KICE: A Knowledge Consolidation and Expansion Framework for Relation Extraction

Yilin Lu, Xiaoqiang Wang, Haofeng Yang, Siliang Tang [*]

School of Computer Science, Zhejiang University
{22121281, xq.wang, 3190105301, siliang}@zju.edu.cn

Abstract

Machine Learning is often challenged by insufficient labeled data. Previous methods employing implicit commonsense knowledge of pre-trained language models (PLMs) or pattern-based symbolic knowledge have achieved great success in mitigating manual annotation efforts. In this paper, we focus on the collaboration among different knowledge sources and present **KICE**, a **K**nowledge-evolving framework by **I**terative **C**onsolidation and **E**xpansion with the guidance of PLMs and rule-based patterns. Specifically, starting with limited labeled data as seeds, **KICE** first builds a Rule Generator by prompt-tuning to stimulate the rich knowledge distributed in PLMs, generate seed rules, and initialize the rules set. Afterwards, based on the rule-labeled data, the task model is trained in a self-training pipeline where the knowledge in rules set is consolidated with self-learned high-confidence rules. Finally, for the low-confidence rules, **KICE** solicits human-enlightened understanding and expands the knowledge coverage for better task model training. Our framework is verified on relation extraction (RE) task, and the experiments on TACRED show that the model performance (F_1) grows from 33.24% to 79.84% with the enrichment of knowledge, outperforming all the baselines including other knowledgeable methods.

Introduction

Relying on the large-scale human-annotated training data, machine learning models presented in recent years have been undergoing unprecedentedly rapid development (Pennington, Socher, and Manning 2014; Yang et al. 2019). However, since the scale of labeled data is limited by the time-consuming and expensive human labor, in most areas the prior knowledge provided by crowdsourcing datasets is quite insufficient. This issue is more serious in relation extraction (RE) task because the large-scaled newly emerging relations are difficult to be handled by the model trained on a limited number of pre-defined relations in an old dataset. To be more specific, no more than 200 relation types could be covered by current RE datasets with rich labeled data (Zhang et al. 2017; Han et al. 2018), while there are much more relations in the real scenario. For example, one of the largest knowledge bases Wikidata (Vrandečić and Krötzsch 2014) currently contains nearly 6,000 relations.

To reduce the manual efforts, recent works try to leverage diverse sources of knowledge to steer an efficient model learning. Some methods provide weak labels for unlabeled corpus by incorporating external knowledge graphs (KGs) (Lin et al. 2016) or summarizing knowledge from the model's high-confidence predictions (Qu et al. 2018; Zhou et al. 2020). Others employ active learning (Margatina et al. 2021; Chen and Qian 2022) to ask the knowledge from extra human annotation for confusing (or uncertain) data. However, on the one hand, they suffer from the coverage of available knowledge because the scale of KGs is limited and only focusing on the self-inferred knowledge could stick the task model in a "comfort zone", rendering the weakness in new knowledge discovery (Gao et al. 2020). On the other hand, the insufficient initial labeled data may hinder the active learning process since the under fitted model fails to estimate the uncertainty and representativeness appropriately in the early stage (*i.e.*cold-start problem).

We argue that integrating the wide variety of knowledge can help alleviate the dependency on manual annotation, and present **KICE**, a knowledge consolidation and expansion framework. In this paper, **KICE** is instantiated for the relation extraction task and generally applicable to other tasks. Specifically, we engage the rule-based patterns as transferred and explainable forms among different knowledge sources, which are also beneficial to match large-scaled unlabeled instances and provide weak labels for those data. Given seed instances as initial training data, our framework continuously develops various knowledge to improve task model learning by the following three steps:

1) **Knowledge Stimulation**: To better build a flexible pattern extractor from a limited number of seed instances, we first stimulate rich contextualized knowledge distributed in pre-trained language models (PLMs) by prompt-tuning. More specifically, a *Rule Generator* is formulated by designing various templates to build prompts from each instance, which are fed to PLM and take the output predicted words as different contextual patterns. Rules are obtained by taking patterns as conditions and the corresponding instance's label as a result. Then the rules set is initialized by these generated patterns from seeds. 2) **Knowledge Consolidation**: Based on the rule-labeled data, we incorporate the task model into

[*] Siliang Tang is the corresponding author.

a self-training pipeline and devise a confidence-based *Self-Reviewing Module* to measure the mastery of knowledge and explore self-learned new patterns. It applies the task model to provide pseudo-labels on unlabeled data and summarizes knowledge from high-confidence ones by our *Rule Generator* to build new rules for rules set enrichment. 3) **Knowledge Expansion**: For those low-confidence data, we build a *Rule-induced Breakthrough Learning Module* to enlighten the confusion of the task model by sampling a few data with the most model uncertainty for human annotation. Different from traditional active learning, those annotations don't add to the training set, but are fed to *Rule Generator* for new rules instead. We expect that these rules can generalize to more instances than actual additional manual annotations and maximize the effect of the human-enlightened knowledge.

