LLM as a metric critic for low resource relation identification

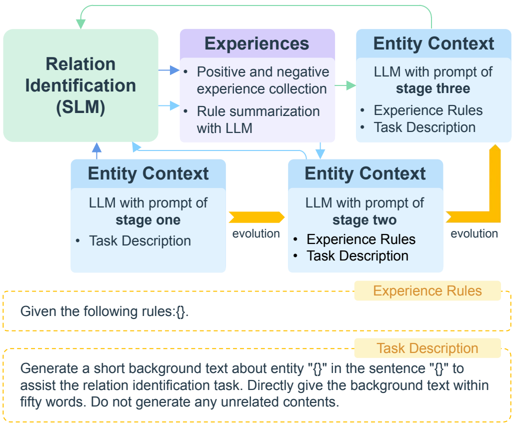
Zhe Yang,Yi Huang[*], Yaqin Chen, Xiaoting Wu, Junlan Feng and Chao Deng
JIUTIAN Team, China Mobile Research Institute
{yangzhe,huangyi,chenyaqin,wuxiaoting,fengjunlan,dengchao}@chinamobile.com

Abstract

In extremely low resource relation identification scenario, small language models (SLMs) incline to overfit, which significantly diminishes their accuracy. Recently, large language models (LLMs) are gradually applied to classification tasks with converting original objective into the generation task via in-context learning. However, abundance of the classifier categories poses challenges in selecting demonstrations. Moreover, the mapping between category labels and textual descriptions requires expensive expert knowledge, thereby constraining the efficacy of in-context learning for LLMs. We uphold that SLM is optimal for handling classification tasks, and its shortcomings in the low resource setting can be mitigated by leveraging LLM. Hence, we propose a co-evolution strategy on SLM & LLM for relation identification. Specifically, LLM provides essential background knowledge to assist training process of the SLM classifier, while evaluation metrics from the classifier, in turn, offer valuable insights to refine the generation prompts of the LLM. We conduct experiments on several datasets which demonstrates preponderance of the proposed model.

1 Introduction

Relation identification aims to identify target relationship between a specify entity pair mentioned in a text. Pre-trained language models (PLMs) (Devlin et al., 2019; Liu et al., 2019) are capable of absorbing and representing a wealth of knowledge from extensive data. Fine-tuning PLMs has been demonstrated an effective approach for relation identification. However, the gap between the objectives of pre-training and fine-tuning often lead to performance decay in low resource scenarios. After that, prompt tuning is proposed to bridge this gap (Gao et al., 2021a; Chen et al., 2022a,b; Li et al., 2024). Concretely, a masked language modeling (MLM) prompt is constructed and concatenated with the given text as input (e.g., "He has a sister Lisa. The relation between ‘He’ and ‘Lisa’ is [MASK]."). The relation identification task can be transferred into a MLM problem by filling the [MASK] token in the input. Recently, LLM has shown remarkable abilities. There are works using LLM for relation identification via generation (Wadhwa et al., 2023; Ma et al., 2023; Pang et al., 2023; Wan et al., 2023) and comparing its performance with SLM (Ma et al., 2023). The results show that LLM succeeds SLM only when the annotation type is extremely limited, i.e., both relation label and samples for each category are extremely scarce. When the number of samples increases slightly, SLM significantly outperforms LLM. Furthermore, due to the constraint of prompt length, LLM is unable to deal with lengthy texts and a large number of relations. Despite these limitations, the question remains: How can the extensive knowledge encapsulated within LLM be leveraged to enhance relation identification tasks? Driven by this inquiry, we propose to harness the LLM as a reservoir of knowledge, offering a com-


Relation Identification (SLM)
Experiences
• Positive and negative experience collection
• Rule summarization with LLM
Entity Context
LLM with prompt of **stage three**
• Experience Rules
• Task Description
Entity Context
LLM with prompt of **stage one**
• Task Description
evolution
Entity Context
LLM with prompt of **stage two**
• Experience Rules
• Task Description
evolution
Experience Rules
Task Description

Given the following rules:{}.

Generate a short background text about entity "{}" in the sentence "{}" to assist the relation identification task. Directly give the background text within fifty words. Do not generate any unrelated contents.

Figure 1: Co-evolution framework operated on SLM & LLM. The three consecutive stages are shown by corresponding colored thin arrows.

[*]Corresponding authors

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 14168–14178
November 12-16, 2024 ©2024 Association for Computational Linguistics

prehensive context that can augment the training and performance of SLM.

Given the imperfections of knowledge produced by LLM, there is a risk yielding counterproductive outcomes. Consequently, maintaining the accuracy of LLM responses is essential for the effective collaboration between SLM and LLM. Various strategies have been developed to improve it, including series of thoughts (Chain of Thoughts (Wei et al., 2022; Wang et al., 2023), Tree of Thoughts (Yao et al., 2023), Graph of Thoughts (Besta et al., 2024)) and post-processing optimization in interactive scenarios (Reflexion (Shinn et al., 2023), Self-Refine (Madaan et al., 2023), Self-Contrast (Zhang et al., 2024a)). Zhang et al. propose to use policy-level reflection and optimization to iteratively update prompt instructions, empowering the agent to progressively evolve. TRAN (Yang et al., 2023) generates rules by observing mistakes from unsatisfactory generated content to guide LLM for better performance. Inspired by these during-interaction LLM evolving approaches, we explore the possibility of collaboration between two different optimization modality, prompt optimization of LLM and parameter optimization of SLM.

