Reliable Data Generation and Selection for Low-Resource Relation Extraction

Junjie Yu[1], **Xing Wang**[2], **Wenliang Chen**[1*]

[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]Tencent AI Lab, Shenzhen, China
jjyu@stu.suda.edu.cn, brightxwang@tencent.com, wlchen@suda.edu.cn

Abstract

Automated construction of annotated data holds significant importance in Relation Extraction (RE) tasks due to the hardness and cost of human annotation. In this work, we propose Self-RDGS, a method for Self-supervised Reliable Data Generation and Selection in low-resource RE tasks. At first, we fully utilize the knowledge of triplets as prompts to generate sentences by employing the Large Language Models (LLMs). Since the auto-generated data contains noise, we then propose a ranking-based data selection method to select reliable sentences. Finally, we integrate the data selection and RE model training within a self-supervised iterative framework. Through experimentation on three datasets with low-resource settings, we demonstrate the effectiveness of our proposed approach in constructing annotated data and achieving noteworthy improvements in comparison to multiple baselines. Code, data and models are available at https://github.com/jjyunlp/GenerationRE.

Introduction

Relation Extraction (RE) aims to identify the pre-defined relations between two entities within a given sentence, thereby plays an important role in many Natural Language Processing (NLP) tasks (Zhou et al. 2005). Nowadays, training a neural network model using a large amount of annotated data stands as the prevailing and efficacious strategy for constructing an RE system (Zeng et al. 2015; Soares et al. 2019). However, obtaining a large amount of annotated data for RE tasks poses a challenge due to the diverse definitions of relations (Hendrickx et al. 2010; Zhang et al. 2017), and human annotation remains a laborious and costly process. Consequently, numerous studies have concentrated on the automatic construction of annotated data.

One possible solution is the utilization of Distant Supervision (DS), a widely employed method for automatically constructing annotated data, under the assumption that if two entities participate in a relation, any sentence that contains the two entities might express the relation (Mintz et al. 2009). During constructing the data, we often apply the exact match strategy to search for the entities. We can easily obtain lots of triplets (e.g. the ones from Freebase[1]) and free text, and consequently obtain a large amount of annotated sentences. However, the automatically annotated data from Exact Match-based Distant Supervision (EM-DS) contains significant noisy sentences, since the information of relations is completely ignored during matching (Lin et al. 2016; Ma et al. 2021).

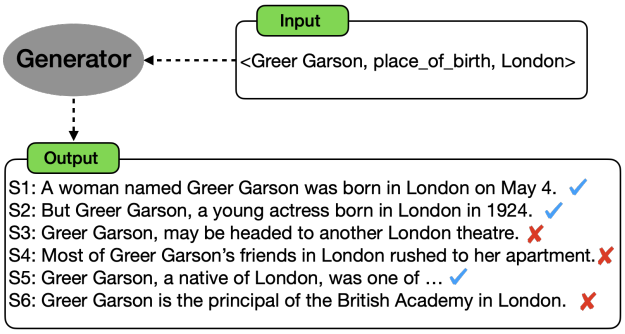

Figure 1: An example of sentence generation for a given triplet. While certain sentences express the correct semantic relation, there are noisy sentences.

Another solution to constructing annotated data for RE is through automatic text generation (ATG). Papanikolaou and Pierleoni (2020) proposes a method to train a generator for each relation by fine-tuning GPT-2 (Radford et al. 2019). Chia et al. (2022), on the other hand, suggests explicitly incorporating relations as a prefix constraint to prompt the generator in generating annotated sentences. These prior studies achieve a certain success in generating synthetic data for RE tasks through automatic text generation. However, their methods do not consider the knowledge of entities during the generation process that may result in noisy sentences. Thus, we argue that fully utilizing the information of triplets, including both entities and relations, can provide valuable guidance to the generator that might generate higher quality training data.

In order to address the aforementioned challenges, we propose a method called Self-supervised Reliable Data Generation and Selection (**Self-RDGS**). In our approach, we first

*The corresponding author is Wenliang Chen.

[1]https://developers.google.com/freebase

propose a novel data generation approach, Distant Supervision with Automatic Text Generation (ATG-DS), which fully uses the triplets to supervise the generation process, combining the merits of DS and ATG. In ATG-DS, we use the structured data guided prompt: the entities as the keywords and the relation as the constraint to build a generator based on Large Language Models (LLMs), which can make the generator more targeted. The preliminary experimental results show that our generator can provide better annotated sentences than the EM-DS and other ATG methods. However, the noise problem is still an issue in the data. As illustrated in Figure 1, for instance, some generated sentences express a relation different from "place_of_birth". For each triplet, the annotated sentences generated by ATG-DS are added to a candidate pool. We then propose a ranking-based data selection to choose reliable annotated sentences from the pool by considering the global distribution over the pool while the generator output the sentences independently. Here, we make an assumption that **"Most of the generated sentences in the pool express the relation in the target triplet, but some do not"**. Consequently, rather than using the entire pool, we convert the de-noising problem to selecting a representative sentence for each triplet from the sentence pool. We further design an iterative training framework to train our RE system.