Knowledge in the rules set could grow more and more complete by conducting step 2) and 3) alternately and in each step after the rules set updating, a few rule-labeled data are utilized to update the training set for better model learning. The knowledge-developing procedure in our framework brings three major benefits: 1) The combination of various knowledge sources significantly reduces human annotation efforts: our framework only needs small-scaled human-labeled data as seeds. Furthermore, instead of asking human for rules annotation, we only solicit annotation on instances and utilize PLM to replace human for flexible rules generation, which is simpler and less time-consuming; 2) Compared with the work simply exploring knowledge by task model itself, *Self-Reviewing Module* could let model steps out of its "comfort zone" by discovering new knowledge contained in most confusing instances; 3) *Rule-induced Breakthrough Learning Module* not only alleviate the cold start problem in active learning but also enhance the effect of solicited human annotation by summarizing them to rules to cluster more instances with similar patterns.

Our framework achieves great performance on TACRED (Zhang et al. 2017) with 5% training data and the experiments results illustrate the significant improvement of the model's performance with knowledge development, which outperforms all the baselines. We further conduct ablation studies to verify our framework collaborates different knowledge effectively.

Related Work

Self-Training This kind of methods explore rules iteratively by the RE model or some modules jointly trained with it and enrich the training set with weakly labeled data. REPEL (Qu et al. 2018) proposes a facts evaluated function utilized to pick rules providing high-score facts and meanwhile improved by the rule-labeled facts. DualRE (Lin et al. 2019) jointly trains a retrieval module with the RE model to retrieve sentences for a given relation. Since at the early iterations the RE model trained on few data may select noise rules, Snowball (Gao et al. 2020) pre-trains the model on relations with rich training data and transfer the knowledge to explore rules for those relations with few seeds. But as reported in Snowball, the model overfits existing rules and fails to discover rules with new knowledge, while our framework asks human annotation for the model's most confused instances and summarizes new knowledge from them.

Learning with Pre-trained Language Model Prompt-tuning has achieved a great performance in relation extraction. PTR (Han et al. 2021) builds prompts consisting of several sub-prompts and infers relation by considering predicted masked words on different sub-prompts. To alleviate labor on constructing label words, Knowprompt (Chen et al. 2022) proposes learnable virtual answer words. To reduce the requirement of human-labeled training data, COSINE (Yu et al. 2020) designs a denoising mechanism to fine-tune pre-trained model on weakly labeled data. However, the rules set in COSINE is fix, while our framework builds a growing rules set with self-consolidated knowledge and human-enlightened knowledge.

Interactive Learning Interactive learning involves human knowledge injection in the learning process and has proposed different strategies to sample the most valuable query data for model learning (Margatina et al. 2021; Chen and Qian 2022). Since the knowledge contained in the annotated data is insufficient, several work ask annotation for rules to match unlabeled data (Hsieh, Zhang, and Ratner 2022; Boecking et al. 2020). However, the rules annotation is more difficult than data annotation, while our framework simply solicits annotation on data and utilizes PLM to generate rules from them to enhance the effect of human knowledge.

Methodology

In this section, we introduce our `KICE` framework in detail, which starts from small-scale labeled data (donated as seeds) and achieves a continuous improvement by both knowledge consolidation and expansion. As shown in Figure 1, our framework consists of three steps: 1) **Knowledge Stimulation**: Given small-scaled seeds, we first employ it as our initial training set for the RE model. Then we build a *Rule Generator* on these seeds by prompt-tuning to generate initial rules, which harnesses the pattern of instances and the contextualized knowledge of PLM. 2) **Knowledge Consolidation**: A *Self-Reviewing Module* is built by applying the RE model to the unlabeled dataset for generating pseudo labels and summarizing new rules from high-confidence ones. 3) **Knowledge Expansion**: For those low-confidence instances, we devise a *Rule-induced Breakthrough Learning Module* by sampling the most confusing unlabeled data for extra human annotations, then obtain new rules from them. The 2) and 3) steps are conducted iteratively. In each step, the training set is built with the weakly labeled data provided by the enriched rules set to learn a new RE model.

Rule Generator

Considering that the instances with similar contextual patterns are more likely to express the same relation, our *Rule Generator* summarizes a rule by extracting contextualized patterns of the corresponding single instance with the guidance of PLM. This kind of rule-based patterns can be utilized to provide weak labels for unlabeled data in a soft-matching manner, as well as serving as a transferred and explainable form among different knowledge sources.

