

# Enhancing Relation Extraction via Supervised Rationale Verification and Feedback

Yongqi Li<sup>1</sup>, Xin Miao<sup>1</sup>, Shen Zhou<sup>1</sup>, Mayi Xu<sup>1</sup>, Yuyang Ren<sup>1,3</sup>, Tiejun Qian<sup>1,2\*</sup>

<sup>1</sup>School of Computer Science, Wuhan University, China

<sup>2</sup>Intellectual Computing Laboratory for Cultural Heritage, Wuhan University, China

<sup>3</sup>Research Institute of Nuclear Power Operation, China

{liyongqi, miaoxin, shenzhou, xumayi}@whu.edu.cn, renyy@cnpn.com.cn, qty@whu.edu.cn

## Abstract

Despite the rapid progress that existing automated feedback methods have made in correcting the output of large language models (LLMs), these methods cannot be well applied to the relation extraction (RE) task due to their designated feedback objectives and correction manner. To address this problem, we propose a novel automated feedback framework for RE, which presents a *rationale supervisor* to verify the rationale and provides *re-selected demonstrations as feedback* to correct the initial prediction. Specifically, we first design a causal intervention and observation method to collect biased/unbiased rationales for contrastive training the rationale supervisor. Then, we present a verification-feedback-correction procedure to iteratively enhance LLMs' capability of handling the RE task. Extensive experiments prove that our proposed framework significantly outperforms existing methods.

**Code** — <https://github.com/NLPGM/SRVF>

## Introduction

The relation extraction (RE) task aims to extract the semantic relation between entities in the text, which is an important task in information extraction. Unlike previous fine-tuning strategies based on small language models (Wu and He 2019), recent studies (Wan et al. 2023; Ma et al. 2023) leverage the strong instruction understanding abilities and rich intrinsic knowledge of large language models (LLMs) (Ouyang et al. 2022; Touvron et al. 2023; Bai et al. 2022) to enhance the performance of RE.

Despite their significant progress, LLM based methods may suffer from relation bias when performing relation extraction. For example, given a sentence “data is derived from a study”, where “data” and “study” form the “Entity-Origin” relation, LLMs may be influenced by the pre-trained knowledge and have the stereotype that “data is the product that someone produces”, thus making a biased relation prediction “Product-Producer”, which ignores that the real producer is investigators (producer of the study). Furthermore, existing LLM based RE methods focus on the pre-selection of in-context demonstrations (Wan et al. 2023; Ma, Li, and Zhang 2023) or instruction design (Zhang, Gutiérrez,

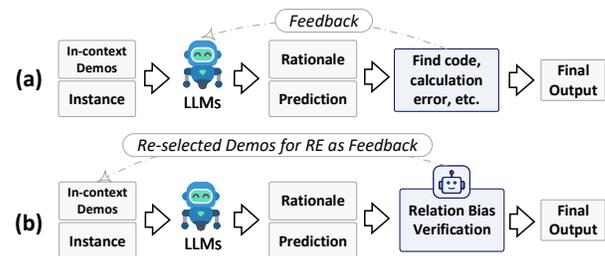


Figure 1: Comparison between current automated feedback methods (a) and ours (b). The main difference is that our rationale supervisor can verify whether the relation bias occurs and provide re-selected demonstrations as feedback.

and Su 2023) to improve the performance. The verification and feedback mechanism for correcting the biased prediction is still missing from current LLM based RE research.

To fill this gap, in this study, we focus on exploring the verification and feedback mechanism (Pan et al. 2023) of LLMs for RE. Specifically, we aim to examine whether the relation prediction of LLMs is biased by verifying the rationale (the generated explanation when LLMs perform RE) and providing feedback for correction. However, the current verification and feedback mechanism faces the following two problems when being applied to RE.

Firstly, existing methods are mainly designed for other tasks, e.g., the reasoning task. The objectives of their feedback are also tailored for those tasks, e.g., correcting code, factual, or calculation errors in initial responses (Zhang et al. 2023; Gou et al. 2023), or choosing an optimal prefix for the next step in multi-step reasoning (Khalifa et al. 2023), as shown in Fig. 1 (a). For example, for the mathematical reasoning task, Self-Refine (Madaan et al. 2023) utilizes the LLM agent to find calculation errors in the initial answer and provide error information as feedback to correct the answer. However, such feedback objectives are based on the logical properties of reasoning tasks, which are not available for RE.

Secondly, existing methods (Madaan et al. 2023; Nathani et al. 2023) do not include demonstrations in their feedback. However, the demonstrations are essential for RE even at the correction stage. This is because without demonstrations in the feedback, the RE task would degrade to zero-shot

\*Corresponding author.

RE and is harder than the initial few-shot one. Moreover, the demonstrations in initial few-shot RE cannot be directly used in feedback since they will mislead the model back to the initial one, and thus the impact of feedback is discarded.

To address the above problems, we propose a novel automated feedback framework for RE, which *trains a rationale supervisor* based on a BERT-like small model and utilizes it to not only *verify the prediction* but also *provide new demonstration improved feedback for correction* during the inference. As shown in Fig. 1 (b), our rationale supervisor provides re-selected demonstrations as feedback for correcting the initial prediction of LLMs.

In order to train a rationale supervisor, we need to collect both unbiased and biased rationales, i.e., positive and negative samples. Though several verification methods have been proposed to collect positive and negative rationales in other tasks, both their purpose and the collection method are not suitable for our RE task. (1) Firstly, their collected positive and negative rationales are used for training the verifier, which only needs to discriminate the positive predictions from negative ones. In contrast, the rationale supervisor in our framework is designed to correct biased predictions, thus needing to further discriminate different negative rationales. (2) Secondly, the way of collecting rationales in current verification methods relies on the manually annotated golden reasoning steps as positive samples and perform rule-based perturbation (Paul et al. 2023; Golovneva et al. 2023) or error step alignment (Khalifa et al. 2023; Li et al. 2023b) to obtain negative samples. Unfortunately, such annotated samples and rules for perturbation are not available in RE.

In view of this, we propose a causal intervention and observation method to address the lack of annotated rationales and collect biased rationales for training the supervisor. Specifically, we first present *a label-guided intervention strategy to collect unbiased rationales*, and we also present *a diversified intervention strategy to collect biased rationales*. In addition, during the inference, we utilize the rationale supervisor to retrieve new demonstrations from the labeled samples and include them in the feedback, which are then used by the LLM for re-generating predictions. Since the supervisor has learned the difference among various biased rationales, the LLM gets the signal to adjust its direction for correction. This verification-feedback-correction procedure iterates until the output rationale is verified as unbiased.