In conclusion, this paper makes the following contributions:

- We propose a sentence generator ATG-DS, which utilizes the triplets to supervise the automatic text generation. Compared with DS and ATG, our generator can yield better annotated sentences by using both the information of entities and relations.
- We propose a ranking-based data selection method which can denoise the candidates generated by the ATG-DS. Our final system Self-RDGS is trained in a self-supervised iterative framework with the data selection procedure.
- Our experimental results on three datasets validate the effectiveness of the Self-RDGS approach. Furthermore, a detailed analysis of the experiments highlights the presence of noise in the generated data, which can be mitigated by our data selection method.

Methods

In this section, we begin by defining the task in this work, followed by a comprehensive description of our Self-RDGS method, which is divided into three main components: 1) the generation of auto-annotated data; 2) the selection of reliable data; and 3) the training of our RE model.

Task Formulation

In this work, our resources consist of the seed data, which comprises a small number of human-annotated sentences denoted as $D_{seed} = \{s_1, s_2, \cdots, s_n\}$. Additionally, we utilize a large set of triplets T that contains relation facts organized in a triplet format $\{h, r, t\}$, where h, r, t represent the head entity, relation name, and tail entity, respectively.

Algorithm 1: Sentence Selection and Training

Input: seed train D_{seed}, triplets T and Generator $\mathbf{M_g}$.
Hyper Parameter: number of sentences generated for each triplet K.
Output: Selected Data D_{sel} and Relation Extractor $\mathbf{M_{re}}$.

$D_{gen} \leftarrow Generate(\mathbf{M_g}, D_{seed}, T, K)$
while Training **do**
  $D_{sel} \leftarrow \emptyset$
  for $t \in T$ **do**
    $x^t \leftarrow DataSelection(\mathbf{M_{re}}, D^t_{gen})$
    $D_{sel} \overset{+}{\leftarrow} x_t$ // Append
  end for
  $\mathbf{M_{re}} \leftarrow Train(\mathbf{M_{re}}, D_{sel})$ // One Epoch
  if End Training **then**
    break
  end if
end while
return $\mathbf{M_{re}}$, D_{sel}

Algorithm 1 formalizes the Self-RDGS proposed in this work. Its purpose is to automatically generate annotated data from triplets and select reliable sentences using our ranking-based data selection method. This process ultimately yields a RE model along with the selected annotated data.

Our study focuses on the automatic text generation for RE tasks. In other words, the data we generate holds applicability across various supervised RE models. To simplicity, we follow the common used supervised RE model training approach in previous studies (Soares et al. 2019; Yu et al. 2022) which fine-tunes a pre-trained model with a task-specific classification layer. In detail, we employ BERT (Devlin et al. 2018) as the pre-trained model and utilize the "entity-marker" method to wrap entity mentions within sentences. For instance, "... [E1] head words [/E1] ... [E2] tail words [/E2] ...". After encoding the sentence, the classifier takes the concatenated representation of token "[E1]" and "[E2]" as input to predict the probabilities of pre-defined relations. Formally, to predict a sentence s from N relations using model $\mathbf{M_{re}}$, we begin by obtaining representations $\mathbf{h}$ for each token in the sentence s through the sentence encoder in $\mathbf{M_{re}}$. Next, we extract a concatenated representation of two entity markers to serve as a relational representation:

$$f(s) = f([E1], [E2]|s) = \mathbf{h}_{[E1]} \oplus \mathbf{h}_{[E2]}. \quad (1)$$

Finally, the relational representation $f(s)$ is fed into the classifier within $\mathbf{M_{re}}$ to output the predicted probability distribution $p(s) = [p_1, p_2, ..., p_N]$ for the N pre-defined relations, according to the following function:

$$p(s) = \text{Softmax}(\mathbf{W} \cdot f(s) + \mathbf{b}), \quad (2)$$

where $\mathbf{W}$ and $\mathbf{b}$ are trainable model parameters.

Data Generation

In recent studies, it has been discovered that Large Language Models (LLMs) possess the ability to generate text with constraints when provided with well-designed prompts. Therefore, we study the solution of taking triplets as input and

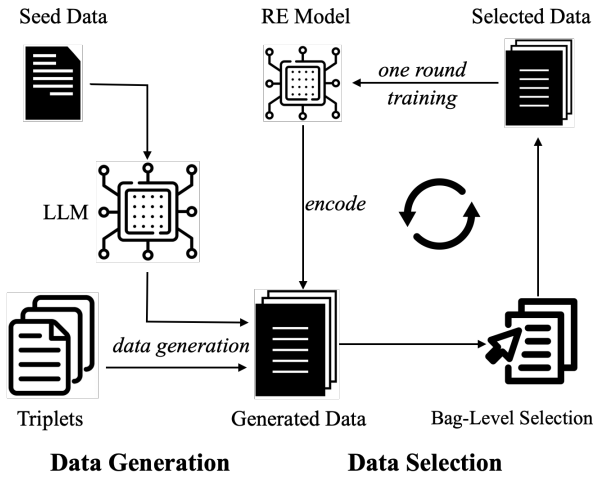

Figure 2: The framework of our Self-RDGS.

LLMs as data generator for RE tasks. More specifically, we apply the idea of distant supervision to the procedure of automatic text generation in LLMs. In this paper, we build our data generator in two modes.