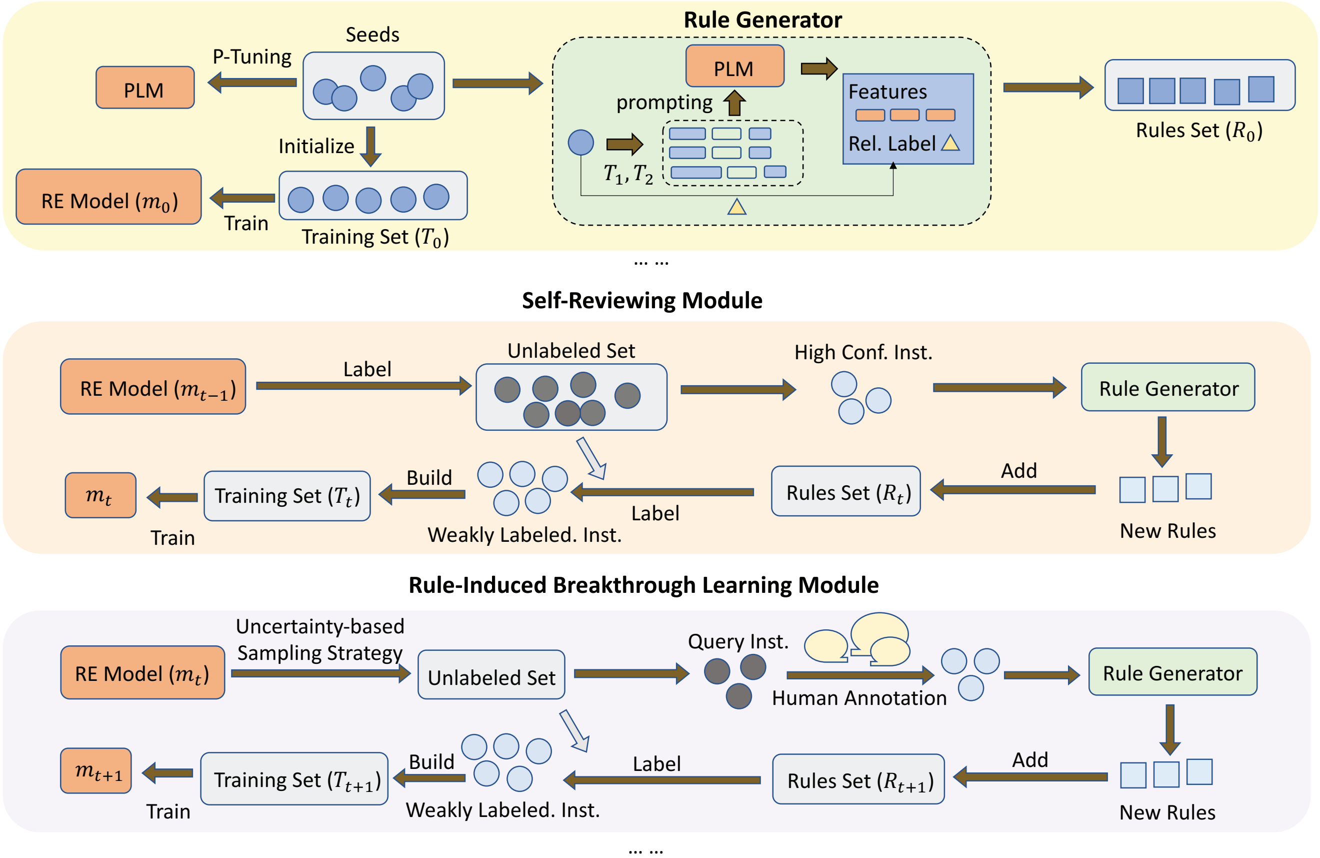

Figure 1: Illustration of our knowledge-evolving framework by iterative consolidation and expansion (**KICE**). Given a few seed instances, a PLM-based patterns summarizer is prompt-tuned to construct Rule Generator, which initializes rules set from seeds. Then two modules are conducted alternatively. In Self-Reviewing Module, new rules are summarized from the pseudo-labeled data with high confidence. And in Rule-induced Breakthrough Learning Module, unlabeled instances with most model's uncertainty are sampled for human annotation, which is also applied for new rules summarization.

Rule Definition Each rule p is generated from an instance and consists of two entities patterns $\{f^p_{e_i}\}^2_{i=1}$, a relation pattern f^p_c, a label l, a threshold TH, and a similarity function $g(\cdot,\cdot)$. Given an unlabeled sample u, u is matched by p if the overall similarity of entities and relation patterns between u and p exceeds the corresponding threshold. Formally,

$$\mathbb{1}(u \text{ matched } p) = \mathbb{1}\big(s(p,u) \geq TH\big) \quad (1)$$

$$s(p,u) = \sum_{i=1}^{2} g(f^p_{e_i}, f^u_{e_i}) + g(f^p_c, f^u_c) \quad (2)$$

In following sections we introduce 1) Pattern Summarization, 2) Similarity Measurement and 3) Weak label generation.

Prompt-Based Pattern Summarization

Since PLM makes inferences by considering the whole sentence's information, we distill contextual patterns by prompting the PLM with masked words put in specific positions. As shown in Figure 2 we design two kinds of templates to extract concepts patterns for entities and the relation pattern. Instead of mapping the predicted words to a relation label, which needs human labor to build suitable label words, we simply take them as patterns.

Concept Pattern Summarization For each entity, a prompt is constructed by filling template T_1 with [MASK] tokens, corresponding entity mentions and learnable continuous tokens, which are fed to PLM M. The top n_c most likely words for [MASK] tokens are taken as concept patterns, take e_1 as an example, $f_{e_1} = \{w^c_i\}, i \in [1, n_c]$.