Overall, we propose a co-evolution framework for low resource relation identification with combining adavantages of both SLM and LLM. As is shown in Figure 1, SLM for relation identification and LLM for knowledge enhancement alternately learn from each other during the training process. Our contributions are summarized as follows:

- We capitalize on LLM to generate background knowledge which helps SLM classifier better understand entities for relation identification.
- We devise an auxiliary task with triplet information to boost embedding learning of label tokens for SLM classifier, which is also an essential linkage that associate SLM with LLM.
- We introduce a framework for collaborative evolution of SLM and LLM, empowering LLM to generate more effective context and synchronously SLM to efficiently adapt to low resource task.

2 Related work

Addressing the challenge of relation identification with few shot samples, prevailing strategies can be broadly categorized into two schools of thought.

The first approach enhances traditional language models through prompt tuning. At its core, the method involves inserting textual snippets or templates, into the input and recasting classification task as the MLM problem, which enhances model performance by integrating textual information or label information into the training process. Know-Prompt (Chen et al., 2022b) incorporates knowledge between relation labels into the prompt tuning phase of identification. Specifically, it infuses the potential knowledge embedded within labels into the creation of prompts that include learnable virtual type tokens and answer terms. Subsequently, their representations are jointly optimized under structured constraints. BayesPrompt (Li et al., 2024) leverages a known distribution to approximate the debiased factual distribution on the target domain. It subsequently performs uniform sampling of certain representative features, thereby generating the final prompts. RetrievalRE (Chen et al., 2022a) regards relation identification as an open-book exam, proposing a retrieval-augmented semi-parametric extraction prompt tuning paradigm. It establishes an open-book data repository, where instance representations based on prompts and their corresponding relation labels serve as key-value pairs for retrieval. During inference, the model infers relations by linearly interpolating the base outputs with a non-parametric nearest neighbor distribution retrieved from the data repository.

The second strategy advocates for the synergy between traditional models and large-scale models, enabling them to capitalize on their respective strengths in low-data scenarios. This collaborative effort harnesses the precision and specialized knowledge of SLM alongside the broader contextual understanding and generative capabilities of LLM, resulting in a more robust and adaptable system even when data is scarce. Ma et al. propose a novel approach named LLM-IE where SLM act as a filter and LLM serves as a reranker. By prompting LLM to rerank a small subset of difficult instances identified by SLM, it achieves notable improvements across various information extraction (IE) tasks. GPT-RE (Wan et al., 2023) employs SimCSE (Gao et al., 2021b) to compute sentence similarities, extracts relation representations from a BERT-based (Devlin et al., 2019) fine-tuning method for retrieving demonstration examples, and takes GPT-3 (Brown et al., 2020) with generating the reasoning logic process for each example under the corresponding factual relation label. This collaborative effort between models compensates for the shortcomings within the GPT-3 framework, thereby enhancing its performance.