Overall, we make three major contributions. 1) We extend the LLM based RE research to the automated feedback paradigm, which equips LLM with the ability of correcting the biased prediction. 2) We propose a novel supervised rationale verification and feedback framework, which first collects rationales with a causal intervention and observation method for training the supervisor, and then employs the supervisor to retrieve sample-related demonstrations as feedback for guiding the LLM in correction. 3) Extensive experiments prove that our proposed method can improve the performance of LLM based RE methods and is superior to existing automated feedback methods.

## Related Work

**LLMs for Relation Extraction** Recently, many studies (Xu et al. 2023; Li et al. 2023a; Wei et al. 2023; Wadhwa, Amir, and Wallace 2023; Li, Wang, and Ke 2023) have explored how to unlock the potential of LLMs for the RE task, including designing the in-context demonstration selection strategy (Wan et al. 2023; Ma, Li, and Zhang 2023; Pang et al. 2023) and optimizing instruction patterns (Zhang, Gutiérrez, and Su 2023; Wang et al. 2023a; Ma et al. 2023). Despite great success, these methods rely solely on optimizing the initial prompt to improve performance. However, we find that due to the relation bias, LLMs may still confuse certain relations with similar entities and thus make biased predictions. To alleviate this issue, we introduce the idea of automated feedback to RE for the first time, expecting to correct biased predictions via the provided feedback.

**LLMs with Automated Feedback** Some researchers have exploited the automated feedback for correcting the undesirable output of LLMs (Pan et al. 2023; Kamoi et al. 2024). However, the feedbacks in existing methods are designed for correcting various reasoning mistakes, e.g., code errors (Zhang et al. 2023), factual errors (Gou et al. 2023), calculation errors (Nathani et al. 2023; Madaan et al. 2023; Paul et al. 2023), or as an optimal prefix for the next step in multi-step reasoning (Khalifa et al. 2023; Li et al. 2023b). These feedbacks are dependent on the reasoning task and unavailable for RE. Moreover, they do not include the demonstrations which are essential for RE. To address this issue, we propose a novel automated feedback framework which provides re-selected demonstrations as feedbacks to help LLMs correct the biased prediction.

## Method

This section presents our proposed supervised rationale verification and feedback (SRVF) framework for the RE task.

**Task Formulation** Given a set of pre-defined relation types  $Y_D$ , the relation extraction (RE) task aims to predict the relation type  $y \in Y_D$  between the head entity  $e^h$  and the tail entity  $e^t$  of each test example  $x = \{s, e^h, e^t\}$ , where  $s$  denotes the sentence. In this study, we adopt in-context learning (ICL) with the rationale to prompt LLMs for the RE task. Specifically, for each test example  $x$ , we need to randomly select or retrieve  $m$  initial in-context demonstrations  $D_{icl} = \{\{x_1, r_1^u, y_1\}, \dots, \{x_m, r_m^u, y_m\}\}$  related to  $x$  from the labeled dataset  $D_l$ <sup>1</sup>. Then, the LLM  $f_\theta$  with parameters  $\theta$  is expected to output the relation type  $y \in Y_D$  between  $e^h$  and  $e^t$ , along with the rationale  $r$ , denoted as  $\{r, y\} = f_\theta(D_{icl}, x)$ .

**Overview** In this paper, we propose a rationale verification and feedback framework to guide LLMs towards better predictions for RE iteratively. Generally, this framework

<sup>1</sup>Since there is no annotated golden rationale in the original dataset, we add the induced unbiased rationale in the following section to  $D_l$  to enable it for the setup of ICL with the rationale, i.e.,  $D_l = \{\{x_1, r_1^u, y_1\}, \dots, \{x_n, r_n^u, y_n\}\}$ .

consists of three phases: 1) causal intervention and observation for rationale collection, 2) contrastive training rationale supervisor, and 3) rationale verification and feedback.

Specifically, we first adopt the causal intervention and observation method to collect unbiased and biased rationales, i.e.,  $R_u$  and  $R_b$ . Then, we use  $R_u$  and  $R_b$  to train the rationale supervisor  $\mathcal{R}_\gamma$  with parameters  $\gamma$ . Finally, as shown in Fig. 3, in the inference time, once the output rationale  $r$  is verified as a biased one by  $\mathcal{R}_\gamma$ , we use  $\mathcal{R}_\gamma$  to retrieve feedback demonstrations  $D_{fb}$  based on  $r$ , where  $D_{fb} \subset D_l$ . The feedback demonstrations are used for re-generating  $r$  and  $y$  using ICL, i.e.,  $\{r, y\} = f_\theta(D_{fb}, x)$ . The procedure iterates until the rationale  $r$  is verified as unbiased, and the corresponding relation prediction  $y$  will finally be output.

### Causal Intervention and Observation for Rationale Collection

Generally, during this phase, for each labeled sample  $\{x_i, y_i\}$ , we aim to collect the unbiased rationale corresponding with the golden label  $\{r_i^u, y_i^u\}$ , as well as the biased rationale with corresponding biased relation prediction  $\{r_i^b, y_i^b\}$ . This process consists of two steps: 1) induce unbiased rationale, and 2) observe biased rationale. As shown in Fig. 2, we use the structural causal model (SCM) in causal inference (Pearl et al. 2000) to illustrate the strategy.

**Preliminary of SCM** As shown in Fig. 2, the SCMs show the relationships among the input ( $X$ ), the relation prediction ( $Y$ ), the rationale for prediction ( $R$ ), the certain bias of LLMs ( $B$ ) and in-context demonstration  $I$ . The arrows between nodes indicate causal directions. For example, “ $X \rightarrow R$ ” means that the LLM generates the rationale  $R$  related to the prediction for the sample  $X$ . “ $X \rightarrow B \rightarrow R$ ” indicates that the LLM activates some biased knowledge  $B$  related to the sample  $X$  and generates a rationale  $R$  influenced by the biased knowledge  $B$ . Besides, in Fig. 2 (b), the “ $do(Y)$ ” indicates that cutting off all factors that could influence the value of  $Y$  and assigning  $Y$  a certain value as needed.