To learn continuous tokens for concept prediction, we take it as a multiple classification task and utilize P-Tuning structure (Liu et al. 2021). Given template T_1 with discrete tokens $[P_{0:m}]$, continuous trainable tokens $[\bar{P}]$ and input sentence $x = [x_{0:n}]$, the embedding of prompt is:

$$\Big\{e(x_{0:n}), \big[\bar{P}\big], e(e_i), e\big([P_{0:i}]\big), e\big([MASK]\big), e\big([P_{i+1:m}]\big)\Big\}$$

The ground-truth concepts are extracted from entities in the initial training set by adapting Microsoft Concept Graph (Wu et al. 2012). Then we compute the frequency for

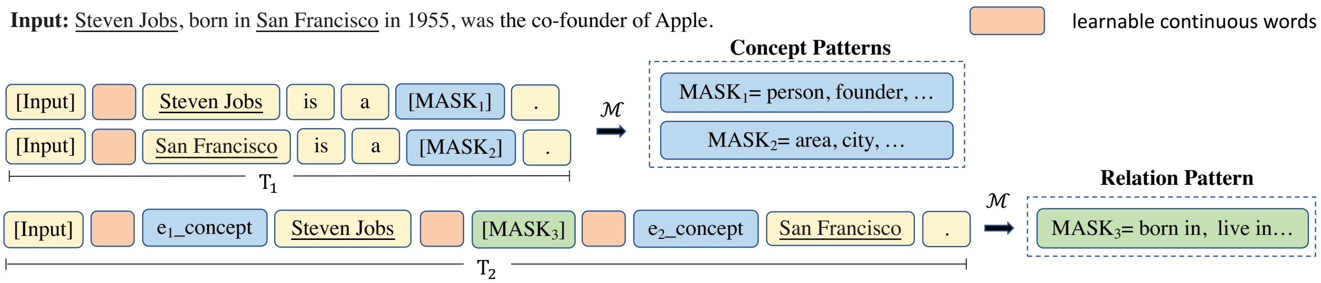

Figure 2: The procedures of summarizing patterns by prompting. e$_i$_concept refers to the most likely word for [MASK$_i$].

each concept and the top k_c most common ones are picked as concept classes $Y_c = \{y_1, y_2, ..., y_{k_c}\}$ and the corresponding concept mentions are taken as their label words, donated as $\{\phi(y_i)\}_{i=1}^{k_c}$. The P-Tuning objective is:

$$\mathcal{L}_{pt} = \frac{1}{|X|}\frac{1}{|k_c|}\sum_{x\in X}\sum_{i=1}^{k_c} y_i \log p\big([M] = \phi(y_i) \mid T_1(x)\big) \tag{3}$$

where [M] refers to the mask token [MASK].

Relation Pattern Summarization Similar with concept exploration, prompt is built by template T_2 with the entities' most likely concept words, which is fed to PLM M and the top n_r most likely words for [MASK] are taken as relation pattern $f_r = \{w_i^r\}, i \in [1, n_r]$.

We take it as a relation classification task and prompt-tune PLM M on the initial training set. To build label words automatically, each relation label split its mention as label words. For instance, the relation "city of birth" has label words {'city', 'birth'} (the stop word 'of' is discarded).

Similarity Measurement

Considering instances or rules expressing the same relation may have synonym words as patterns and the hard-matching manner leads to a low rule coverage. Thus, we embed the pattern's words by BERT (Devlin et al. 2019) and take the average embedding to represent the pattern. Finally, we obtain the matching score by computing their cosine similarity:

$$g(f^p, f^u) = Cos(e(f^p), e(f^u)) \tag{4}$$

$$e(f) = \frac{\sum e(w_i)}{|f|} \tag{5}$$

Weak Label Generation

Given an unlabeled instance u and rules set R, to create a weak label for u, we go through the whole rules set to find the rules that match u. When u is matched by multiple rules with conflicting labels, the majority voting mechanism is adapted and the label with the most matching rules is chosen as u's weak label.

Annotation Confidence Assume u's weak label is l and the matching rules with label l is R^l. We design a confidence metric and encourage the labeled results that are voted by rules with higher matching scores to get greater confidence:

$$Conf^l = \sum_{r_i \in R^l} s(r_i, u) \tag{6}$$

Self-Reviewing Module

In this stage, we embed the RE model into a self-training pipeline and devise *Self-Reviewing Module* to perform RE model training and summarize new rules from the unlabeled dataset iteratively. Based on it, new rules explored from unlabeled dataset by RE model are added to the rules set. Then the RE model could consolidate its learned knowledge by training on new high-quality data labeled by the updated rules set.

Specifically, in step t, given model m_{t-1} trained with the labels created by the previous rules set R_{t-1}, by adapting m_{t-1} to unlabeled dataset, instances with model-confirmed patterns will be assigned pseudo-labels with high confidence. New rules are built from the top n_R pseudo-labeled instances with the highest confidence and enrich R_{t-1} for a new rules set R_t to enlarge the rules set's coverage.

Finally, R_t is adapted to the unlabeled dataset and the top n_d weakly labeled data with highest annotated confidence are used to build new training set T_t, prepared for learning a new model m_t.