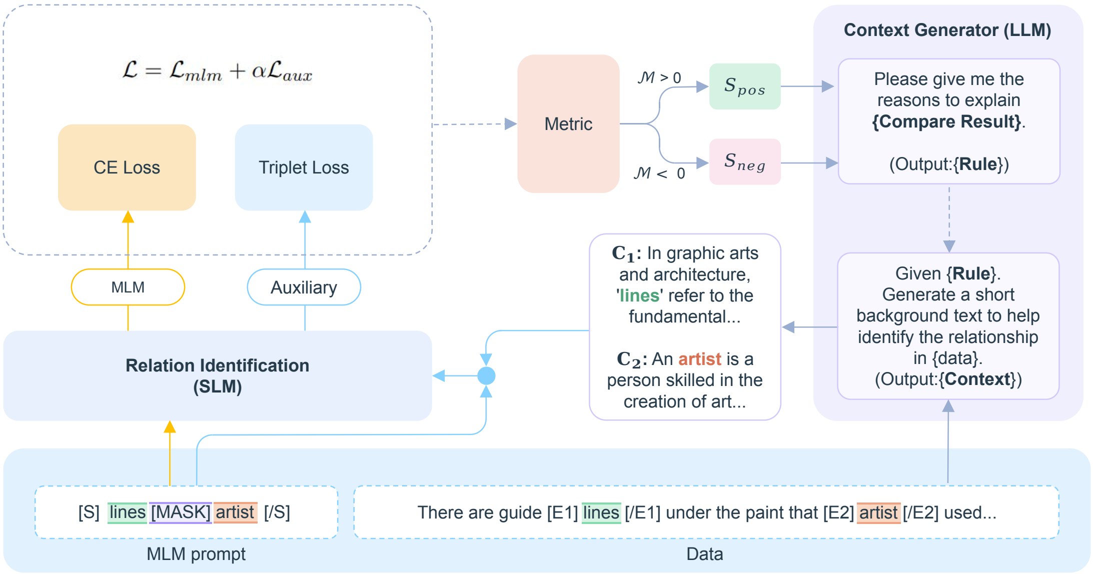

$\mathcal{L} = \mathcal{L}_{mlm} + \alpha\mathcal{L}_{aux}$
CE Loss
Triplet Loss
MLM
Auxiliary
Relation Identification (SLM)
Metric
$\mathcal{M} > 0$
$\mathcal{M} < 0$
S_{pos}
S_{neg}
Context Generator (LLM)
Please give me the reasons to explain {Compare Result}.
(Output:{Rule})
Given {Rule}. Generate a short background text to help identify the relationship in {data}. (Output:{Context})
C1: In graphic arts and architecture, 'lines' refer to the fundamental...
C2: An artist is a person skilled in the creation of art...
[S] lines [MASK] artist [/S]
There are guide [E1] lines [/E1] under the paint that [E2] artist [/E2] used...
MLM prompt
Data

Figure 2: The co-evolution framework for **SLM** & **LLM**. For a training sample, MLM prompt is operating on it for relation identification (the yellow line). In addition, an auxiliary task with triplet information is introduced to enhance the embedding learning (the blue line). LLM provides entity context for auxiliary loss calculation and derives feedback from SLM metric for generation prompt update (the gray line).

3 Proposed method

This part demonstrates our co-evolution framework comprehensively. We perceive SLM as the bedrock for relation identification task and introduce triplet constraint as the training auxiliary (see in Section 3.1). With LLM, we supply entity context to its representation in auxiliary task (Section 3.2). We update the generation prompt on LLM according to current training metrics, which procures a more robust context afterwards (Section 3.3).

3.1 Relation identification with auxiliary task via SLM in low-resource setting

Fortunately, due to the MLM ability of popular bi-directional pre-trained SLM, it is possible to transfer original identification task to the "cloze test" for further prompt tuning in low resource setting. Specifically, we treat the relation label as a special token "[MASK]" surrounded by a pre-defined prompt, and decode the "[MASK]" token to derive the appropriate label expression. Therefore, we model the MLM process for relation identification as:

$$p(y|x, e_x^1, e_x^2) = p([MASK]|\mathcal{P}_{mlm}(x, e_x^1, e_x^2)) \tag{1}$$

where x is the under-test sentence containing entities e_x^1 and e_x^2, y is the relation label which satisfies a triplet (e_x^1, y, e_x^2). $\mathcal{P}_{mlm}(.)$ is the designed prompt. In this paper, we establish the prompt as "[S]e_x^1[MASK]e_x^2[/S]x[/S]" where "[S]" and "[/S]" are start and separator tokens, respectively. For the sentence x, we also mark the positions of both entities:

$$\begin{aligned} x = x(&t_1, ..., [SUB_s], e_x^1, [SUB_e]..., \\ &[OBJ_s], e_x^2, [OBJ_e], ..., t_n) \end{aligned} \tag{2}$$

where special tokens like $[SUB_s]$ are symbols encircling entities for location indication.

For decoding facilitation, we expand the original vocabulary space V to $V^{'}$ which covers label's expression as a single condensed token, i.e., $[CLASS_i]$ for label *i*. Hence, the decoding loss function is instated to:

$$\mathcal{L}_{mlm} = -\frac{\sum_{x \in X} ylogp(M{=}C_y|\mathcal{P}_{mlm}(x, e_x^1, e_x^2))}{|\mathcal{D}|}$$
$$\textbf{s.t. } \mathcal{D} = \{(e_x^1, e_x^2, x, y)|x \in X\} \tag{3}$$

where $M{=}C_y$ is the simplification of $[MASK]{=}[CLASS_y]$. $\mathcal{D}$ is the training dataset.

With substituting for learning the embedding of label token $[CLASS_i]$ from scratch, we make average value of embeddings from label expression tokens for initialization. Moreover, an auxiliary task inspired by TransE (Bordes et al., 2013) is proposed to aid for the learning process. Concretely, for a triplet item (e_x^1, y, e_x^2), we aim to minimize:

$$\begin{aligned} \mathcal{L}_{aux} &= -log\sigma(d_m - d(e_x^1, y, e_x^2)) \\ \textbf{s.t. } d(e_x^1, y, e_x^2) &= |R_{e_x^1} + R_y - R_{e_x^2}|_2 \end{aligned} \tag{4}$$

where σ is the *sigmoid* function, d_m is the difference margin. $R_{(.)}$ means the representation of which R_y is derived from *output embedding* of MLM over its corresponding label token, i.e., $Emb_{out}([CLASS_y])$.