**Induce Unbiased Rationale** Previous methods rely on the human-annotated rationales, e.g., golden reasoning steps in mathematical tasks (Khalifa et al. 2023), which are not available in the RE dataset. To address this issue, we propose a *label-guided intervention strategy to obtain the unbiased rationale for each labeled sample*, which explains why the sample  $x_i$  should be predicted as the golden label  $y_i$ .

As shown in Fig. 2 (b), this strategy consists of two steps: 1) cut causal directions that could make bias ( $B$ ) influence the prediction ( $Y$ ), and let the golden label guide the rationale ( $R$ ) generation, formally denoted as  $do(Y = y_i)$  and  $do(Y) \rightarrow R$ . The observed generated rationale is  $R = r_i^u$ ; 2) conduct similar do-operation to the rationale  $R$  and let  $do(R)$  point to  $Y$ , i.e.,  $do(R = r_i^u)$ ,  $do(R) \rightarrow Y$ . If the observed value of  $Y$  is equal to the golden label  $y_i$ , we treat  $\{r_i^u, y_i\}$  as the unbiased one and add it to  $R_u$ .

**Observe Biased Rationale** In previous methods, incorrect rationales are synthesized from golden ones using perturbation or error step alignment based on certain rules (Golovneva et al. 2023; Khalifa et al. 2023). However,

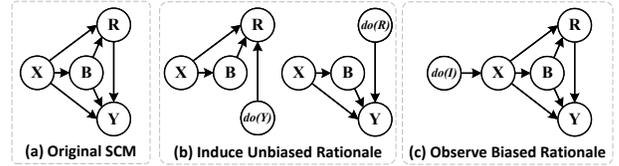


Figure 2: The structure causal model for illustrating the proposed causal intervention and observation strategy.

these rules are designed based on the logical properties of reasoning tasks, which are not available in RE. To tackle this problem, we propose a *diversified intervention strategy for collecting the biased rationales*.

Specifically, for the labeled sample  $\{x_i, y_i\}$ , we first randomly select a demonstration set  $D_{di}$  with diverse labels, where  $D_{di} \subset D_l$  and the label of each demonstration in  $D_{di}$  is not equal to  $y_i$ . The diversity of labels in  $D_{di}$  is designed to induce LLMs to make diverse errors on the same sample, to increase the diversity of collected biased rationales. Then, as shown in Fig. 2 (c), we set the in-context demonstration  $I$  as  $\{x_j, r_j^u, y_j\}$  from  $D_{di}$ , i.e.,  $do(I = \{x_j, r_j^u, y_j\})$ . Finally, the observed value of rationale  $R$  is  $r_{obs}$  while the observed value of rationale  $Y$  is  $y_{obs}$ . If  $y_{obs} \neq y_i$ , we treat the observed  $r_{obs}$  with its corresponding relation prediction  $y_{obs}$  as a potentially biased one, i.e.,  $\{r_i^b, y_i^b\}$ , and add it to  $R_b$ .

### Contrastive Training Rationale Supervisor

We expect the rationale supervisor to 1) verify whether the output rationale is biased, and 2) provide different feedbacks for different bias situations to correct the initial prediction. To reach this, we adopt contrastive learning to train the rationale supervisor to acquire two abilities: 1) discriminating biased and unbiased rationales, and 2) learning the difference of various biased rationales.

We design two kinds of positive and negative pairs for contrastive training.

For positive pairs, we treat “unbiased rationales with the same golden label”, and “biased rationales under the same bias situation” as the two kinds of positive pairs. For example, if samples  $s_1$  and  $s_2$ , which have the same label, are also predicted as the same wrong relation, we call “samples  $s_1$  and  $s_2$  are in the same bias situation”. Thus, the biased rationales ( $r_1^b$  and  $r_2^b$ ) of  $s_1$  and  $s_2$ , are treated as a positive pair and should be pulled together in the rationale representation space, i.e.,  $r_1^b \rightarrow \leftarrow r_2^b$ .

For negative pairs, we first consider the “biased and unbiased rationales from the same sample” as a negative pair. This is designed to train the rationale supervisor to distinguish between biased and unbiased rationales. For example, a sample  $s_1 = \{r_1^u, y_1\}$  where  $y_1$  is the golden label and  $r_1^u$  is the corresponding unbiased rationale, is wrongly predicted as relation  $y_2$  and corresponding biased rationale is  $r_1^b$ . Thus,  $r_1^b$  and  $r_1^u$  are treated as a negative pair and should be pushed away in the rationale representation space, i.e.,  $r_1^b \leftrightarrow r_1^u$ . Second, we also treat “biased rationales under different bias situations” as a negative pair to train the rationale supervisor, which can distinguish different bias situa-

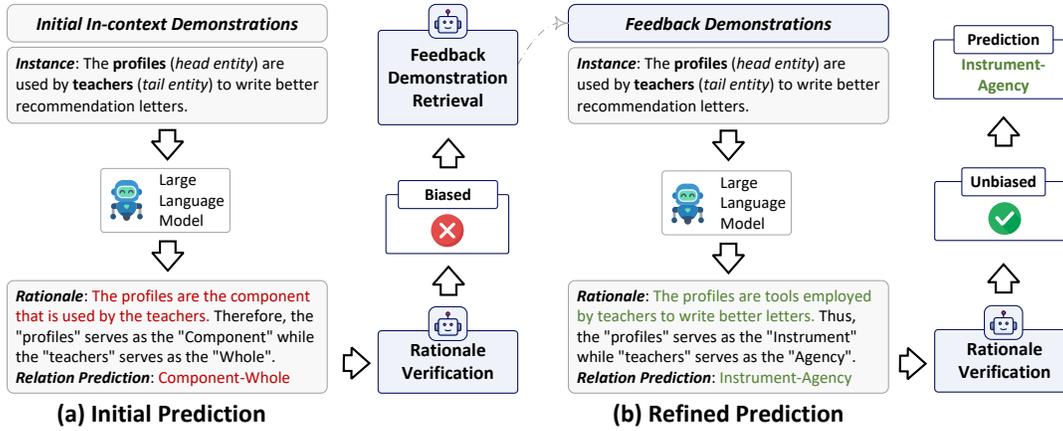


Figure 3: An example of correcting the initial biased prediction of LLMs via the proposed SRVF framework in the inference time. The rationale supervisor first verifies the initial prediction in (a) as biased. Then, with the feedback demonstrations retrieved by the rationale supervisor, the LLM makes a correct relation prediction in (b). Note: The rationale supervisor here is obtained by contrastive training using collected biased and unbiased rationales as described before.

tions and provide feedback based on the biased rationale in the inference time.