In-Context Learning based Generation In this mode, we apply the In-Context Learning (ICL) based generation methods on LLMs. In detail, we design the following instruction to prompt the LLMs for generating a relational sentence. Based on the given triplet, we use the entities as the keywords and the relation as the constraints:

```
Follow 5 examples then write a sentence
expressing the given relation between the
head word and tail word. The sentence must
contain the given head word and tail word.
Relation: r, Head: h, Tail: t, Context: s.
...
Relation: r, Head: h, Tail: t, Context:
```

For each given triplet, we randomly select 5 samples from the seed data that share the same relation r but have different head and tail pairs.

Fine-Tune based Generation We also take fine-tuned Pre-trained Language Models (PLMs) (e.g., GPT-2) as a comparison. To train the PLMs with the ability of generating relational sentences, we add an initial step of our approach involves fine-tuning the PLM using the seed data D_{seed}. Specifically, we convert the samples within D_{seed} into plain text instance x using the following template:

```
Relation: r, Head: h, Tail: t, Context: s.
```

This template incorporates tokens from the relation name r, head entity h, tail entity t, and the annotated sentence s. To enhance the readability of relation names, we undertake a conversion process by transforming pre-defined relations into more intelligible representations. For instance, we convert "cause-effect" to "cause - effect" and "per:title" to "person : title", and so on. During the training phase, we adopt the standard language modeling objective in the causal language model, which involves performing next-word prediction (Bengio, Ducharme, and Vincent 2000) on each training instance.

For both ICL based and fine-tune based generation modes, we add a post-filtration on the generated sentences where the generated sentences must contain the given entities. In total, we try to obtain a maximum of $K = 8$ sentences into the sentence pool for each triplet in triplet set T.

Data Selection

To mitigate the noise problem in auto-generated data, we propose a ranking-based data selection method to identify reliable sentences. Building upon our assumption that **"Most of the generated sentences in the pool express the relation in the target triplet, but some do not"**, our objective is to choose a representative sentence that captures the target relation from the sentences pool.

Formally, we begin by acquiring the relational representations $\{f(s_1), f(s_2), ..., f(s_K)\}$ for K sentences in the sentence pool $S = \{s_1, ..., s_K\}$ corresponding to a given triplet from T. The function $f()$, as defined in Eqn 1, serves as a relational encoder within the trained RE model. Then, we calculate the global distribution score of each sentence s_i to rank the sentences in the pool. We expect the global distribution score to measure the representativeness of individual sentences in the sentence pool. In this paper, we define the global distribution score as the averaged cosine similarity between s_i and every other sentence in S:

$$score(s_i, f, S) = \frac{1}{|S|} \sum_{s_j \in S} cos(f(s_i), f(s_j)). \quad (3)$$

Additionally, we calculate the sentence with itself to cover the special situation in which a sentence pool consists of only one sentence. Finally, we rank the candidates according to the global distribution scores and select the sentence s^* with the highest score in S:

$$s^* = \underset{i \in \{1,...,K\}}{\arg\max}\ score(s_i, f, S). \quad (4)$$

Iterative Training Framework

Our data selection method leverages the sentence distribution within the sentence pool, aided by an encoder from the RE model. To enhance this procedure, our Self-RDGS approach involves a self-supervised iterative procedure in which data selection is incorporated into the training of the RE model at each epoch, as illustrated in Figure 2.

The process begins by obtaining initial selected data through sentence encoding using the raw pre-trained BERT. Considering each pass over the selected data as one iteration, we train the RE model for one round on the selected data and the newly trained RE model is used to update the selected data for the subsequent training epoch. Finally, we obtain the best RE model by evaluating its performance on the validation set and output the selected data.

Experiments

Datasets and Low-Resource Setting

To verify our Self-RDGS approach, we conduct experiments on three datasets, including two human-annotated datasets

and one DS-annotated dataset.

SemEval A human-annotated dataset from SemEval-2010 Task 8 (Hendrickx et al. 2010), including 10,717 sentences covering 9 bidirectional relations and 1 special "NA" relation.

Re-TACRED A revised version of the human-annotated dataset TACRED (Zhang et al. 2017) proposed by (Stoica, Platanios, and Póczos 2021) . There are 91,467 sentences for 39 common relations and 1 special "NA" relation which means none of the above relations.

NYT10m An updated version of the widely used DS dataset NYT10 (Riedel, Yao, and McCallum 2010) with a human-annotated test set. In total, there are 474,059 sentences for 25 relations. [2]

To emulate low-resource scenarios across three datasets, we adopt the setup in Yu et al. (2022). The low-resource statistics of these datasets are detailed in Table 1, where we randomly select 20 human-annotated sentences per relation as the seed data. Considering NYT10m, which employs EMDS labeling for training data but with a human-annotated test set, we retain the relations that own at least 50 sentences in the original test set. This selection facilitates the extraction of human-annotated seed data, while the remaining serves as test data. Besides, we focus on the sentence generation of meaningful relation types, hence the special "NA" relation is excluded in this work. To enhance the realistic of low-resource scenarios, we do not create a separate validation set in our approach. Instead, the seed data serves as the validation set while the automatically generated sentences serves as training data. Triplet resources, typically hailing from Knowledge Graphs such as Freebase, are commonly employed. In order to enable a fair comparison between our generated data and the existing human-annotated data, the triplets in this paper are derived from the unused training sentences. In summary, the resources we use include the seed data and the existing triplets.