Therefore, the complete loss function for relation identification task will serve as:

$$\mathcal{L} = \mathcal{L}_{mlm} + \alpha\mathcal{L}_{aux} \tag{5}$$

3.2 LLM as a knowledge base to enrich context for auxiliary task

As for representations of each entity in Equation 4, instead of encoding directly on their detailed tokens, we make recourse to the correspondingly global context from external knowledge. LLM is proficient in text generation and can furnish prodigal context on any key words touched upon. Hence, for an entity e_x mentioned in sentence x, we devise a prompt ranging over both of them to compel LLM to generate a piece of constructive context, which distinguishes the entity from view of the relation type in x:

$$\begin{aligned} \mathcal{C}_{e_x} &= (t_1, ..., t_n) \\ \textbf{s.t. } t_i &= argmax(p(t|\mathcal{P}_{gen}(e_x), t_{1:i-1})) \end{aligned} \tag{6}$$

where $\mathcal{P}_{gen}(.)$ is the aforementioned prompt, t_i is the token to be generated by LLM. Furthermore, we indicate representation of the entity as the mean pooling on encodings of its context tokens:

$$R_{e_x} = \frac{1}{|\mathcal{C}_{e_x}|} \sum_{t \in \mathcal{C}_{e_x}} Encoder(t) \tag{7}$$

where $Encoder(.)$ is the SLM encoder in Section 3.1.

3.3 LLM as a metric critic for co-evolution with SLM

Subjected to quality of the generation prompt, LLM suffers from the limited effectiveness (Wang and Li, 2023), which circumscribes the context applicability in Section 3.2. To alleviate the problem, we propose a reflection method that the LLM realigns its generation prompt vitally in conformity with the instant training state on SLM of Section 3.1.

More specifically, following Equation 6, LLM affords entities the initial context to usher in the training phase on SLM. With several epochs, we gather experiences from current training metrics, e.g., the precision:

$$\begin{aligned} \mathcal{S}_{pos} &= \{(\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x) | \mathcal{M}(SLM(x)) > 0\} \\ \mathcal{S}_{neg} &= \{(\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x) | \mathcal{M}(SLM(x)) < 0\} \end{aligned} \tag{8}$$

Algorithm 1: Three-Stage Train Process

```
Input  : $\mathcal{D}$, SLM, LLM, $\mathcal{P}_{mlm}$, $\mathcal{P}_{init}$, $\mathcal{P}_{pos}$, $\mathcal{P}_{neg}$, $\mathcal{P}_{rl}$, $k$
Output : $\mathcal{C}_{e_x}$, SLM, Rule
DEF Func():
    $\mathcal{P}_{gen} \leftarrow Rule \circ \mathcal{P}_{init}$
    $\mathcal{C}_{e_x^i} \leftarrow LLM(\mathcal{P}_{gen}(e_x^i, x)), i \in \{1, 2\}$
    $x_o, x_l \leftarrow SLM(\mathcal{P}_{mlm}(x, \mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}))$
    $SLM \leftarrow SLM - \alpha\nabla x_l$
    $\mathcal{H} \leftarrow \mathcal{H} + (x_o[y])$
DEF Rule():
    $ep \leftarrow sample(ep, k), ep \in \{\mathcal{S}_{pos}, \mathcal{S}_{neg}\}$
    $RSN \leftarrow LLM(\mathcal{P}_{pos}(\mathcal{S}_{pos}), \mathcal{P}_{neg}(\mathcal{S}_{neg}))$
    $Rule \leftarrow LLM(\mathcal{P}_{rl}(RSN))$
Stage One ($\mathcal{S}_{pos}$, $\mathcal{S}_{neg}$, $\mathcal{H}$, Rule)
for $e_x^1$, $e_x^2$, $x$, $y$ in $\mathcal{D}$ do
    Func()
    if $argmax(x_o) = y$ then
        $\mathcal{S}_{pos} \leftarrow \mathcal{S}_{pos} + (\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x)$
    else
        $\mathcal{S}_{neg} \leftarrow \mathcal{S}_{neg} + (\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x)$
    end if
end for
Rule()
Stage Two ($\mathcal{S}_{pos}$, $\mathcal{S}_{neg}$)
for $di$, $e_x^1$, $e_x^2$, $x$, $y$ in $enum(\mathcal{D})$ do
    Func()
    if $\mathcal{H}[-1][di] - \mathcal{H}[-2][di] > 0$ then
        $\mathcal{S}_{pos} \leftarrow \mathcal{S}_{pos} + (\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x)$
    else
        $\mathcal{S}_{neg} \leftarrow \mathcal{S}_{neg} + (\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x)$
    end if
end for
Rule()
Stage Three
for $e_x^1$, $e_x^2$, $x$, $y$ in $\mathcal{D}$ do
    Func()
end for
```

Type	Content	Input
$\mathcal{P}_{init}$	Generate a short background text about entity "{}" in the sentence "{}" to assist the relation identification task. Directly give the background text within fifty words. Do not generate any unrelated contents.	e_x, x
$\mathcal{P}_{pos}$	According to the experiment, the background texts "{}" and "{}" improve relation identification between entities "{}" and "{}" in the sentence "{}". Please give me the reasons to explain the improvement.	$\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x$
$\mathcal{P}_{neg}$	According to the experiment, the background texts "{}" and "{}" are unfavorable for relation identification between entities "{}" and "{}" in the sentence "{}". Please give me the reasons.	$\mathcal{C}_{e_x^1}, \mathcal{C}_{e_x^2}, e_x^1, e_x^2, x$
$\mathcal{P}_{rl}$	Please rewrite these reasons into rules to generate background texts in aid of relation identification, using the format of "if..., then...". Give it in sections. Each is an independent rule. Reasons:{}	*Reason*
$\mathcal{P}_{gen}$	Given the following rules:{}.{}	*Rule*, $\mathcal{P}_{init}$