In general, the contrastive loss is calculated as:

$$\mathcal{L}_{cl} = -\log \frac{\frac{1}{\|S^{pos}\|} \sum_{\{r_1, r_2\} \in S^{pos}} \exp(\text{sim}(r_1, r_2)/\tau)}{\sum_{\{r_1, r_2\} \in (S^{pos} \cup S^{neg})} \exp(\text{sim}(r_1, r_2)/\tau)}, \quad (1)$$

$$\text{sim}(r_1, r_2) = \mathcal{R}_\gamma(r_1) \cdot \mathcal{R}_\gamma(r_2)^\top, \quad (2)$$

where  $S^{pos} = S_1^{pos} \cup S_2^{pos}$ ,  $S^{neg} = S_1^{neg} \cup S_2^{neg}$ .  $S_1^{pos}$  and  $S_2^{pos}$  denote the two kinds of positive pair set, and  $S_1^{neg}$  and  $S_2^{neg}$  denote two kinds of negative pair set. Here we adopt the dot product as the similarity function  $\text{sim}()$  and add a temperature hyper-parameter  $\tau$  to focus more on difficult pairs (Chen et al. 2020). During the procedure of rationale contrastive training, the parameters  $\gamma$  of  $\mathcal{R}_\gamma$  are updated to minimize  $\mathcal{L}_{cl}$ .

### Rationale Verification and Feedback

As shown in Fig. 3, in the inference time, the trained rationale supervisor  $\mathcal{R}_\gamma$  first verifies whether the prediction is biased. If the prediction is biased, the rationale supervisor will retrieve a feedback demonstration set, which then guides LLMs toward refined predictions. In this subsection, we will elaborate on the ‘‘Rationale Verification’’ and ‘‘Feedback Demonstration Retrieval’’ in Fig. 3 in detail. Here we denote the test example, output rationale, and relation prediction of LLMs as  $x$ ,  $r$ , and  $y$ , respectively.

**Rationale Verification** For verification, we need to select the subsets  $S_b$  and  $S_u$  related to the prediction  $y$  from  $R_b$  and  $R_u$ , respectively, which are then used as anchors to determine whether the current output rationale is close to the biased or unbiased groups.  $S_b$  and  $S_u$  are defined as follows:

$$S_b = \{\{r^b, y^b\} \mid \{r^b, y^b\} \in R_b, y^b = y\}, \quad (3)$$

$$S_u = \{\{r^u, y^u\} \mid \{r^u, y^u\} \in R_u, y^u = y\}, \quad (4)$$

Then, the indicator score to judge whether  $r$  is a biased rationale is calculated as follows:

$$p_b = \max_{\{r^b, y^b\} \in S_b} \text{sim}(r, r^b) - \max_{\{r^u, y^u\} \in S_u} \text{sim}(r, r^u), \quad (5)$$

where the similarity function  $\text{sim}()$  is defined in Eq. 2. When  $p_b$  is greater than 0, it implies that the feature of  $r$  is closer to the feature field of  $S_b$  than that of  $S_u$ , which means  $r$  and corresponding relation prediction  $y$  should be regarded as biased, and feedback is needed to correct them.

**Feedback Demonstration Retrieval** Once the output rationale  $r$  is verified as biased, we need to retrieve a new set of in-context demonstrations based on the feature of  $r$  for guiding LLMs toward correct predictions. Specifically, we first select the  $k$  most similar biased rationales to  $r$  in  $S_b$ , denoted as  $S_b^{topk}$ , which is defined as:

$$S_b^{topk} = \{\{r^b, y^b\} \mid \text{rank}_{\{r^b, y^b\} \in S_b}(\text{sim}(r, r^b)) \leq k\}, \quad (6)$$

Then, we select the labeled samples corresponding to the biased rationales in  $S_b^{topk}$  from  $D_l$  as the feedback demonstrations  $D_{fb}$ , which is defined as:

$$D_{fb} = \{\{x_i, r_i^u, y_i\} \mid \{x_i, r_i^u, y_i\} \in D_l, \{r_i^b, y_i^b\} \in S_b^{topk}\}, \quad (7)$$

where the biased  $\{r_i^b, y_i^b\}$  and unbiased  $\{r_i^u, y_i^u\}$  correspond to the same labeled sample  $\{x_i, y_i\}$ .

**Correction via In-context Learning** After the feedback demonstrations  $D_{fb}$  are selected, we re-generate  $r$  and  $y$  using the LLM  $f_\theta$ , i.e.,  $\{r, y\} = f_\theta(D_{fb}, x)$ . This process will be iteratively performed until  $r$  is verified as unbiased, and the corresponding prediction  $y$  will be finally output.