Rule-Induced Breakthrough Learning Module

Simply repeating the *Self-Reviewing Module* may narrow the model's comprehension for some relations and stick in specific patterns (as reported in Snowball (Gao et al. 2020)). To discover data with new patterns helpful for learning, an uncertainty-based sampling strategy is adapted to unlabeled dataset to obtain query data for human annotation. To maximize the effect of the human annotation budget, new rules are built from the human-labeled data to further cluster instances with similar patterns.

Uncertainty Measurement With the assumption that given a rule p, all instances matching p have similar patterns with p. For each instance u in unlabeled dataset, we summarize u's concept and relation patterns and treat it as a rule without relation label. Then we go through the rest unlabeled data and the first one that matches u is taken as a perturbation $\overline{u}$. By feeding u and $\overline{u}$ to the RE model, we could get the prediction y and $\overline{y}$. Finally we measure the model's uncertainty for u's pattern by computing the KL-divergence between y and $\overline{y}$:

$$KLD_u = \mathcal{D}_{KL}\big(p(y \mid u;\theta), p(\overline{y} \mid \overline{u};\theta)\big) \tag{7}$$

where the task model is parameterized by θ and denoted as $p(\cdot \mid \cdot;\theta)$

Rules Expansion In step $t+1$, given a model m_t trained in the previous step, by computing all the data's KLD in the unlabeled dataset (if one has no perturbation, its KLD is set to zero), we sort all instances by KLD decreasingly and choose top n_a ones for human annotation.

Finally, n_a new rules are built from the human-labeled results and the enriched rules set R_{t+1} could provide weak labels for instances with new patterns and help the model step out of the old "knowledge zone". The top n_d rule-annotated weakly labeled data with high annotated-confidence is utilized to build a new training set T_{t+1} for learning a new RE model m_{t+1}.

Model Training & Denoising

Inspired by COSINE (Yu et al. 2020), we try to alleviate the noisy weak labels' influence with the help of pseudo labels generated from the model in the last step. In step t, to learn a RE model m_t on the training set T_t, we apply model m_{t-1} to generate soft pseudo label y_{t-1} for each instance x. Then we weigh y_{t-1} by its entropy (donated as w_{t-1}) and compute loss $\mathcal{L}_p$ by weighted KL-divergence between the predicted distribution y_t from model m_t with y_{t-1}:

$$w_{t-1} = 1 - \frac{Ent(y_{t-1})}{log(|Y|)} \tag{8}$$

$$\mathcal{L}_p = \frac{1}{|\mathcal{C}|}\sum_{x\in\mathcal{C}} w_{t-1}(x)\mathcal{D}_{KL}(y_{t-1}, y_t) \tag{9}$$

where $|Y|$ is the number of labels. $\mathcal{C} = \{x \in T_t | w_{t-1}(x) \leq \xi\}$ is used to filer out unreliable pseudo labels (T_t is the training set and ξ is a threshold). We also introduce the cross-entropy loss:

$$\mathcal{L}_c = \sum_{x\in T_t} y \log p(y_t \mid x;\theta_t) \tag{10}$$

where y is the ground-truth label of x. For the Knowledge Stimulation stage, y refers to the labels of seeds, otherwise, y are the weak labels from the rules set. The final objective is $\mathcal{L} = \mathcal{L}_c + \mathcal{L}_p$. Notice that the model m_0 obtained in the Knowledge Stimulation stage only learns with $\mathcal{L}_c$, since there is no previous model providing pseudo labels for it.

Experiments

Datasets

Our experiments are conducted on two benchmark datasets, including TACRED (Zhang et al. 2017) and Re-TACRED (Stoica, Platanios, and Póczos 2021). The statistic of those datasets is shown in Table 1. To imitate the real scenario with insufficient labeled data, for each dataset we randomly sample **5%** training data as initial seeds D_{seed} and take the rest as unlabeled data D_u. Since the advantage of the large development set is against our label-scarce setting, as suggested in recent work (Gao, Fisch, and Chen 2021), we keep the development set D_{dev} of the same size as the seeds' size, donated as $|D_{dev}| = |D_{seed}|$. Notice that to avoid bias, each relation is assigned the same number of initial seeds (86 seed data per relation for TACRED and 73 seed data per relation for Re-TACRED) and our F1 metric setting follows the RE task metric setting in WRENCH (Zhang et al. 2021), a weak supervision benchmark platform for standardized evaluation.

Datasets	Class Num.	Train	Dev	Test
TACRED	41	68124	22631	15485
Re-TACRED	40	58465	19584	13418

Table 1: Statistics of the datasets in our experiments.

Parameters Settings

1) In *Rule Generator*, the threshold TH is set to 0.97. After feeding the prompt to PLM, $n_c = 3$ words are picked to represent concept patterns and $n_r = 5$ words for relation patterns. 2) If one step is in *Self-Reviewing Module*, then $n_R = 120$ self-inferred rules are generated. 3) If one step is in *Rule-induced Breakthrough Learning Module*, it will ask annotation for $n_a = 60$ most confusing data. To make **KICE** more reproducible, the human annotation for each data is the same as its original label in the dataset. In each step, the training set is built by $n_d = 200$ weakly labeled data with the highest annotated confidence. In the model training procedure, the threshold ξ is set to 0.5.