Parameter Settings

Data Generation To select the LLMs for in-context learning based generation, we employ two recently released open LLMs: `ChatGLM2-6B` (released in June 2023) and `LLaMa2-7B-chat` (released in July 2023). During the sentence generation phrase, we set a limit of generating a maximum of $K = 8$ sentences for each triplet. For fine-tune based generation, we employ $\texttt{GPT-2}_{\texttt{large}}$ (Radford et al. 2019) to train the sentence generator. The fine-tuning process follows the hyper-parameters specified in Chia et al. (2022).

Relation Extraction Training We utilize $\texttt{BERT}_{\texttt{base}}$ (Devlin et al. 2018) to build the RE models. Throughout the training process, we set the learning rate to 5e-5 and maintain a batch size of 32, according to the performance on the validation set. The model is trained for a maximum of 20 epochs, and early stopping is determined by the validation performance. Additionally, we conduct 5 runs with different seeds for each system to report the averaged Micro-F1 score, along with the corresponding standard deviation.

Dataset	#Rel	#Seed	#Test	#Tri	#Sen
SemEval	9	180	2,263	6,053	6,319
Re-TACRED	39	780	5,648	9,180	18,938
NYT10m	10	200	5,985	6,647	8,664

Table 1: The statistics of low-resource settings for three datasets. #Rel, #Seed, #Test, #Tri and #Sen are the number of relations, seed data, test set, triplets, and original annotated sentences.

Baselines

To ensure a fair comparison between baselines and our approach, we employ the same supervised RE framework. The different comparison systems are described below:

SUPERVISED The most basic baseline system involves using the original annotated data as the training set. This method has two systems: with Low Resource using the seed data as the training set, and with High Resource using the full training set. The former plays the role of the bottom bound and the latter is the upper bound. However, this approach lacks annotated data for validation. To address this limitation, we adopt an additional validation set consisting of 10 sentences per relation for the SUPERVISED baselines. [3]

LLMPREDICT Following generative models for RE (Wan et al. 2023; Wadhwa, Amir, and Wallace 2023; Zhu et al. 2023), we apply `LLaMa2-7B-chat` to predict relations through a generative approach with the help of in-context learning (Brown et al. 2020). The prompt is as follows:

```
Follow 5 examples then predict the
relation that expresses the semantic
relationship between head and tail words
in a given context. Candidate relations
are <all_relations>.
Context: s, Head: h, Tail: t, Relation: r.
...
Context: s, Head: h, Tail: t, Relation:
```

Meanwhile, we also employ the $\texttt{GPT-2}_{\texttt{large}}$ model as a comparison to predict relations by fine-tuning on seed data. During the prediction phase, we discard cases where the generated text for the r slot does not correspond to a valid relation name within the pre-defined relations. Additionally, drawing inspiration from Wang et al. (2022), we introduce the voting mechanism for candidate generated relations. Specifically, we generate 8 valid predictions for each test sentence and select the one that predicted most frequently.

STAD We compare our approach with the self-training based RE system proposed by Yu et al. (2022). The primary distinction lies in the utilization of resources: they utilize

[2] https://github.com/thunlp/OpenNRE

[3] Please note that this extra validation set is not utilized in our proposed methods.

Datasets		SemEval		Re-TACRED		NYT10m		Avg.
Data Source	Data Selection	Val	Test	Val	Test	Val	Test	Test
GPT-2-large	MERGEALL	$77.2_{\pm 0.9}$	$79.4_{\pm 0.7}$	$81.1_{\pm 0.5}$	$80.2_{\pm 0.9}$	$65.9_{\pm 1.0}$	$68.4_{\pm 1.5}$	76.0
	FIRSTONE	$83.4_{\pm 1.0}$	$83.0_{\pm 0.1}$	$77.8_{\pm 0.8}$	$86.0_{\pm 0.9}$	$71.1_{\pm 2.1}$	$69.0_{\pm 3.8}$	79.3
	RANDONE	$83.7_{\pm 2.3}$	$82.6_{\pm 0.6}$	$79.6_{\pm 0.5}$	$86.1_{\pm 0.9}$	$\mathbf{71.3}_{\pm 0.9}$	$68.9_{\pm 2.2}$	79.2
	MINONE	$81.2_{\pm 1.1}$	$81.2_{\pm 0.5}$	$78.9_{\pm 0.5}$	$83.0_{\pm 1.4}$	$68.4_{\pm 1.2}$	$69.7_{\pm 2.2}$	78.0
	MAXONE	$\mathbf{84.2}_{\pm 0.9}$	$\mathbf{84.9}_{\pm 0.7}$	$\mathbf{83.5}_{\pm 0.6}$	$\mathbf{86.8}_{\pm 0.8}$	$71.2_{\pm 1.9}$	$\mathbf{71.4}_{\pm 2.5}$	**81.0**
LLaMa2-7B-chat	MERGEALL	$88.8_{\pm 1.0}$	$86.1_{\pm 0.9}$	$\mathbf{89.3}_{\pm 0.4}$	$87.0_{\pm 0.8}$	$\mathbf{82.1}_{\pm 1.5}$	$70.3_{\pm 2.2}$	81.1
	FIRSTONE	$91.0_{\pm 0.9}$	$87.4_{\pm 0.6}$	$85.3_{\pm 0.4}$	$87.7_{\pm 0.7}$	$79.4_{\pm 1.8}$	$72.1_{\pm 2.1}$	82.4
	RANDONE	$90.9_{\pm 0.9}$	$87.6_{\pm 0.6}$	$88.2_{\pm 0.5}$	$87.2_{\pm 0.6}$	$76.0_{\pm 1.2}$	$73.1_{\pm 1.6}$	82.6
	MINONE	$90.6_{\pm 0.8}$	$87.3_{\pm 0.6}$	$86.9_{\pm 0.9}$	$87.5_{\pm 0.9}$	$78.6_{\pm 1.3}$	$71.7_{\pm 2.7}$	82.2
	MAXONE	$\mathbf{91.2}_{\pm 0.5}$	$\mathbf{88.4}_{\pm 0.4}$	$86.0_{\pm 1.3}$	$\mathbf{88.7}_{\pm 0.6}$	$77.2_{\pm 0.5}$	$\mathbf{74.0}_{\pm 2.0}$	**83.7**