Method	SemEval				TACRED				Re-TACRED				Avg.	
	5-shot	10-shot	20-shot	50-shot	5-shot	10-shot	20-shot	50-shot	5-shot	10-shot	20-shot	50-shot		
Random	In-context Learning	48.40	49.11	49.65	49.31	24.17	23.69	24.66	24.21	21.37	21.99	21.52	21.10	31.60
	w/ Self-Refine	47.93	48.50	49.19	48.88	23.15	23.05	24.18	23.12	21.04	21.51	20.71	21.32	31.05
	w/ Self-Consistency	49.30	49.09	50.11	50.35	25.69	24.76	25.64	25.42	22.08	22.56	21.84	22.01	32.40
	w/ GRACE	50.80	49.22	54.28	54.83	25.89	25.78	26.49	26.46	22.50	22.67	23.65	24.34	33.91
	<b>w/ our SRVF</b>	<b>54.89</b>	<b>59.67</b>	<b>62.98</b>	<b>71.27</b>	<b>30.07</b>	<b>31.42</b>	<b>32.84</b>	<b>34.58</b>	<b>28.36</b>	<b>31.49</b>	<b>32.87</b>	<b>36.52</b>	<b>42.25</b>
SimCSE	In-context Learning	57.33	59.13	62.49	64.26	27.48	28.64	30.08	27.81	34.78	41.85	42.82	43.69	43.36
	w/ Self-Refine	57.01	58.91	62.27	63.89	27.13	26.89	29.11	27.30	34.33	41.87	42.30	43.16	42.85
	w/ Self-Consistency	57.54	58.81	62.98	65.00	28.82	29.85	30.98	25.42	35.83	42.84	43.71	44.59	43.86
	w/ GRACE	57.93	58.48	66.32	67.48	28.76	28.60	30.03	26.46	33.95	41.53	42.35	44.37	43.86
	<b>w/ our SRVF</b>	<b>60.76</b>	<b>64.12</b>	<b>69.54</b>	<b>76.32</b>	<b>32.99</b>	<b>33.50</b>	<b>34.81</b>	<b>36.13</b>	<b>39.48</b>	<b>46.54</b>	<b>49.73</b>	<b>54.31</b>	<b>49.85</b>
Task-specific	In-context Learning	58.68	64.90	65.67	77.32	26.11	26.35	31.15	33.35	42.75	45.53	52.89	56.22	48.41
	w/ Self-Refine	58.38	64.96	65.68	77.35	25.01	25.48	30.62	32.67	42.10	44.98	52.11	55.62	47.91
	w/ Self-Consistency	59.62	65.45	65.74	77.48	26.93	26.83	31.67	33.61	43.54	46.04	53.34	56.69	48.91
	w/ GRACE	60.83	65.14	66.21	76.98	27.12	26.34	30.95	33.40	43.12	45.23	52.61	55.83	48.65
	<b>w/ our SRVF</b>	<b>62.12</b>	<b>67.03</b>	<b>68.94</b>	<b>80.08</b>	<b>30.50</b>	<b>30.92</b>	<b>34.83</b>	<b>36.32</b>	<b>46.13</b>	<b>48.09</b>	<b>55.07</b>	<b>59.82</b>	<b>51.65</b>

Table 1: Results (micro-F1 scores) on the SemEval, TACRED, and Re-TACRED datasets under various few-shot settings. Here we adopt the Llama-2-7b-chat as the LLM. The best results are in **bold**.

## Experiments

### Evaluation Protocol

**Datasets and Metric** We adopt three commonly used datasets for RE, including SemEval (Hendrickx et al. 2010), TACRED (Zhang et al. 2017), and Re-TACRED (Stoica, Platanios, and Póczos 2021). Besides, compared to the scenario with full data, the potential of LLMs under few-shot settings is of more concern (Ma et al. 2023; Xu et al. 2023). Hence we adopt the  $k$ -shot ( $k \in \{5, 10, 20, 50\}$ ) settings to validate the effectiveness of the proposed method. For all experiments, we report micro-F1 scores where `Other` and `no_relation` are considered negative labels.

**Backbones** We experiment with three different methods as backbones for selecting initial in-context demonstrations for LLM based RE, including: 1) *Random*, which randomly selects initial demonstrations without any retriever. 2) *SimCSE*, which uses SimCSE (Gao, Yao, and Chen 2021) to retrieve samples that have similar sentence semantics with the test example as initial in-context demonstrations. 3) *Task-specific*, which uses a task-specific retriever that has been trained on the labeled samples (Wan et al. 2023).

**Baselines** To the best of our knowledge, we are the first to explore the verification and feedback mechanism for LLM based RE. Thus, we can only make modifications on current feedback methods in other tasks to adapt them for RE. Specifically, we choose the following baselines:

- *Self-Refine* (Madaan et al. 2023) consists of three LLM based agents, i.e., RE agent, verifier agent, and refiner agent, for iterative feedback and refinement.
- *Self-Consistency* (Wang et al. 2023b) is proposed to conduct verification for the multiple candidate responses and choose the best response by majority voting.

- *GRACE* (Khalifa et al. 2023) trains a verifier to select the best intermediate reasoning step, which is then used as feedback for generating the next step.

For Self-Consistency, GRACE, and ours, the number of iterations or candidate responses is set to 5 for fairness. For Self-Refine, the iteration number is set to 1 since we find that more iteration rounds result in performance degradation<sup>2</sup>.

### Main Results

Table 1 reports the experimental results with various initial demonstration selection strategies on *Llama-2-7b-chat* on the SemEval, TACRED, and Re-TACRED datasets. From Table 1, we can draw the following conclusions: 1) Our proposed SRVF framework yields significant enhancements upon various backbones with different demonstration selection strategies. Specifically, the improvement is most significant when randomly selecting the initial demonstrations, getting a 10.65% absolute micro-F1 score increase on average. Besides, when using SimCSE and task-specific retriever as backbones to carefully select initial in-context demonstrations, there are also 6.49% and 3.24% absolute micro-F1 score boosts on average, respectively. 2) Our proposed method exhibits significant superiority over existing verification and feedback methods under all settings. The multi-agent based Self-Refine method is the worst, which is mainly due to its unsuitable feedback objectives and correction manner. Existing methods for verifying the output of LLMs, i.e., Self-Consistency and GRACE, can enhance the performance of in-context learning to some extent. However, since they do not provide explicit feedback signals for LLMs to correct the prediction, their improvements are limited.

<sup>2</sup>Please refer to Appendix for detailed implementation details of baselines and ours.

Method	SemEval		TACRED		Re-TACRED		Avg.
	5-shot	10-shot	5-shot	10-shot	5-shot	10-shot	
<b>Our SRVF</b>	<b>59.26</b>	<b>63.61</b>	<b>31.19</b>	<b>31.95</b>	<b>37.99</b>	42.04	<b>44.34</b>
<i>w/o LGI</i>	55.31	55.46	25.13	29.83	24.44	30.46	36.77
<i>w/o DI</i>	57.90	62.50	28.52	29.78	35.64	39.99	42.39
<i>w/o RCT</i>	58.37	62.78	30.31	30.87	37.23	<b>43.73</b>	43.88
<i>w/o FDR</i>	57.03	60.23	29.27	29.85	35.59	39.39	41.89
<i>w/o RG</i>	52.27	62.09	27.52	29.33	35.14	38.38	40.79

Table 2: The ablation results (micro-F1) averaged over three backbones. The best results are in **bold**.

