ratios ranging from 1 to 6 under the Shot-20 sampling setting on the SemEval dataset. The results are presented in Figure 4.

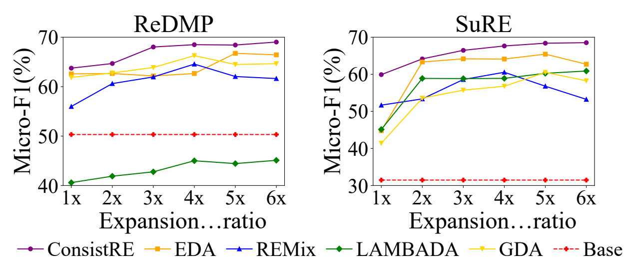

Figure 4: Performance under different expansion ratio.

Most data augmentation methods exhibit considerable performance improvements as the expansion ratio increases from 1 to 4. However, as the expansion ratio increases, the improvements gradually become smaller and level off. REMix and GDA experienced significant performance drops, indicating that an excess of enhanced data changes data distribution. Meanwhile, EDA shows more minor performance improvements when increasing the expansion ratio in most cases, possibly due to poorer diversity in data generation. Additionally, LAMBADA performs lower than **Base** on ReDMP, likely due to insufficient training data. Our method consistently performs best under all ratio settings, illustrating that our approach can maintain the distribution of sampled original sentences unchanged under keyword hints constraints while increasing generated sentence diversity.

Diversity Evaluation

To assess the diversity of synthetic sentences, we introduce the Distinct [Li *et al.*, 2016], which quantifies the number of distinct unigrams and bigrams divided by the total number of generated words. The calculation formula is as follows:

$$\text{Distinct}\,(N) = \frac{\text{Unique N-grams}}{\text{Total N-grams}} \times 100\% \tag{7}$$

We set N as 1 and 2, representing the proportion of unique words and bigrams, respectively. The scores under all sampling settings on the SemEval dataset are presented in Figure 5. Overall, generative methods (LAMBADA, GDA) exhibit better diversity than editing methods (WSS, EDA). Notably, our method consistently outperforms others in diversity across almost all settings, providing further evidence of the effectiveness of our approach in enhancing the diversity of synthetic sentences.

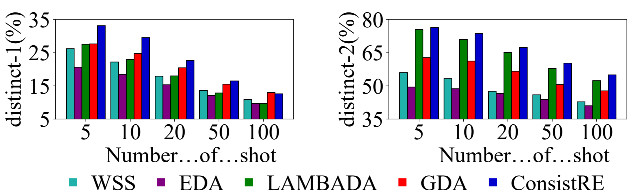

Figure 5: Diversity evaluation using Distinct.

3.6 Case Study

To provide further insight, we illustrate a case in Table 5. It can be observed that editing methods effectively maintain semantic consistency between sentences and their originals. However, simple replacement may struggle to enhance the diversity of samples, posing challenges to the generalization of RE models. Generative methods can synthesize sentences with significant contextual differences from the original ones, but they alter the distribution, leading to a biased dependency toward relation types like *org:member_of*. When the keyword hints *student* is not used, the relation between *Jane Bolin* and *Yale Law School* in the generated sentence is not explicitly stated, potentially introducing semantic bias. Adding keyword hints *student* to the prompt helps the LLM better focus on the relation between the two entities, ensuring that the generated sentence maintains the dependency between them.

4 Related Work

4.1 Data Augmentation

(1) For *editing methods*, Mueller and Thyagarajan [2016] generate additional training data by replacing selected tokens with synonyms from WordNet. Wei and Zou [2019] propose a set of token-level word operations for data augmentation, encompassing synonym replacement, random insertion, swap, and deletion. Jiao et al. [2020] utilize word embeddings to obtain augmented data. (2) For *generative methods*, Xie et al. [2020] and Fabbri et al. [2021] utilize back-translation on each sentence. Lowell et al. [2021] adopt a strategy of masking multiple words in a sentence and generating new sentences by filling these masks. Anaby-Tavor et al. [2020] fine-tune GPT-2 and generate candidate examples for each class. Hu et al. [2023] employ two complementary modules to train a model, one maintaining semantics through reordering and the other preserving grammatical structure through a unified pattern. However, editing methods cannot satisfy diversity, and generative methods cannot maintain relation consistency. Our method applies semantically consistent contextual constraints and leverages LLMs to generate sentences simultaneously satisfying relation dependency consistency and diversity.