Table 2: The results of our Self-RDGS framework with different data selection methods.

sentences without relation annotations, whereas we employ triplets without sentence annotations.

RELATIONPROMPT Following Chia et al. (2022), we copy their method into the LLMs scenario. In detail, RELATIONPROMPT takes each relation name as prompt and let `LLaMa2-7B-chat` ouput sentences with entities under the ICL method.

Results of Different Data Selection Methods

In order to verify the effectiveness of our data selection method in SELF-RDGS, we conduct experiments on different data selection methods on two data sources: the generated data from $\texttt{GPT-2}_{\texttt{large}}$ and `LLaMa2-7B-chat`, as shown in Table 2. The comparison data selection methods are as follows:

- **MERGEALL**: Without selection, we merge all the generated sentences in the sentence pool, allowing each triplet to have a maximum of 8 sentences.
- **FIRSTONE**: As the sentences in the sentence pool are ordered by the sequence of generation, this method naturally selects the first generated sentence for each triplet.
- **RANDONE**: We randomly select one generated sentence from the sentence pool to build the training data.
- **MINONE**: As a comparison, this method chooses a sentence with the lowest global distribution score in the sentence pool.
- **MAXONE**: The data selection method in our SELF-RDGS, which selects a sentence with the highest global distribution score in the sentence pool for each triplet.

MERGEALL vs Others Among all the data selection methods on generated data, MERGEALL, which utilizes all the sentences in the sentence pool, achieves the lowest performance on two data sources, respectively. This result indicates the presence of noise in the generated data, highlighting the importance of our data selection method.

FIRSTONE vs RANDONE To investigate the influence of generation order, we compare the results between FIRSTONE and RANDONE. Both methods yield comparable results across the three datasets, with nearly identical average performance (79.3% vs 79.2% on $\texttt{GPT-2}_{\texttt{large}}$ and 82.4% vs 82.6% on `LLaMa2-7B-chat`). This indicates that sentences are generated independently and the order of sentence generation has no discernible impact on the data quality.

MAXONE vs MINONE Among various selection methods, our MAXONE achieves the highest performance. Specifically, when comparing the two opposite methods, the MAXONE consistently outperforms the MINONE across all corpora, exhibiting an average absolute improvement of 3.0% (78.0% vs. 81.0%) on $\texttt{GPT-2}_{\texttt{large}}$ and 1.5% (82.2% vs 83.7%) on `LLaMa2-7B-chat`. Furthermore, MINONE even performs worse than RANDONE (78.0% vs 79.2% and 82.2% vs 82.6%). These results indicate that our ranking-based data selection method effectively distinguishes high-quality data from low-quality data.

Main Results

As shown in Table 3, we analyze the experimental results from the following aspects:

SELF-RDGS vs Baselines We first compare our SELF-RDGS method to three baselines, all of which share comparable data size: SUPERVISED in the high-resource setting, STAD, and RELATIONPROMPT. From the table, we observe that our method effectively narrows the performance gap when compared to results of human-annotated data (SemEval and Re-TACRED), especially for the results from SELF-RDGS with `LLaMa2-7B-chat`(88.4% vs 95.4, 88.7% vs 94.3). Moving on to the EM-DS data NYT10m, our ATG-DS demonstrates superior performance compared to the original EM-DS method (74.0% vs 64.0%). Notably, in the case of STAD and RELATIONPROMPT, the absence of triplet-based supervision results in a performance discrepancy. Specifically, the generation-based approach of RELATIONPROMPT proves less effective than the self-training based approach STAD (59.5% vs 74.8%). However, with the infusion of supervision from triplets, our SELF-RDGS generates data of enhanced quality, leading to substantial overall improvement compared to STAD (83.7% vs 74.8%).