## Ablation Study

To validate the effectiveness of components in our method, we introduce the following variants for ablation studies:

- *w/o label-guided intervention (LGI)*, where the labels do not guide the collecting of unbiased rationales.
- *w/o diversified intervention (DI)*, which replaces the DI with random sampling for collecting biased rationales.
- *w/o rational contrastive training (RCT)*, which trains the rationale supervisor with cross-entropy loss.
- *w/o feedback demonstration retrieval (FDR)*, which removes the FDR strategy and uses the initially selected demonstrations as the feedback.
- *w/o RG*, which skips the re-generation process and directly adopts the label of the top-1 retrieved demonstration as the final prediction.

The results of the ablation study are shown in Table 2. From the table, we make the following observations. 1) Removing LGI and DI strategies significantly degrades performance, indicating that LLMs struggle to collect unbiased rationales based solely on generation without causal intervention. 2) Eliminating RCT also reduces performance, demonstrating its effectiveness in helping the rationale supervisor distinguish between unbiased and various biased situations. 3) Omitting FDR significantly decreases performance, highlighting its crucial role in guiding LLMs toward corrected predictions despite iterative verification. 4) Removing the re-generation process results in a substantial performance drop, showcasing that simple assignment of retrieved top-1 demonstrations isn’t sufficient and that in-context feedback for re-generation adds robustness to the correction process.

## Analysis

**Effectiveness on Various-scale LLMs** To examine whether the proposed method remains effective for various-scale LLMs, we conduct experiments on various sizes of LLMs from the *Llama-2-chat* (Touvron et al. 2023), *Meta-Llama-3-Instruct* (AI@Meta 2024), and *GPT-3.5* (Ouyang et al. 2022), and present their results in Table 3.

From Table 3, it can be seen that our rationale supervisor can boost the performance of LLMs with various sizes. Specifically, even with the most powerful *Meta-Llama-3-70B-Instruct*, there is still a 2.47% micro-F1 score improvement over the original in-context learning. The

Method	SemEval		TACRED		Re-TACRED		Avg.
	5-shot	10-shot	5-shot	10-shot	5-shot	10-shot	
R-BERT	42.75	57.25	9.87	16.24	26.64	35.01	31.29
KnowPrompt	53.92	56.42	27.86	30.34	50.08	55.41	45.67
<i>Llama-2-7b-chat</i>							
ICL	58.68	64.90	26.11	26.35	42.75	45.53	44.05
w/ SRVF	62.12	67.03	30.50	30.92	46.13	48.09	47.47
<i>Llama-2-70b-chat</i>							
ICL	68.92	69.86	27.32	27.12	43.63	44.94	46.97
w/ SRVF	69.97	70.00	27.69	29.47	45.13	46.93	48.20
<i>Meta-Llama-3-8B-Instruct</i>							
ICL	69.90	69.79	32.63	32.26	48.23	50.69	50.58
w/ SRVF	71.14	71.41	35.26	34.29	52.23	55.25	53.26
<i>Meta-Llama-3-70B-Instruct</i>							
ICL	71.21	72.40	34.71	34.97	56.10	57.41	54.47
w/ SRVF	74.68	74.33	37.05	36.35	59.27	59.96	56.94
<i>GPT-3.5-turbo</i>							
ICL	67.26	70.58	32.46	31.38	43.56	46.88	48.69
w/ SRVF	69.62	71.67	37.78	34.63	46.22	49.66	51.60

Table 3: Results (micro-F1 scores) using various LLMs with the task-specific retriever.

experimental results indicate that the “relation bias” issue exists in LLMs of various scales, and our proposed method can function as a plug-in module for various LLMs to effectively mitigate this problem.

## Comparison with Well-designed Few-shot Methods for RE

As shown in Table 3, we include two established supervised fine-tuning methods for RE as baselines: 1) R-BERT (Wu and He 2019), which fine-tunes a BERT for the RE task, and 2) KnowPrompt (Chen et al. 2022), which is tailored for few-shot scenarios and has shown good few-shot performance. As we can see from the results, with the help of our proposed SRVF, even the relatively weak *Llama-2-7b-chat* can outperform KnowPrompt by 1.80% averagely. Moreover, when deploying our SRVF on the most powerful *Meta-Llama-3-70B-Instruct*, there is an average performance improvement of 11.27% compared to KnowPrompt.

**Analysis on Successfully Corrected Samples** To visualize which samples are successfully corrected by the proposed method, we compare the error matrix on the SemEval dataset before and after correction. The results are obtained by summing the number of error predictions of all settings in Table 1. The results are shown in Fig. 4.

From Fig. 4 (a), we observe that LLMs struggle to distinguish between relations that share similar entities, e.g., 687 samples labeled as “Entity-Destination” are incorrectly predicted as “Content-Container”. Such error can arise when, for example, given sentences “please move the eggs into the box” and “there are 5 eggs in the box”, where the same entity pair “eggs” and “box” form “Entity-Destination” and “Content-Container” relations, respectively. Such ambigu-

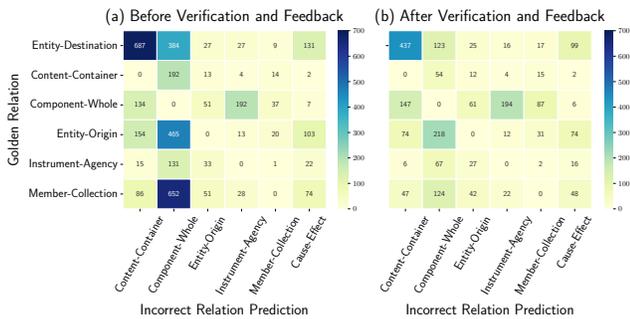


Figure 4: Error matrix before and after the verification-feedback-correction procedure. The numbers show how many samples labeled  $y$  (on the vertical axis) are incorrectly predicted as  $x$  (on the horizontal axis).

ity often leads LLMs to misclassify relations when they fail to focus on context, resulting in numerous errors. However, as shown in Fig. 4 (b), the number of samples labeled as “Entity-Destination” but incorrectly predicted as “Content-Container” is reduced by 250. This indicates that our method effectively alleviates the above issue.

**Analysis on Method Efficiency** Considering possible concerns on the inference efficiency due to the iterative feedbacks, we compare the inference time on the SemEval dataset of different methods. Besides, we also evaluate the pre-inference time of each method, e.g., the time to obtain biased/unbiased data and train the rationale supervisor in our SRVF. The comparison results are shown in Fig. 5.