For dataset TACRED and Re-TACRED, after the Knowledge Stimulation step, we report the **KICE**'s performance after 4 iterations with Knowledge Consolidation and Knowledge Expansion steps conducted alternatively.

Baselines

KICE's RE Model with Rich Data: We evaluate our RE model's performance with full clean labeled training data for each dataset.

Baselines with PLM Knowledge: 1) P-Tuning (Liu et al. 2021) proposes to build task-related prompts with learnable continuous tokens. Instead of utilizing P-Tuning for pattern extraction (as in **KICE**'s *Rule Generator*), we take it as a baseline to predict relations directly. 2) **KnowPrompt** (Chen et al. 2022) injects the knowledge of relation labels to prompt with learnable type words and answer words to alleviate human labor in label words construction. 3) **PTR** (Han et al. 2021) is a prompt-tuning method utilizing manual rules to build flexible prompts with several sub-prompts for multi-class classification tasks. 4) **COSINE** (Yu et al. 2020) proposes a denoising training manner for fine-tuning PLM with weakly labeled data.

Baselines with Self-Explored Knowledge: 1) **NERO** (Zhou et al. 2020) trains a soft matching module with the classifier to enlarge the manual rules' coverage. 2) **DualRE** (Lin et al. 2019) trains a retrieval module together with a relation classifier to continuously retrieve high-quality instances from unlabeled data and improve the classifier. 3) **Snowball** (Gao et al. 2020) first pre-trains the Relational

Labeled Data Num.	Methods	Extra Human Anno.	TACRED (F1)	Re-TACRED (F1)
100% Training Data	KICE w. Rich Data	w/o	87.23	88.33
5% Training Data	P-Tuning	w/o	25.42	32.54
	KnowPrompt	w/o	35.61	63.98
	PTR	w/o	37.65	55.37
	COSINE	w/o	41.00	57.82
	NERO	270 (Rules)	56.55	42.99
	DualRE	w/o	50.50	63.70
	Snowball	w/o	23.03	23.56
	PRBOOST	1000 (Rules)	48.10	-
	VAE	w/o	26.56	33.42
	KICE	120 (Insts.)	**79.84**	**68.11**

Table 2: Overall performance on two datasets. We evaluate the model's performance with different proportions of training data to simulate the scenarios with sufficient/insufficient labeled data. Since the 5% labeled data is randomly sampled from the training set, for each method we take the average of 3 runs as its final result. Extra Human Anno. refers to the number of human-annotated rules/instances besides the training data and 'w/o' means no additional data is used in this method.

Siamese Network with a classifier in data-rich relations. Then reliable instances are explored by the two modules to improve themself on few-data relation learning.

Baseline with Interactive Human's Knowledge: PRBOOST (Zhang et al. 2022) generates candidate rules automatically from the model's large-error instances in a fixed labeled dataset and ask human annotators to pick high-quality rules from candidates to provide weakly labeled data for model training.

Baseline with KB's knowledge: We include the distant supervised (DS) baseline since it also aims to alleviating the labeling labor. **VAE** (Christopoulou, Miwa, and Ananiadou 2021) is a DS framework trained on knowledge base labeled data and improve sentence expressivity by sentence reconstruction. Following the assumption of DS, for each seed, data in D_u is annotated with the seed's label if they share the same entity pair and added to DS training set.

Overall Performance

Table 2 shows the performance of **KICE** and baselines on two datasets. **KICE** outperforms all baselines on both TACRED and Re-TACRED datasets.

The evaluation of P-Tuning, KnowPrompt and PTR shows that the pretty limited training data could hinder those prompt-tuning methods to learn the most suitable prompts and achieve their best performance. Thus, the approach to learning from PLM's knowledge designed in **KICE** is more reasonable, which summarizes patterns from PLM's output for weakly labeling in a soft-matching manner.

The performance of NERO, DualRE, Snowball and **KICE**'s RE Model w. Rich Data shows the gap between self-explored knowledge simulated from few initial labeled data/rules and the knowledge contained in rich training data. Compared with the best baselines among them, **KICE** achieves a great improvement (+23.29% F_1 in TACRED) and narrows the gap with only 120 extra human-labeled data.

Compared with PRBOOST, which queries 100 human-annotated rules for each iteration and totally conducts 10 iterations, **KICE** gets F_1 score 31.74% higher than it by asking 60 human-annotated instances in each Knowledge Expansion step and totally conducting 2 steps. Furthermore, the annotation on instances is simpler than that on rules and needs less professional knowledge. Thus, **KICE** is more labor-reducing and time-saving for annotators.

By extracting triples from 5% training data as prior knowledge to obtain distantly supervised data, we simulate a scenario where the knowledge provided by KB is uncompleted, which is common in many emerging industries. The evaluation of VAE under this scenario shows that the limited prior knowledge is still challenging for it to achieve great performance, while **KICE** could continuously explore new knowledge by the *Self-Reviewing Module* and *Rule-induced Breakthrough Learning Module* and gradually improve model's performance.