SUPERVISED vs LLMPREDICT In the low-resource scenario, LLMPREDICT yields poor performance when com-

Datasets		SemEval		Re-TACRED		NYT10m		Avg.
Methods	Data Source	Val	Test	Val	Test	Val	Test	Test
SUPERVISED								
Low Resource	None	-	$78.3_{\pm 1.7}$	-	$77.9_{\pm 1.6}$	-	$54.8_{\pm 2.5}$	70.3
High Resource	Human/DS	$\mathbf{95.3}_{\pm 0.6}$	$\mathbf{95.4}_{\pm 0.7}$	$\mathbf{89.3}_{\pm 0.8}$	$\mathbf{94.3}_{\pm 0.3}$	$76.0_{\pm 0.7}$	$64.0_{\pm 3.3}$	**84.5**
LLMPREDICT								
GPT-2-large	None	-	31.4	-	70.8	-	16.9	39.7
LLaMa2-7B-chat	None	-	33.4	-	39.6	-	15.9	29.6
STAD	Self Training	-	$82.9_{\pm 2.0}$	-	$81.6_{\pm 0.3}$	-	$60.0_{\pm 1.8}$	74.8
RELATIONPROMPT	LLaMa2-7B-chat	$81.7_{\pm 1.7}$	$70.2_{\pm 2.7}$	$78.2_{\pm 1.3}$	$67.2_{\pm 3.8}$	$78.5_{\pm 1.5}$	$41.2_{\pm 2.5}$	59.5
SELF-RDGS (ours)	GPT-2-large	$84.2_{\pm 0.9}$	$84.9_{\pm 0.7}$	$83.5_{\pm 0.6}$	$86.8_{\pm 0.8}$	$71.2_{\pm 1.9}$	$71.4_{\pm 2.5}$	81.0
	ChatGLM2-6B	$92.2_{\pm 1.2}$	$85.4_{\pm 0.5}$	$83.8_{\pm 0.9}$	$83.8_{\pm 2.5}$	$\mathbf{78.6}_{\pm 1.3}$	$70.7_{\pm 1.9}$	80.0
	LLaMa2-7B-chat	$91.2_{\pm 0.5}$	$88.4_{\pm 0.4}$	$86.0_{\pm 1.3}$	$88.7_{\pm 0.6}$	$77.2_{\pm 0.5}$	$\mathbf{74.0}_{\pm 2.0}$	83.7

Table 3: Results of different systems on three datasets. An averaged Micro-F1 (%) score is reported with the standard deviation on 5 seeds. For LLMPREDICT, we use the vote mechanism on multiple candidate predictions for the test set, so the results for validation set and the standard deviation are omitted. We run the STAD on SemEval and NYT10m with our low-resource setting and copy the result in Yu et al. (2022) for Re-TACRED. We omit the results on validation set for SUPERVISED with low resource mode and STAD because they use an extra validation set.

pared to SUPERVISED, no matter through fine-tuning on $\texttt{GPT-2}_{\texttt{large}}$ (39.7%) or in-context learning on `LLaMa2-7B-chat` (29.6%). This observation underscores the inadequacy of employing vanilla ICL method on LLMs for a direct prediction of relational labels in RE tasks.

SELF-RDGS with Difference Generators Upon analyzing the results of our SELF-RDGS across various generators, a noteworthy observation emerges. The fine-tuned small-sized model, $\texttt{GPT-2}_{\texttt{large}}$, achieves a competitive performance when compared to two Large Models): `ChatGLM2-6B` and `LLaMa2-7B-chat` (81.0% vs 80.0% and 83.7%). This finding underscores the versatility of our approach, showcasing its efficacy not only on large-sized models but also on their smaller counterparts.

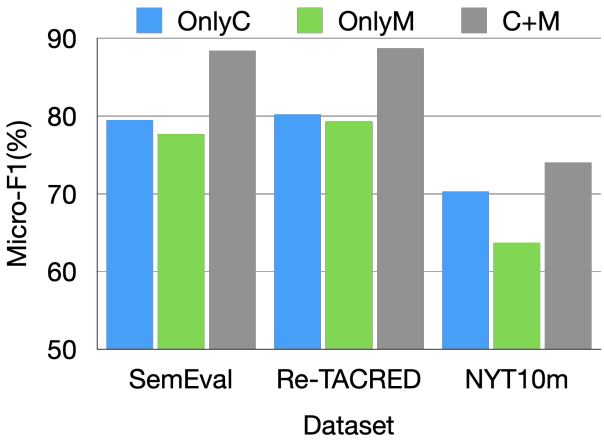

Figure 3: The results of different masking methods.

Discussion

Does the Improvement Come From Triplets?

When analyzing the training data, it is essential to understand the contribution of two key components: the newly added relation triplets and the generated sentences. It is natural to question whether the performance improvement is primarily attributed to the information of triplets or the context of generated sentences.

To address this question, we conduct experiments on the selected training data (generated by `LLaMa2-7B-chat`) using two different text modes: the context-only mode (**OnlyC**) and the mention-only mode (**OnlyM**). Following Peng et al. (2020), we mask all tokens related to mentions (head and tail entities) in the context-only mode, while in the mention-only mode, we mask all context tokens but retain mention tokens. Additionally, the **C+M** mode employs all tokens in the sentences serves as the standard benchmark.

The results for the three datasets are illustrated in Figure 3. It is evident from the figure that the performance of both **OnlyC** and **OnlyM** drop a lot because they lack the information of mentions or contexts, respectively. When comparing the performance between **OnlyC** and **OnlyM**, the former performs better than the latter on SemEval and NYT10m and they achieve similar performance on Re-TACRED.