From Fig. 5, we can observe that:

- 1) Basic in-context learning (ICL) is the most efficient.
- 2) Self-Refine does not require pre-inference time, but its inference time is more than the sum of our pre-inference time and inference time. Moreover, Self-Refine has the worst performance among all methods (Table 1).
- 3) Self-Consistency and GRACE have much higher computational costs than our SRVF, especially in terms of inference time. This is mainly because the proposed rationale supervisor can verify whether the LLM prediction is biased. Only the test samples verified as biased by the rationale supervisor will proceed to the correction round for regeneration. This greatly reduces the time cost of our method in inference time after correction.

Overall, our SRVF is the second-best in computational efficiency while achieving the best performance (Table 1).

**Experiments on Document-level RE** To explore the effectiveness of our method for document-level RE, we apply SRVF on three backbones and conduct experiments on two commonly used document-level RE datasets, DocRED (Yao et al. 2019) and Re-DocRED (Tan et al. 2022). The random and SimCSE backbones are kept the same as before. For the task-specific backbone, we borrow the idea from RE-PLM (Ozyurt, Feuerriegel, and Zhang 2024), which obtains the final prediction by aggregating the predictions based on multiple retrieved demonstrations. The experimental results are reported in Table 4.

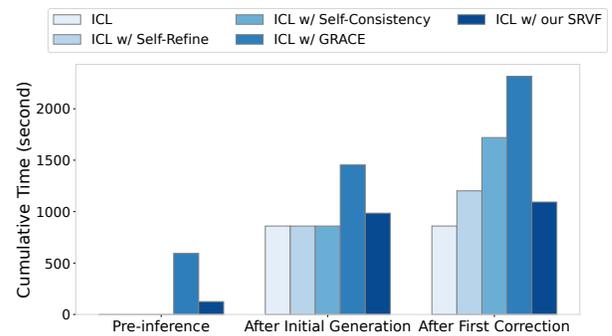


Figure 5: Efficiency comparison of different methods on the 5-shot SemEval setting. The results are accumulated along the X axis. For example, “After Initial Generation” refers to the sum time of “pre-inference” and “initial generation”.

Method	DocRED		Re-DocRED		Avg.
	5-shot	10-shot	5-shot	10-shot	
ICL (Random)	7.76	7.82	8.27	7.73	7.90
w/ our SRVF	<b>15.40</b>	<b>18.00</b>	<b>15.29</b>	<b>15.65</b>	<b>16.09</b>
ICL (SimCSE)	15.67	16.40	11.97	12.53	14.14
w/ our SRVF	<b>18.39</b>	<b>21.55</b>	<b>17.38</b>	<b>17.87</b>	<b>18.80</b>
ICL (Task-specific)	18.29	18.40	17.44	18.67	18.20
w/ our SRVF	<b>20.04</b>	<b>21.55</b>	<b>19.98</b>	<b>21.69</b>	<b>20.82</b>

Table 4: Results (micro-F1) on the DocRED (document-level RE task). The best results are in **bold**.

From Table 4, we can observe that: 1) LLM performs poorly on document-level RE, which is consistent with empirical observations in Li, Jia, and Zheng (2023); Sun et al. (2024). This is due to the difficulty LLMs face in selecting entity pairs that have certain relations from a vast space of candidate entity pairs. Besides, the large number of candidate relation labels (96 in DocRED and Re-DocRED) further increases the difficulty in assigning each entity pair a relation. 2) Our proposed SRVF effectively enhances the performance of LLM under various settings on DocRED and Re-DocRED, indicating that our method remains to be effective in such challenging scenarios.

## Conclusion

In this paper, we propose a novel automated feedback framework for LLM based relation extraction (RE), which includes a rationale supervisor to iteratively correct the biased relation prediction of LLMs. Specifically, we first present a causal intervention and observation method to collect unbiased and biased rationales, which are then used to train the rationale supervisor. Then, we develop a verification-feedback-correction procedure to iteratively enhance LLMs’ ability to correct the biased prediction. Extensive experiments demonstrate the superiority of our framework over existing methods. In the future, we will try to extend the proposed framework to other NLP tasks.

## Acknowledgments

This work was supported by the grant from the National Natural Science Foundation of China (NSFC) project (No. 62276193).