Table 1: A reflection prompt machinery for LLM generation.

where $\mathcal{C}_{e_x^i}$ is the context for entity e_x^i, $\mathit{SLM}(x)$ is output of the relation identification model in Section 3.1. $\mathcal{M}(.)$ is the metric on which we conduct an assessment and varies in training stages. Exercising the experiences (after sampling), LLM summarizes specific **Rules** with reflection mechanism for metric change explanation (see Algorithm 1):

$$\begin{aligned} \mathit{Reason} &= \mathit{LLM}(\mathcal{P}_{pos}(\mathcal{S}_{pos}), \mathcal{P}_{neg}(\mathcal{S}_{neg})) \\ \mathit{Rule} &= \mathit{LLM}(\mathcal{P}_{rl}(\mathit{Reason})) \\ \mathcal{P}_{gen} &= \mathit{Rule} \circ \mathcal{P}_{init} \end{aligned} \tag{9}$$

where $\mathcal{P}_{pos}$ and $\mathcal{P}_{neg}$ are feedback induction prompts for positive and negative experiences, respectively. $\mathcal{P}_{rl}$ wraps occasions for generating a more abstract rule. $\mathcal{P}_{init}$ is the initial context generation prompt without any rules. $\mathcal{P}_{gen}$ is a compound prompt with rules (referring to Table 1).

Being grounded in dissimilitude of metric expression, we separate the training process as three consecutive stages:

Stage One ($\mathit{Rule} = \mathit{None}, \mathcal{S}_{pos} = \mathit{None}, \mathcal{S}_{neg} = \mathit{None}$) We initial experiences on both $\mathcal{S}_{pos}$ and $\mathcal{S}_{neg}$, and provide a variable $\mathcal{H}$ to chronicle predict probabilities on the true labels from relation identification model:

$$\mathcal{H}[-1] = SLM(x)[y] \tag{10}$$

The metric we select is to judge if the prediction result is proper:

$$\mathcal{M} = \begin{cases} 1 & argmax(\mathit{SLM}(x)) = y \\ -1 & other \end{cases} \tag{11}$$

Following Equations 8 & 9, we derive Rule_{s1} as the reflection on experiences.

Stage Two ($\mathit{Rule} = \mathit{Rule}_{s1}, \mathcal{S}_{pos} = \mathit{None}, \mathcal{S}_{neg} = \mathit{None}$) In this stage, we employ $\mathcal{H}$ to establish the metric as:

$$\mathcal{M} = \mathcal{H}[-1][di] - \mathcal{H}[-2][di] \tag{12}$$

where di means index of the training item x. -1 and -2 indicate the predict probabilities of x on **Stage Two** and **Stage One** which evaluates improvement of Rule_{s1}. Similarly, we will attain an updated rule as Rule_{s2} for reflection on context improvement after the stage cessation.

Stage Three (Rule=$\mathit{Rule}_{s2}, \mathcal{S}_{pos}$=$\mathit{None}, \mathcal{S}_{neg}$=$\mathit{None}$) Assembled by the latest rule, LLM supplies entities with credible context which guides the training process of the SLM in a fine-grained manner.

4 Experiments

In this section, we conduct our experiments on several popular relation identification datasets, being in comparison with the SOTA baselines (Section 4.1). We make an ablation study on the coevolution structure to verify its effectiveness (Section 4.2). Moreover, a case study is demonstrated to embody the dynamic improvement of both context and the rule (Section 4.3). Similar to the SLM based methods, inferring part of our model covers a smaller size of parameters with the number of *355,435,632*, which is much more efficient than the LLMs. All the experiments are implemented

k	Methods	SemEval	TACRED	TACRED-Revisit	Re-TACRED	Average
1	KnowPrompt	0.164	0.050	0.048	0.034	0.074
	RetrievalRE	0.213	0.031	0.035	0.082	0.090
	BayesPrompt	0.345	0.204	0.269	0.023	0.210
	LLM-IE	0.219	0.070	0.070	0.046	0.101
	LLM-IE †	0.205	0.066	0.061	0.049	0.095
	our model	0.351	0.215	0.245	0.357	**0.292**
	our model†	0.213	0.208	0.244	0.328	0.248
2	KnowPrompt	0.164	0.245	0.290	0.441	0.285
	RetrievalRE	0.394	0.195	0.219	0.090	0.224
	BayesPrompt	0.375	0.259	0.342	0.430	0.351
	LLM-IE	0.219	0.277	0.304	0.444	0.311
	LLM-IE †	0.205	0.266	0.300	0.416	0.297
	our model	0.449	0.272	0.316	0.450	**0.372**
	our model†	0.399	0.237	0.281	0.394	0.328
5	KnowPrompt	0.565	0.361	0.360	0.549	0.459
	RetrievalRE	0.612	0.343	0.315	0.274	0.386
	BayesPrompt	0.725	0.333	0.347	0.538	**0.486**
	LLM-IE	0.529	0.392	0.383	0.553	0.464
	LLM-IE †	0.417	0.385	0.368	0.548	0.429
	our model	0.620	0.385	0.343	0.556	0.476
	our model†	0.466	0.321	0.323	0.507	0.404
✘	GPT-RE	0.659	0.275	0.314	0.485	0.433
	GPT-RE †	0.604	0.263	0.251	0.417	0.384