Qualitative Results To analyze when to stop the iteration in **KICE**, we totally conduct 7 steps and evaluate the model's F_1 in each step on TACRED. Therefore, we can take a closer look at the change of model performance during the training stages in Figure 3. From the figure, we can observe that the model performance increases gradually as **KICE** training progresses (from 33.24% to 79.84%), outperforming the improvement of DualRE (from 44.9% to 50.5%), which is one of the best baselines with gradually learning procedure. This demonstrates that **KICE** evolves knowledge more helpful for model learning. Since the improvement after step 4 is much smaller (+0.39%) compared with previous steps, to save the human labor for extra annotation, we report the performance in step 4 as the final result of **KICE**.

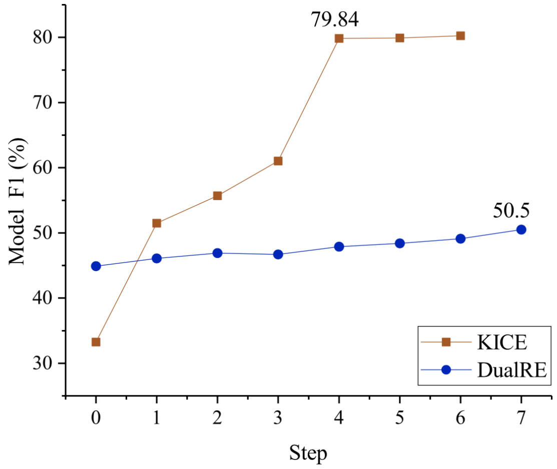

Figure 3: Model F_1 of each step in **KICE** and **DualRE**.

Analysis on Knowledge Collaboration

To illustrate alternatively conducting Knowledge Consolidation (KC) step and Knowledge Expansion (KE) step could collect knowledge helpful for model learning effectively and significantly, we show the F_1 score of RE model on TACRED obtained in each iteration within three knowledge collection manner: 1) Standard **KICE**; 2) **KICE** without KE step (only conducting KC step after Knowledge Simulation); 3) **KICE** without KC step (only conducting KE steps after Knowledge Simulation).

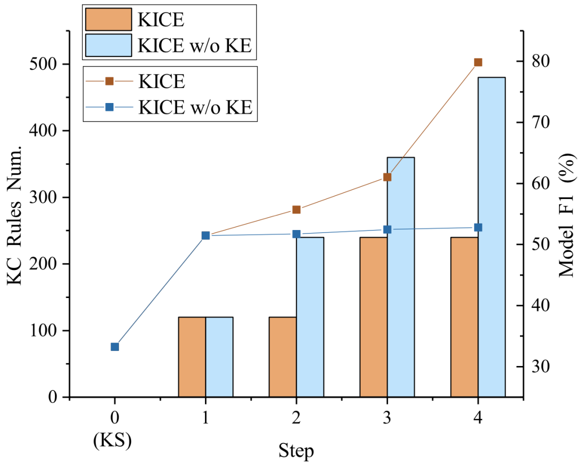

Figure 4: Model F_1 under Standard **KICE** and **KICE** without KE step. KC Rules Num. refers to the total number of rules collected by Self-Reviewing Module in KC step.

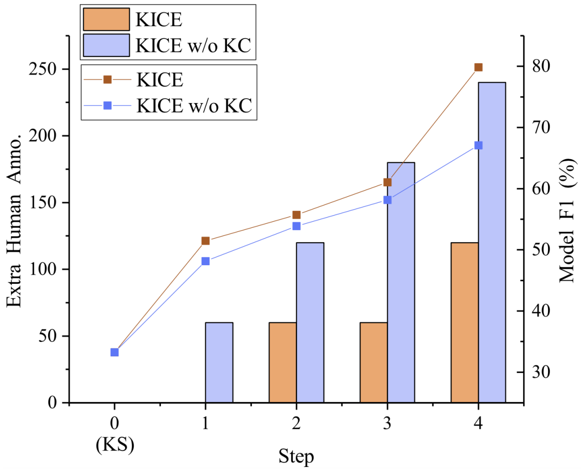

Figure 5: Model F_1 under Standard **KICE** and **KICE** without KC step. Extra Human Anno. refers to the total number of extra human annotated instances.

The results are shown in Figure 4 and Figure 5. Notice that step 0 in each figure refers to Knowledge Stimulation (KS) stage. On one hand, by removing the KE step and only conducting KC steps after Knowledge Simulation, we found that the model performance grows much slower after step 1 (total +1.08%), which demonstrates the model overfits on the self-reviewing rules and fails to discover new patterns as rules number increasing. On the other hand, without the Knowledge Consolidation step, the human annotation's effect on model learning gets smaller in each step, which means to get the equivalent model's performance, **KICE** without KC needs much more labeling labor and is more time-consuming than **KICE**.