## References

- AI@Meta. 2024. Llama 3 Model Card. [https://github.com/meta-llama/llama3/blob/main/MODEL\\_CARD.md](https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md).
- Bai, Y.; Jones, A.; Ndousse, K.; Askell, A.; Chen, A.; Das-Sarma, N.; Drain, D.; Fort, S.; Ganguli, D.; Henighan, T.; Joseph, N.; Kadavath, S.; Kernion, J.; Conerly, T.; El-Showk, S.; Elhage, N.; Hatfield-Dodds, Z.; Hernandez, D.; Hume, T.; Johnston, S.; Kravec, S.; Lovitt, L.; Nanda, N.; Olsson, C.; Amodei, D.; Brown, T.; Clark, J.; McCandlish, S.; Olah, C.; Mann, B.; and Kaplan, J. 2022. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *arXiv:2204.05862*.
- Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020. A Simple Framework for Contrastive Learning of Visual Representations. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 1597–1607. PMLR.
- Chen, X.; Zhang, N.; Xie, X.; Deng, S.; Yao, Y.; Tan, C.; Huang, F.; Si, L.; and Chen, H. 2022. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web conference 2022*, 2778–2788.
- Gao, T.; Yao, X.; and Chen, D. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6894–6910. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.
- Golovneva, O.; Chen, M.; Poff, S.; Corredor, M.; Zettlemoyer, L.; Fazel-Zarandi, M.; and Celikyilmaz, A. 2023. ROSCOE: A Suite of Metrics for Scoring Step-by-Step Reasoning. *arXiv:2212.07919*.
- Gou, Z.; Shao, Z.; Gong, Y.; Shen, Y.; Yang, Y.; Duan, N.; and Chen, W. 2023. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. *arXiv:2305.11738*.
- Hendrickx, I.; Kim, S. N.; Kozareva, Z.; Nakov, P.; Ó Séaghdha, D.; Padó, S.; Pennacchiotti, M.; Romano, L.; and Szpakowicz, S. 2010. SemEval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, 33–38. Uppsala, Sweden: Association for Computational Linguistics.
- Kamoi, R.; Zhang, Y.; Zhang, N.; Han, J.; and Zhang, R. 2024. When Can LLMs Actually Correct Their Own Mistakes? A Critical Survey of Self-Correction of LLMs. *arXiv preprint arXiv:2406.01297*.
- Khalifa, M.; Logeswaran, L.; Lee, M.; Lee, H.; and Wang, L. 2023. GRACE: Discriminator-Guided Chain-of-Thought Reasoning. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Findings of the Association for Computational Linguistics: EMNLP 2023*, 15299–15328. Singapore: Association for Computational Linguistics.
- Li, B.; Fang, G.; Yang, Y.; Wang, Q.; Ye, W.; Zhao, W.; and Zhang, S. 2023a. Evaluating ChatGPT’s Information Extraction Capabilities: An Assessment of Performance, Explainability, Calibration, and Faithfulness. *arXiv preprint arXiv:2304.11633*.
- Li, G.; Wang, P.; and Ke, W. 2023. Revisiting Large Language Models as Zero-shot Relation Extractors. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Findings of the Association for Computational Linguistics: EMNLP 2023*, 6877–6892. Singapore: Association for Computational Linguistics.
- Li, J.; Jia, Z.; and Zheng, Z. 2023. Semi-automatic Data Enhancement for Document-Level Relation Extraction with Distant Supervision from Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 5495–5505.
- Li, Y.; Lin, Z.; Zhang, S.; Fu, Q.; Chen, B.; Lou, J.-G.; and Chen, W. 2023b. Making Language Models Better Reasoners with Step-Aware Verifier. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5315–5333. Toronto, Canada: Association for Computational Linguistics.
- Ma, X.; Li, J.; and Zhang, M. 2023. Chain of Thought with Explicit Evidence Reasoning for Few-shot Relation Extraction. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Ma, Y.; Cao, Y.; Hong, Y. C.; and Sun, A. 2023. Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard Samples! In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegrefe, S.; Alon, U.; Dziri, N.; Prabhunoye, S.; Yang, Y.; et al. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.
- Nathani, D.; Wang, D.; Pan, L.; and Wang, W. 2023. MAF: Multi-Aspect Feedback for Improving Reasoning in Large Language Models. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6591–6616. Singapore: Association for Computational Linguistics.
- Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback. *arXiv:2203.02155*.
- Ozyurt, Y.; Feuerriegel, S.; and Zhang, C. 2024. Document-Level In-Context Few-Shot Relation Extraction via Pre-Trained Language Models.

- Pan, L.; Saxon, M.; Xu, W.; Nathani, D.; Wang, X.; and Wang, W. Y. 2023. Automatically Correcting Large Language Models: Surveying the landscape of diverse self-correction strategies. *arXiv:2308.03188*.
- Pang, C.; Cao, Y.; Ding, Q.; and Luo, P. 2023. Guideline Learning for In-Context Information Extraction. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Paul, D.; Ismayilzada, M.; Peyrard, M.; Borges, B.; Bosselut, A.; West, R.; and Faltings, B. 2023. RE-FINER: Reasoning Feedback on Intermediate Representations. *arXiv:2304.01904*.
- Pearl, J.; et al. 2000. Models, reasoning and inference. *Cambridge, UK: CambridgeUniversityPress*, 19(2): 3.
- Stoica, G.; Plataniotis, E. A.; and Póczos, B. 2021. Re-tacred: Addressing shortcomings of the tacred dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, 13843–13850.
- Sun, Q.; Huang, K.; Yang, X.; Tong, R.; Zhang, K.; and Poria, S. 2024. Consistency guided knowledge retrieval and denoising in llms for zero-shot document-level relation triplet extraction. In *Proceedings of the ACM on Web Conference 2024*, 4407–4416.
- Tan, Q.; Xu, L.; Bing, L.; Ng, H. T.; and Aljunied, S. M. 2022. Revisiting DocRED-Addressing the False Negative Problem in Relation Extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 8472–8487.
- Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.
- Wadhwa, S.; Amir, S.; and Wallace, B. 2023. Revisiting Relation Extraction in the era of Large Language Models. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 15566–15589. Toronto, Canada: Association for Computational Linguistics.
- Wan, Z.; Cheng, F.; Mao, Z.; Liu, Q.; Song, H.; Li, J.; and Kurohashi, S. 2023. GPT-RE: In-context Learning for Relation Extraction using Large Language Models. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 3534–3547. Singapore: Association for Computational Linguistics.
- Wang, F.; Mo, W.; Wang, Y.; Zhou, W.; and Chen, M. 2023a. A Causal View of Entity Bias in (Large) Language Models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.
- Wang, X.; Wei, J.; Schuurmans, D.; Le, Q. V.; Chi, E. H.; Narang, S.; Chowdhery, A.; and Zhou, D. 2023b. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations*.
- Wei, X.; Cui, X.; Cheng, N.; Wang, X.; Zhang, X.; Huang, S.; Xie, P.; Xu, J.; Chen, Y.; Zhang, M.; et al. 2023. Zero-shot information extraction via chatting with chatgpt. *arXiv preprint arXiv:2302.10205*.
- Wu, S.; and He, Y. 2019. Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM international conference on information and knowledge management*, 2361–2364.
- Xu, D.; Chen, W.; Peng, W.; Zhang, C.; Xu, T.; Zhao, X.; Wu, X.; Zheng, Y.; and Chen, E. 2023. Large Language Models for Generative Information Extraction: A Survey. *arXiv:2312.17617*.
- Yao, Y.; Ye, D.; Li, P.; Han, X.; Lin, Y.; Liu, Z.; Liu, Z.; Huang, L.; Zhou, J.; and Sun, M. 2019. DocRED: A Large-Scale Document-Level Relation Extraction Dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 764–777.
- Zhang, K.; Gutiérrez, B. J.; and Su, Y. 2023. Aligning Instruction Tasks Unlocks Large Language Models as Zero-Shot Relation Extractors. In *Findings of ACL*.
- Zhang, K.; Wang, D.; Xia, J.; Wang, W. Y.; and Li, L. 2023. ALGO: Synthesizing Algorithmic Programs with Generated Oracle Verifiers. *arXiv preprint arXiv:2305.14591*.
- Zhang, Y.; Zhong, V.; Chen, D.; Angeli, G.; and Manning, C. D. 2017. Position-aware Attention and Supervised Data Improve Slot Filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 35–45. Copenhagen, Denmark: Association for Computational Linguistics.