Table 2: F1 score for relation identification in low-resource settings. k means examples' number per label. Meta-Llama3-8b-instruct is employed for LLM conditions (except for †). † means Llama2-13b-chat is utilized for LLM generation.

on a couple of GPUs (*Tesla V100S-PCIE-32GB*), with one for *SLM training* and another for *LLM generation*.

We set the learning rate of the SLM part as *5e-5*, and the training batch size as *8*. Coefficients of d_m and α in Equations 4 & 5 are 0.2 and 0.02 separately. For a stable training process, we execute three epochs for each training stage. Correspondingly, the variable $\mathcal{H}$ in Equations 10 and 12 updates its value every three epochs.

Evaluation Datasets: SemEval (SemEval 2010 Task 8 (Hendrickx et al., 2010)), TACRED series [1] (TACRED (Zhang et al., 2017), TACRED-Revisit (Alt et al., 2020) and Re-TACRED (Stoica et al., 2021)). These datasets are public for research utilization.

Baselines: SLM series (KnowPrompt, RetrievalRE, BayesPrompt), LLM series (LLM-IE, GPT-RE).

4.1 Evaluation results

We make low-resource settings on the four datasets aforementioned with randomly sampling k ($= 1/2/5$) training items for each label. Distinguished from the general few-shot setting, which usually covers 8/16/32 samples per label, we refer to extremely low resource setting for several reasons. To start with, the instance number of a relation type would be limited in the dataset, e.g., just 6 instances for *per:country_of_death* relation type, and 23 instances for *org:dissolved* relation. Furthermore, in actual scenario, especially when LLM gets popular, people desire a reliable model but with less labeled data or to devote as less as possible for labeling data, i.e., just labeling one instance for a label type.

In this paper, owing to generation costs of LLM related baselines, we also subsample 1000+ items for testing with the label ratios (original & subsample datas) being in concert strictly. Yet we make a full-test-data comparison with SLM based methods in Figure 3.

[1] https://catalog.ldc.upenn.edu/LDC2018T24

We employ Roberta [2] (Liu et al., 2019) large model as the backbone for relation identification classifier (also for baselines with SLM). The LLM utilized is Meta-Llama3-8b-instruct [3] which covers a public use for language generation (we make comparison on Llama2-13b-chat [4] (Touvron et al., 2023) as well). Referring to the work (Yang et al., 2023), a reflection prompt machinery mentioned in Equation 9 for LLM is designed in Table 1.

As for the baseline LLM-IE, we plant KnowPrompt as the SLM filter and alter its output logic as "with recourse to SLM when out-of-label appears during LLM generation". Another LLM series baseline GPT-RE, as it doesn't operate on the low-resource setting, we afford 15 demonstrations over each dataset for in-context learning.

The main results are shown in Table 2 from which we make observations:

(1) The proposed model surpasses most of the baselines among four datasets. In one-shot setting, our model exhibits great advantages with 8.2% superiority than the best baseline (19.1% than the suboptimal baseline). The numbers are 2.1% and 6.1% correspondingly in the two-shot setting. It manifests potential in extremely low-resources-setting applications. When $k = 5$, our method demonstrates a slight inferiority compared to *BayesPrompt* (-1%). Nevertheless, our method still dominates most datasets where results for **TACRED** and **Re-TACRED** are superior to those of *BayesPrompt* with 5.2% and 1.8%.

(2) In contrast with *GPT-RE*, our model with five-shot setting performs better with 4.3% (with Llama3-8b-instruct as the co-evolution LLM) and 2.0% (Llam2-13b-chat), which signifies our model with a pennyworth of training samples can be comparable to LLM generation methods. Moreover, it is noteworthy that the inference style of our model (inferring merely on the SLM) owns natural advantages compared to LLM based models in the aspect of time-consuming.

(3) Evaluation results varies in different LLM based generators. Generally, LLM with a more powerful ability will take precedence, e.g., average 5.3% ascendancy from Llama2-13b-chat to Llama3-8b-instruct in our model setting. The phenomenon remains pervasive in other methods of *LLM-IE* and *GPT-RE*.