In conclusion, both the triplets and the contexts of generated sentences play crucial roles in training the final relation RE system. The performance improvement is not solely attributed to the newly added triplets but also to the sentences generated by our generator.

Is Generated Data Better Than DS Data?

Among the three datasets used in this work, the original training data of NYT10m is constructed using the EM-DS method. Since both the EM-DS data and our ATG-DS data are constructed based on triplets, we proceed to conduct ad-

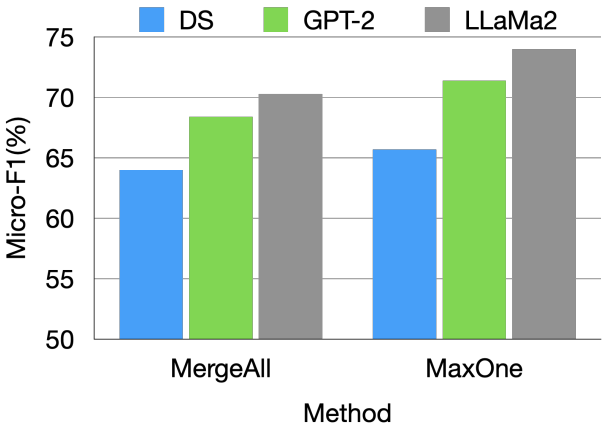

Figure 4: The results of utilizing data selection method on the original EM-DS data and our ATG-DS data for NYT10m.

ditional comparisons between these two methods. The results of three data (EM-DS, ATG-DS from $\texttt{GPT-2}_{\texttt{large}}$ and `LLaMa2-7B-chat`) on two data selection methods are presented in Figure 4. Firstly, when considering the utilization of all data without selection (MERGEALL), wherein one triplet may have multiple sentences, it becomes evident that the performance of our ATG-DS data notably outperforms that of the EM-DS data. We attribute this disparity to the substantial presence of noise within the EM-DS data. After applying our data selection method (MAXONE) on the EM-DS data, it can be observed that our MAXONE yields little improvements on the EM-DS data while the gap between the EM-DS and ATG-DS further widens.

To investigate the possible reasons behind these observations, we examine the distribution of sentences for each triplet in both datasets. Our examination reveals that only 11.2% of the triplets in the EM-DS data of NYT10m are associated with more than one supporting sentence, indicating that the data selection method fails to operate on the majority of triplets. In contrast, our generated data exhibits an opposite distribution, with more than 98% of the triplets having multiple supporting sentences. This finding suggests that ATG-DS inherently alleviate the long-tail problem commonly observed in EM-DS scenarios. This holds significant importance as the long-tail problem can influence the effectiveness of data selection methods. Given these analyses, we conclude that our ATG-DS data outperforms the EM-DS data in terms of both data quality and data coverage.

Related Work

Distant Supervision To build annotated data for RE, Mintz et al. (2009) first proposes a exact match distant supervision (EM-DS) method which assumes every sentence that mentions two related entities in a relational triplet expresses the corresponding relation. Riedel, Yao, and McCallum (2010) further introduces multi-instance learning in distant supervision RE, relaxing the assumption to "at least one sentence mentioning two related entities expresses the corresponding relation." Subsequently, numerous researchers focus on enhancing distant supervision RE by addressing the noise (Lin et al. 2016; Vashishth et al. 2018; Ye and Ling 2019; Zhu et al. 2020; Ma et al. 2021). Despite the advancements achieved by previous studies, the serve noise problem of EM-DS data continues to hinder the progress of RE. This limitation is primarily attributed to the oversight of relational information during data construction. In order to build high-quality data, we propose an alternative approach ATG-DS. This novel approach embraces the generation ability of LLMs and the information of both entities and relations.

Automatic Text Generation In recent years, there has been growing interest among researchers in automatic text generation for various NLP tasks (Anaby-Tavor et al. 2020; Yu et al. 2020; Schick and Schütze 2021; Ross et al. 2022; Ye et al. 2023). For RE tasks, Papanikolaou and Pierleoni (2020) proposes the use of fine-tuned GPT-2 to generate new samples for a given relation. Without any prompts about relation and entity information, they assign the relation label for the generated sentences by train one generator per relation. To explicitly integrate the relation information, Chia et al. (2022) proposes RelationPrompt, which trains GPT-2 with the ability of generating sentences for the given relation names. Recently, Josifoski et al. (2023) queries OpenAI's GPT-3.5 in an "instruction + demonstration" mode to generate data from sampled triplet sets for closed information extraction tasks. In our work, we propose to fully utilize the knowledge of entities and relations to guide LLMs in generating sentences for RE tasks. According to the global distribution of generated sentences for each triplet, we further propose a ranking-based data selection method to select high-quality sentences in an iterative framework.

Conclusion

In this paper, we propose a novel approach called Self-supervised Reliable Data Generation and Selection (Self-RDGS), aimed at constructing high-quality training data for low-resource RE tasks. Our method combines the merits of distant supervision and automatic text generation. To address the challenge of noise inherent in generated data, we propose a ranking-based selection method within an iterative framework during RE model training. Through comprehensive experiments, we evaluate various variants of our approach on three datasets. The experimental results demonstrate that our approach consistently and significantly outperforms the baselines, highlighting its efficacy in constructing annotated data for RE tasks. Notably, our method outperforms exact match based distant supervision methods in terms of data quality and coverage.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (62376177, 61936010) and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions. We thank the anonymous reviewers for their constructive comments.