Original	Sentence: Jane Bolin , who was the first black woman to graduate from Yale Law School and became america 's first black female judge , has died at age 98 . Relation: per:schools_attended
Editing Method	Jane Bolin, who was the first lightlessness woman to graduate from elihu Yale Law School, and became america's first lightlessness female judge, has died at age 98.
Generative Method	It came as a surprise to many Yale Law School staff members when jane bolin took over the chair in June.
ConsistRE w/o keyword	The legacy of Jane Bolin extends beyond her achievements at Yale Law School, as she left an indelible mark on the legal profession, inspiring future generations.
ConsistRE	Jane Bolin was one of the distinguished Yale Law School students, showcasing her commitment to legal education. (Keyword hint: student)

Table 5: Comparing the results of ConsistRE and other baselines, entities in both the original and generated sentences are highlighted.

4.2 LLMs for Low-resource RE

The rise of LLMs demonstrates the advance in low-resource RE. Li et al. [2023] propose the summarize-and-ask prompting, exploring the possibilities of LLMs in zero-shot RE. Wan et al. [2023] add task-aware representation to demonstration retrieval and enrich the demonstrations with gold label-induced reasoning logic. Wang et al. [2023] unify modeling of various IE tasks based on instruction tuning tasks and capturing inter-task dependencies. However, the efficiency of mapping inputs and labels with demonstrations needs to be improved to thoroughly express complex RE tasks [Deng *et al.*, 2023]; computing resources will also limit prompt-tuning LLMs. Therefore, it is more practical to use LLMs for data generation and then transfer it to the RE model.

5 Conclusion

This paper posits that the primary challenge in low-resource RE DA is ensuring the semantic consistency and contextual diversity of generated sentences. To address this, we propose a novel method named ConsistRE. ConsistRE incorporates keyword hints closely related to the relation instances as contextual constraints in sentence generation with LLMs and complements it with syntactic dependency selection. Experiments conducted on three public datasets under low-resource settings substantiate the effectiveness of our approach.

Acknowledgments

We thank the reviewers for their insightful comments. This work was supported by National Science Foundation of China (Grant Nos.62376057) and the Start-up Research Fund of Southeast University (RF1028623234). All opinions are of the authors and do not reflect the view of sponsors.

Contribution Statement

The contributions of Yifan Zheng and Wenjun Ke to this paper were equal.

References

[Alt *et al.*, 2020] Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. TACRED revisited: A thorough evaluation of the TACRED relation extraction task. In *Proc. of ACL*, 2020.

[Anaby-Tavor *et al.*, 2020] Ateret Anaby-Tavor, Boaz Carmeli, Esther Goldbraich, Amir Kantor, George Kour, Segev Shlomov, Naama Tepper, and Naama Zwerdling. Do not have enough data? deep learning to the rescue! In *Proc. of AAAI*, 2020.

[Bougouin *et al.*, 2013] Adrien Bougouin, Florian Boudin, and Béatrice Daille. TopicRank: Graph-based topic ranking for keyphrase extraction. In *Proc. of ACL*, 2013.

[Brown *et al.*, 2020] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Proc. of NeurIPS*, 2020.

[Church and Hanks, 1989] Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. In *Proc. of ACL*, 1989.

[Deng *et al.*, 2023] Shumin Deng, Yubo Ma, Ningyu Zhang, Yixin Cao, and Bryan Hooi. Information extraction in low-resource scenarios: Survey and perspective, 2023.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL*, 2019.

[Fabbri *et al.*, 2021] Alexander Fabbri, Simeng Han, Haoyuan Li, Haoran Li, Marjan Ghazvininejad, Shafiq Joty, Dragomir Radev, and Yashar Mehdad. Improving zero and few-shot abstractive summarization with intermediate fine-tuning and data augmentation. In *Proc. of NAACL*, 2021.

[Fellbaum, 1998] Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

[Gururaja *et al.*, 2023] Sireesh Gururaja, Ritam Dutt, Tinglong Liao, and Carolyn Rosé. Linguistic representations for fewer-shot relation extraction across domains. In *Proc. of ACL*, 2023.

[Hendrickx *et al.*, 2009] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the Workshop on Semantic Evaluations: Recent Achievements and Future Directions (SEW-2009)*, 2009.

[Hsu *et al.*, 2021] Ting-Wei Hsu, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. Semantics-preserved data augmentation for aspect-based sentiment analysis. In *Proc. of EMNLP*, 2021.

[Hu *et al.*, 2021] Xuming Hu, Chenwei Zhang, Fukun Ma, Chenyao Liu, Lijie Wen, and Philip S. Yu. Semi-supervised relation extraction via incremental meta self-training. In *Proc. of EMNLP Findings*, 2021.

[Hu *et al.*, 2023] Xuming Hu, Aiwei Liu, Zeqi Tan, Xin Zhang, Chenwei Zhang, Irwin King, and Philip S. Yu. GDA: Generative data augmentation techniques for relation extraction tasks. In *Proc. of ACL Findings*, 2023.

[Jiao *et al.*, 2020] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. TinyBERT: Distilling BERT for natural language understanding. In *Proc. of EMNLP Findings*, 2020.