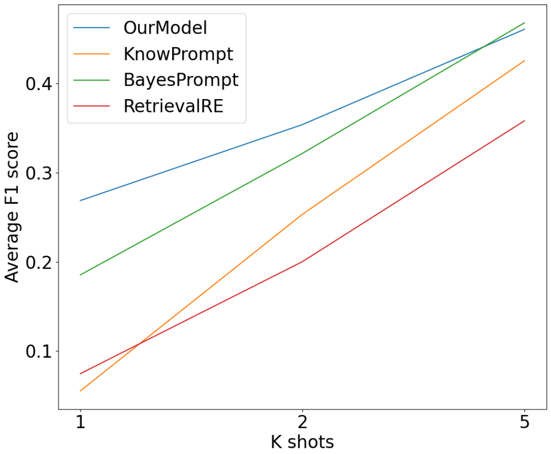


Figure 3: Comparison with SLM based methods in full-data test.

For full-data test results, Figure 3 illustrates the predominance of our model in extremely low resource setting (i.e., k=1 & 2).

4.2 Ablation study

For the sake of validity check on our co-evolution framework, we conduct ablation study from three aspects progressively.

Validation on auxiliary task: We remove the auxiliary part in Section 3.1 and retain only the $\mathcal{L}_{mlm}$ as loss function for SLM training. In this way, i.e., **-AT**, the MLM ability is purely applied for relation identification task. From Table 3, the auxiliary task plays a pivotal role as it affords an external structure information (the triplet loss) which facilitates learning of the relation's embedding (i.e., $Emb_{out}([CLASS_y])$) in low resource settings. Without the auxiliary part, identification result will descend by average 7.1%.

Validation on entity context: We make a surrogate **-CL** for entity context generated by LLM in Section 3.2, which takes entity tokens directly for representation calculation (Equation 4). In comparison with the context based representation, entity tokens provide less explicit trails to establish relational connection, hence limit the performance with average 6.9% drop.

Validation on generation prompt update: With freezing Equations 8 & 9 as **-PU**, we evacuate the rule and generate entity context exclusively by

[2] https://huggingface.co/FacebookAI/roberta-large

[3] https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

[4] https://huggingface.co/meta-llama/Llama-2-13b-chat

k	Methods	SemEval	TACRED	TACRED-Revisit	Re-TACRED	Average
1	**our model**	0.351	0.215	0.245	0.357	**0.292**
	-AT	0.241	0.175	0.200	0.286	0.225$_{(-0.067)}$
	-CL	0.241	0.176	0.200	0.287	0.226$_{(-0.066)}$
	-PU	0.289	0.201	0.238	0.345	0.268$_{(-0.024)}$
2	**our model**	0.449	0.272	0.316	0.450	**0.372**
	-AT	0.352	0.244	0.266	0.361	0.306$_{(-0.066)}$
	-CL	0.337	0.244	0.264	0.360	0.301$_{(-0.071)}$
	-PU	0.417	0.229	0.285	0.399	0.333$_{(-0.039)}$
5	**our model**	0.620	0.385	0.343	0.556	**0.476**
	-AT	0.443	0.318	0.320	0.505	0.397$_{(-0.079)}$
	-CL	0.459	0.323	0.328	0.519	0.407$_{(-0.069)}$
	-PU	0.602	0.314	0.314	0.487	0.429$_{(-0.047)}$

Table 3: Ablation study on each dataset. **-AT** means without auxiliary task. **-CL** applies entity tokens instead of entity context generated by LLM for auxiliary task. **-PU** is training without generation prompt update.

$\mathcal{P}_{init}$. The context is derived only once for an entire training phase. Being contradistinction to our *three-stage training process*, prompt-update free method is confronted by the finitude on the context quality constantly. Yet the performance degenerates by 3.7%, it has a minimal impact on relation identification among all the three ablation settings.

4.3 Case study

We implement a case study to track the context's crossover from rule-free to rule-based, which is anticipated to manifest context generation improvement via LLM reflection. As is displayed in the Reflection Process .1, context generated by initial prompt $\mathcal{P}_{init}$ for the entity "artist" is the stuff to describe the expression in a general way, which ensures no linkage to its couple entity thus limits the relation identification ability.

Reflection Process

x**:** There are guide lines under the paint that the artist used to create the pedestal in perfect perspective.
e_x**:** (lines, artist)
*rel***:** Instrument-Agency

Stage One
$\mathcal{C}_{e_x^1}$**:** "Guidelines" refer to invisible lines or marks on a surface used as a reference by an artist or craftsman to ensure accurate...
$\mathcal{C}_{e_x^2}$**:** The artist is a skilled craftsman who uses perspective to create a realistic representation of a three-dimensional scene...

Rule 1: If there is no explicit connection between entities, then provide a connection to link them.

Stage Two
$\mathcal{C}_{e_x^1}$**:** In graphic arts and architecture, 'lines' refer to the fundamental elements used to convey shape, form, and perspective. Guide lines, in particular, are helpful tools artists and designers use to...
$\mathcal{C}_{e_x^2}$**:** An artist is a person skilled in the creation of art, using various mediums such as painting, drawing, sculpture, or photography. They often use techniques and tools...

Rule 2: If the background texts overemphasis on technical aspects, then it might lead the model not to focus on their relation.

Stage Three
$\mathcal{C}_{e_x^1}$**:** Guide lines are subtle marks made on a surface to help artists and craftsmen achieve precise control when creating...
$\mathcal{C}_{e_x^2}$**:** The term 'artist' refers to a person who creates original works of art, such as paintings, sculptures, or drawings, using various techniques and tools...