To demonstrate the effect of PLM's knowledge for patterns summarization, an entity pair matching manner is designed to replace the PLM-summarized patterns matching manner in **KICE** (donated as **KICE** w. ep). To be more specific, given an unlabeled instance u, it matches the rule p if they share the same entity pair. If u is matched with rules with conflict labels, the label voted by most rules is taken as its weak label. After conducting **KICE** w. ep on the TACRED with 5% training data for 5 iterations, we obtain a RE model with the performance of 36.11% F_1 on the test set, while standard **KICE** achieves 79.84% F_1. This gap shows the patterns summarized from PLM knowledge in **KICE** could provide weak labels more helpful for model learning.

Conclusion

In this paper, we propose a knowledge consolidation and expansion framework, **KICE**, for evolving knowledge iteratively with the guidance of PLMs to help RE model learning under insufficient labeled data. Our framework can continuously discover new knowledge in the form of rule-based patterns from the unlabeled dataset, enhance the effect of extra human-labeled data and significantly reduce the human annotation efforts. Experiments conducted on TACRED and Re-TACRED shows **KICE** achieves great improvement in the performance of relation extraction with little human labeling labor and outperforms all the baselines. Our work shows that the combination and collaboration of different sources of knowledge help alleviate manual annotation efforts and is also generally applicable to other tasks.

Acknowledgments

This work has been supported in part by the Zhejiang NSF (LR21F020004), NSFC (No. 62272411), Alibaba-Zhejiang University Joint Research Institute of Frontier Technologies, Chinese Knowledge Center of Engineering Science and Technology (CKCEST).

References

Boecking, B.; Neiswanger, W.; Xing, E.; and Dubrawski, A. 2020. Interactive Weak Supervision: Learning Useful Heuristics for Data Labeling. In *International Conference on Learning Representations*.

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.

Chen, Z.; and Qian, T. 2022. Description and demonstration guided data augmentation for sequence tagging. *World Wide Web*, 25(1): 175–194.

Christopoulou, F.; Miwa, M.; and Ananiadou, S. 2021. Distantly Supervised Relation Extraction with Sentence Reconstruction and Knowledge Base Priors. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 11–26.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT (1)*.

Gao, T.; Fisch, A.; and Chen, D. 2021. 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)*, 3816–3830.

Gao, T.; Han, X.; Xie, R.; Liu, Z.; Lin, F.; Lin, L.; and Sun, M. 2020. Neural snowball for few-shot relation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 7772–7779.

Han, X.; Zhao, W.; Ding, N.; Liu, Z.; and Sun, M. 2021. PTR: Prompt Tuning with Rules for Text Classification. *arXiv e-prints*, arXiv–2105.

Han, X.; Zhu, H.; Yu, P.; Wang, Z.; Yao, Y.; Liu, Z.; and Sun, M. 2018. FewRel: A Large-Scale Supervised Few-shot Relation Classification Dataset with State-of-the-Art Evaluation. In *EMNLP*.

Hsieh, C.-Y.; Zhang, J.; and Ratner, A. 2022. Nemo: Guiding and Contextualizing Weak Supervision for Interactive Data Programming. *arXiv e-prints*, arXiv–2203.

Lin, H.; Yan, J.; Qu, M.; and Ren, X. 2019. Learning dual retrieval module for semi-supervised relation extraction. In *The World Wide Web Conference*, 1073–1083.

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.

Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; and Tang, J. 2021. GPT Understands, Too. *arXiv preprint arXiv:2103.10385*.

Margatina, K.; Vernikos, G.; Barrault, L.; and Aletras, N. 2021. Active Learning by Acquiring Contrastive Examples. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 650–663.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Qu, M.; Ren, X.; Zhang, Y.; and Han, J. 2018. Weakly-supervised relation extraction by pattern-enhanced embedding learning. In *Proceedings of the 2018 World Wide Web Conference*, 1257–1266.

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.

Vrandečić, D.; and Krötzsch, M. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10): 78–85.

Wu, W.; Li, H.; Wang, H.; and Zhu, K. Q. 2012. Probase: A probabilistic taxonomy for text understanding. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, 481–492.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R. R.; and Le, Q. V. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Yu, Y.; Zuo, S.; Jiang, H.; Ren, W.; Zhao, T.; and Zhang, C. 2020. Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach. *arXiv preprint arXiv:2010.07835*.

Zhang, J.; Yu, Y.; Li, Y.; Wang, Y.; Yang, Y.; Yang, M.; and Ratner, A. 2021. WRENCH: A Comprehensive Benchmark for Weak Supervision. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Zhang, R.; Yu, Y.; Shetty, P.; Song, L.; and Zhang, C. 2022. Prompt-Based Rule Discovery and Boosting for Interactive Weakly-Supervised Learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 745–758.

Zhang, Y.; Zhong, V.; Chen, D.; Angeli, G.; and Manning, C. D. 2017. Position-aware attention and supervised data improve slot filling. In *Conference on Empirical Methods in Natural Language Processing*.

Zhou, W.; Lin, H.; Lin, B. Y.; Wang, Z.; Du, J.; Neves, L.; and Ren, X. 2020. Nero: A neural rule grounding framework for label-efficient relation extraction. In *Proceedings of The Web Conference 2020*, 2166–2176.