[Ke *et al.*, 2023] Wenjun Ke, Zongkai Tian, Qi Liu, Peng Wang, Jinhua Gao, and Rui Qi. Towards incremental ner data augmentation via syntactic-aware insertion transformer. In *Proc. of IJCAI*, 2023.

[Lewis *et al.*, 2020] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proc. of ACL*, 2020.

[Li *et al.*, 2016] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *Proc. of NAACL*, 2016.

[Li *et al.*, 2023] Guozheng Li, Peng Wang, and Wenjun Ke. Revisiting large language models as zero-shot relation extractors. In *Proc. of EMNLP Findings*, 2023.

[Liu *et al.*, 2022] Fangchao Liu, Hongyu Lin, Xianpei Han, Boxi Cao, and Le Sun. Pre-training to match for unified low-shot relation extraction. In *Proc. of ACL*, 2022.

[Lowell *et al.*, 2021] David Lowell, Brian Howard, Zachary C. Lipton, and Byron Wallace. Unsupervised data augmentation with naive augmentation and without unlabeled data. In *Proc. of EMNLP*, 2021.

[Lu *et al.*, 2022] Keming Lu, I-Hung Hsu, Wenxuan Zhou, Mingyu Derek Ma, Muhao Chen, et al. Summarization as indirect supervision for relation extraction. In *EMNLP - Findings*, 2022.

[Luan *et al.*, 2018] Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proc. of EMNLP*, 2018.

[Mueller and Thyagarajan, 2016] Jonas Mueller and Aditya Thyagarajan. Siamese recurrent architectures for learning sentence similarity. In *Proc. of AAAI*, 2016.

[OpenAI, 2024] OpenAI. Gpt-4 technical report. https://openai.com/index/gpt-4-research, 2024. Accessed: 2024-01-18.

[Pouran Ben Veyseh *et al.*, 2023] Amir Pouran Ben Veyseh, Franck Dernoncourt, Bonan Min, and Thien Nguyen. Generating labeled data for relation extraction: A meta learning approach with joint GPT-2 training. In *Proc. of ACL Findings*, 2023.

[Raffel *et al.*, 2020] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 2020.

[Reimers and Gurevych, 2019] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proc. of EMNLP*, 2019.

[Sarhan and Spruit, 2020] Injy Sarhan and Marco Spruit. Can we survive without labelled data in nlp? transfer learning for open information extraction. *Applied Sciences*, 2020.

[Sun *et al.*, 2021] Haitian Sun, Patrick Verga, Bhuwan Dhingra, Ruslan Salakhutdinov, and William W. Cohen. Reasoning over virtual knowledge bases with open predicate relations. In *Proc. of ICML*, 2021.

[Teru, 2023] Komal Teru. Semi-supervised relation extraction via data augmentation and consistency-training. In *Proc. of EACL*, 2023.

[Tian *et al.*, 2022] Yuanhe Tian, Yan Song, and Fei Xia. Improving relation extraction through syntax-induced pre-training with dependency masking. In *Proc. of ACL Findings*, 2022.

[Touvron *et al.*, 2023] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[Wan *et al.*, 2023] Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. GPT-RE: In-context learning for relation extraction using large language models. In *Proc. of EMNLP*, 2023.

[Wang *et al.*, 2023] Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. Instructuie: Multi-task instruction tuning for unified information extraction. *ArXiv preprint*, 2023.

[Wei and Zou, 2019] Jason Wei and Kai Zou. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proc. of EMNLP*, 2019.

[Xiang and Wang, 2019] Wei Xiang and Bang Wang. A survey of event extraction from text. *IEEE Access*, 2019.

[Xie *et al.*, 2020] Qizhe Xie, Zihang Dai, Eduard H. Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *Proc. of NeurIPS*, 2020.

[Xie *et al.*, 2022] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. In *Proc. of ICLR*, 2022.

[Xu *et al.*, 2023] Benfeng Xu, Quan Wang, Yajuan Lyu, Dai Dai, Yongdong Zhang, and Zhendong Mao. S2ynRE: Two-stage self-training with synthetic data for low-resource relation extraction. In *Proc. of ACL*, 2023.

[Zhang and Shasha, 1989] Kaizhong Zhang and Dennis Shasha. Simple fast algorithms for the editing distance between trees and related problems. *SIAM Journal on Computing*, 1989.

[Zhang *et al.*, 2017] Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. Position-aware attention and supervised data improve slot filling. In *Proc. of EMNLP*, 2017.

[Zhao *et al.*, 2023] Jun Zhao, WenYu Zhan, Xin Zhao, Qi Zhang, Tao Gui, Zhongyu Wei, Junzhe Wang, Minlong Peng, and Mingming Sun. RE-matching: A fine-grained semantic matching method for zero-shot relation extraction. In *Proc. of ACL*, 2023.