With the metric critic in **Stage One**, contexts for both the entities demonstrate explicit association with emphasize the couple entity in the corresponding context. Nevertheless, "fundamental elements" makes overemphasis on measurement and perspective, not artistic intent, which may mislead the model about the relation type. After a further reflection in **Stage Two**, a more robust prompt $\mathcal{P}_{gen}$ is derived which attaches "lines" with a concise but focused depiction relevant to "artist", hence a perspicuous trail creeps in for relation identification.

5 Conclusions

We propose a co-evolution framework operated on **SLM** & **LLM** for relation identification task in the extremely low resource scenario. In the model, SLM is the classifier which treats the task as prompt tuning, and LLM serves as a knowledge base to provide entity contexts. We introduce an auxiliary task with triplet information as a bond between them. Moreover, a three-stage training process is established which conveys training states of SLM to LLM for its generation reflection, hence requited contexts with improvement produced by LLM will further foster the training of SLM. We conduct experiments on four datasets with comparison to both SLM and LLM related methods which displays advances of our model.

Limitations

In this paper, we explore the collaboration between optimizations of SLM and LLM and call on it for relation identification in low resource scenarios. We establish the co-evolution strategy with a three-stage training process and conduct experiments on several datasets to verify advancement of our method. Moreover, it confirms that LLM can make guidance for SLM learning with continuous feedback.

However, current strategy rests on the context that LLM generates for each training items of SLM. Hence the generation quality is still a demerit to deal with (although we update the generation prompt in accordance with the instant SLM training metrics) and the time-consuming for context generation will be unpalatable if the training data size gets larger. Therefore, we will study a more efficient co-evolution strategy in the future.

Acknowledgements

This work is supported by Beijing Natural Science Foundation (L222006), National Key R&D Program of China (2021ZD0140408) and China Mobile Holistic Artificial Intelligence Major Project Funding (R22105ZS, R22105ZSC01).

References

Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. 2020. TACRED revisited: A thorough evaluation of the TACRED relation extraction task. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1558–1569, Online. Association for Computational Linguistics.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michał Podstawski, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2024. Graph of Thoughts: Solving Elaborate Problems with Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17682–17690.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 2787–2795, Red Hook, NY, USA. Curran Associates Inc.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Xiang Chen, Lei Li, Ningyu Zhang, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022a. Relation extraction as open-book examination: Retrieval-enhanced prompt tuning. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '22, page 2443–2448, New York, NY, USA. Association for Computing Machinery.

Xiang Chen, Ningyu Zhang, Xin Xie, Shumin Deng, Yunzhi Yao, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022b. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 2778–2788, New York, NY, USA. Association for Computing Machinery.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021a. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021b. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 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*, pages 33–38, Uppsala, Sweden. Association for Computational Linguistics.

Jiangmeng Li, Fei Song, Yifan Jin, Wenwen Qiang, Changwen Zheng, Fuchun Sun, and Hui Xiong. 2024. Bayesprompt: Prompting large-scale pre-trained language models on few-shot inference via debiased domain abstraction. In *The Twelfth International Conference on Learning Representations*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10572–10601, Singapore. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46594. Curran Associates, Inc.

Chaoxu Pang, Yixuan Cao, Qiang Ding, and Ping Luo. 2023. Guideline learning for in-context information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15372–15389, Singapore. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pages 8634–8652. Curran Associates, Inc.

George Stoica, Emmanouil Antonios Platanios, and Barnabás Póczos. 2021. Re-tacred: Addressing shortcomings of the TACRED dataset. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13843–13850. AAAI Press.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Somin Wadhwa, Silvio Amir, and Byron Wallace. 2023. Revisiting relation extraction in the era of large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15566–15589, Toronto, Canada. Association for Computational Linguistics.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. GPT-RE: In-context learning for relation extraction using large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3534–3547, Singapore. Association for Computational Linguistics.

Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. 2023.

Towards understanding chain-of-thought prompting: An empirical study of what matters. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2717–2739, Toronto, Canada. Association for Computational Linguistics.

Danqing Wang and Lei Li. 2023. Learning from mistakes via cooperative study assistant for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10667–10685, Singapore. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Zeyuan Yang, Peng Li, and Yang Liu. 2023. Failures pave the way: Enhancing large language models through tuning-free rule accumulation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1751–1777, Singapore. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc.

Wenqi Zhang, Yongliang Shen, Linjuan Wu, Qiuying Peng, Jun Wang, Yue Ting Zhuang, and Weiming Lu. 2024a. Self-contrast: Better reflection through inconsistent solving perspectives. *ArXiv*, abs/2401.02009.

Wenqi Zhang, Ke Tang, Hai Wu, Mengna Wang, Yongliang Shen, Guiyang Hou, Zeqi Tan, Peng Li, Yueting Zhuang, and Weiming Lu. 2024b. Agent-pro: Learning to evolve via policy-level reflection and optimization. *CoRR*, abs/2402.17574.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark. Association for Computational Linguistics.