References

Anaby-Tavor, A.; Carmeli, B.; Goldbraich, E.; Kantor, A.; Kour, G.; Shlomov, S.; Tepper, N.; and Zwerdling, N. 2020. Do not have enough data? Deep learning to the rescue! In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 7383–7390.

Bengio, Y.; Ducharme, R.; and Vincent, P. 2000. A neural probabilistic language model. *Advances in neural information processing systems*, 13.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chia, Y. K.; Bing, L.; Poria, S.; and Si, L. 2022. RelationPrompt: Leveraging Prompts to Generate Synthetic Data for Zero-Shot Relation Triplet Extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, 45–57.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Hendrickx, I.; Kim, S. N.; Kozareva, Z.; Nakov, P.; Séaghdha, D. Ó.; 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 *SemEval*.

Josifoski, M.; Sakota, M.; Peyrard, M.; and West, R. 2023. Exploiting asymmetry for synthetic training data generation: Synthie and the case of information extraction. *arXiv preprint arXiv:2303.04132*.

Lin, Y.; Shen, S.; Liu, Z.; Luan, H.; and Sun, M. 2016. Neural Relation Extraction with Selective Attention over Instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2124–2133. Berlin, Germany: Association for Computational Linguistics.

Ma, R.; Gui, T.; Li, L.; Zhang, Q.; Huang, X.-J.; and Zhou, Y. 2021. SENT: Sentence-level Distant Relation Extraction via Negative Training. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 6201–6213.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, 1003–1011.

Papanikolaou, Y.; and Pierleoni, A. 2020. Dare: Data augmented relation extraction with gpt-2. *arXiv preprint arXiv:2004.13845*.

Peng, H.; Gao, T.; Han, X.; Lin, Y.; Li, P.; Liu, Z.; Sun, M.; and Zhou, J. 2020. Learning from Context or Names? An Empirical Study on Neural Relation Extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3661–3672.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.

Riedel, S.; Yao, L.; and McCallum, A. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III 21*, 148–163. Springer.

Ross, A.; Wu, T.; Peng, H.; Peters, M.; and Gardner, M. 2022. Tailor: Generating and Perturbing Text with Semantic Controls. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 3194–3213. Dublin, Ireland: Association for Computational Linguistics.

Schick, T.; and Schütze, H. 2021. Generating Datasets with Pretrained Language Models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6943–6951.

Soares, L. B.; Fitzgerald, N.; Ling, J.; and Kwiatkowski, T. 2019. Matching the Blanks: Distributional Similarity for Relation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2895–2905.

Stoica, G.; Platanios, 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.

Vashishth, S.; Joshi, R.; Prayaga, S. S.; Bhattacharyya, C.; and Talukdar, P. 2018. RESIDE: Improving Distantly-Supervised Neural Relation Extraction using Side Information. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1257–1266.

Wadhwa, S.; Amir, S.; and Wallace, B. C. 2023. Revisiting relation extraction in the era of large language models. *arXiv preprint arXiv:2305.05003*.

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. *arXiv preprint arXiv:2305.02105*.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; and Zhou, D. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Ye, J.; Jiao, W.; Wang, X.; and Tu, Z. 2023. Scaling Back-Translation with Domain Text Generation for Sign Language Gloss Translation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, 463–476.

Ye, Z.; and Ling, Z. 2019. Distant Supervision Relation Extraction with Intra-Bag and Inter-Bag Attentions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2810–2819.

Yu, J.; Wang, X.; Zhao, J.; Yang, C.; and Chen, W. 2022. STAD: Self-Training with Ambiguous Data for

Low-Resource Relation Extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, 2044–2054.

Yu, J.; Zhu, T.; Chen, W.; Zhang, W.; and Zhang, M. 2020. Improving Relation Extraction with Relational Paraphrase Sentences. In Scott, D.; Bel, N.; and Zong, C., eds., *Proceedings of the 28th International Conference on Computational Linguistics*, 1687–1698. Barcelona, Spain (Online): International Committee on Computational Linguistics.

Zeng, D.; Liu, K.; Chen, Y.; and Zhao, J. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 conference on empirical methods in natural language processing*, 1753–1762.

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.

Zhou, G.; Su, J.; Zhang, J.; and Zhang, M. 2005. Exploring various knowledge in relation extraction. In *Proceedings of the 43rd annual meeting of the association for computational linguistics*, 427–434.

Zhu, T.; Ren, J.; Yu, Z.; Wu, M.; Zhang, G.; Qu, X.; Chen, W.; Wang, Z.; Huai, B.; and Zhang, M. 2023. Mirror: A Universal Framework for Various Information Extraction Tasks. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 8861–8876. Singapore: Association for Computational Linguistics.

Zhu, T.; Wang, H.; Yu, J.; Zhou, X.; Chen, W.; Zhang, W.; and Zhang, M. 2020. Towards Accurate and Consistent Evaluation: A Dataset for Distantly-Supervised Relation Extraction. In Scott, D.; Bel, N.; and Zong, C., eds., *Proceedings of the 28th International Conference on Computational Linguistics*, 6436–6447. Barcelona, Spain (Online): International Committee on Computational Linguistics.