KnowCoder-X: Boosting Multilingual Information Extraction via Code

Yuxin Zuo[1,2,3*], Wenxuan Jiang[4*], Wenxuan Liu[1,2,3*], Zixuan Li[1,2†], Long Bai[1,2], Hanbin Wang[5], Yutao Zeng[1], Xiaolong Jin[1,2,3†], Jiafeng Guo[1,2,3], Xueqi Cheng[1,2,3]

[1]Key Laboratory of Network Data Science and Technology,
Institute of Computing Technology, Chinese Academy of Sciences
[2]State Key Laboratory of AI Safety
[3]School of Computer Science, University of Chinese Academy of Sciences
[4]School of Software, Northeastern University
[5]School of Software, Peking University
{zuoyuxin24s, liuwenxuan2024z, lizixuan24h, jinxiaolong}@ict.ac.cn

Abstract

Empirical evidence indicates that LLMs exhibit spontaneous cross-lingual alignment. However, although LLMs show promising cross-lingual alignment in Information Extraction (IE), a significant imbalance across languages persists, highlighting an underlying deficiency. To address this, we propose KnowCoder-**X**, a powerful code LLM with advanced cross-lingual and multilingual capabilities for universal IE. Firstly, it standardizes the representation of multilingual schemas using Python classes, ensuring a consistent ontology across different languages. Then, IE across languages is formulated as a unified code generation task. Secondly, we conduct IE cross-lingual alignment instruction tuning on the translated instance prediction task to enhance the model's cross-lingual transferability. During this phase, we also construct a high-quality and diverse bilingual IE parallel dataset with $\mathbf{257k}$ samples, called **ParallelNER**, synthesized by our proposed robust three-stage pipeline, with manual annotation to ensure quality. Although without training in $\mathbf{29}$ unseen languages, KnowCoder-**X** surpasses ChatGPT by $\mathbf{30.17}\%$ and SoTA by $\mathbf{20.03}\%$, thereby demonstrating superior cross-lingual IE capabilities. Comprehensive evaluations on $\mathbf{64}$ IE benchmarks in Chinese and English under various settings demonstrate that KnowCoder-**X** significantly enhances cross-lingual IE transfer through boosting the IE alignment. Our code and dataset are available at: `https://github.com/ICT-GoKnow/KnowCoder`.

1 Introduction

Cross-lingual Information Extraction (IE) focuses on automatically extracting structured information from texts in unseen languages, following manually designed schemas. Cross-lingual schema understanding and representation have long been a challenge. Large Language Models (LLMs), trained on massive multilingual corpora, have significantly advanced multilingual language processing (Brown et al., 2020; Touvron et al., 2023). Previous studies (Qi et al., 2023; Wang et al., 2024; Gao et al., 2024; Li et al., 2024a) have shown that LLMs exhibit spontaneous cross-lingual alignment to facilitate the transfer of abilities and knowledge across languages. Our findings suggest the presence of this alignment in IE, indicating a strong potential for improving the IE cross-lingual transfer. We first define IE parallel data, which is across various languages that share the same schema, sentences, and extracted instances. Previous studies used label projection to generate such data to alleviate the challenges of IE in low-resource (Kolluru et al., 2022; Hennig et al., 2023; Rios et al., 2024). We construct an IE parallel dataset in Chinese based on CoNLL 2003 (Sang and Meulder, 2003) manually, to evaluate the LLaMA2-7B trained on English Named Entity Recognition (NER) datasets used in KnowCoder (Li et al., 2024b). Table 1 shows the results. After training on English IE datasets, there was a notable enhancement in the parallel dataset of Chinese IE (5.9→52.7), strongly supporting the presence of IE cross-lingual alignment. However, the significant performance gaps between the two languages (95.1 *vs* 52.7) indicate that the alignment remains weak. To address this, we propose two strategies to enhance IE cross-lingual alignment in LLMs by aligning the schema and extraction process across different languages, respectively:

Model	CoNLL 2003	CoNLL 2003 (zh)
LLaMA2-7B	26.9	5.9
LLaMA2-7B (w/ training)	95.1	52.7

Table 1: Results on CoNLL 2003 and CoNLL 2003 (zh). The latter is the constructed Chinese parallel dataset.

Firstly, we standardize IE schemas across languages, particularly non-English ones, into unified

[*] Co-first authors.
[†] Corresponding authors.

Findings of the Association for Computational Linguistics: ACL 2025, pages 14486–14509
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

Python classes. X-GEAR (Huang et al., 2022) introduced a language-agnostic output template to maintain multilingual IE uniformity, aiding cross-lingual transfer. However, this uniformity is not thorough due to the absence of a schema in the input context, which hinders both cross-lingual transfer and multilingual schema understanding. Moreover, its highly customized input and output limit scalability to UIE and LLM-based IE. Thus, we introduce a unified UIE schema representation in the input and output, ensuring thorough uniformity in multilingual IE to improve cross-lingual alignment within LLMs. We also observe that object-oriented features in code are well-suited for unified schema representation and knowledge sharing across languages. Although code-based approaches (Wang et al., 2022; Li et al., 2024b; Liu et al., 2025) have shown promising results in IE, their effectiveness in cross-lingual IE still needs to be validated. Therefore, we leverage Python classes to represent multilingual schemas, ensuring consistent ontology representation across languages and facilitating efficient IE knowledge transfer. The consistent schema allows the model to learn and efficiently share the knowledge of the same ontology across languages.

Secondly, we introduce an IE cross-lingual alignment instruction tuning phase. We first propose a task that predicts target language instances, aiding in the alignment of the extraction process (see Figure 1 for an example). Using prediction tasks with parallel data to enhance multilingual alignment through instruction tuning is a widely adopted paradigm (Zhu et al., 2023; Gao et al., 2024). The key challenge is ensuring that extracted instances maintain semantic consistency with their source-language counterparts while appearing in the target language sentence. Unlike directly predicting complete IE parallel data, which focuses on aligning sentences, our method prioritizes the alignment of translated instances, which is the central goal of IE cross-lingual alignment.

To use high-quality IE parallel data for the phase, we propose an automatic three-stage LLM-based pipeline, which achieves 99% average faithfulness across 10 languages evaluated on the label projection benchmark WikiANN (Pan et al., 2017). Utilizing the pipeline, we construct a high-quality and diverse NER English-Chinese parallel dataset ParallelNER, using the GPT-4o series model supplemented with manual annotation. By distilling the IE alignment capabilities of advanced proprietary models, the IE cross-lingual alignment phase enhances cross-lingual generalization in IE tasks.

Ultimately, we obtain the KnowCoder-**X** through Chinese and English IE instruction tuning. We evaluate the KnowCoder-**X** on 64 benchmarks for the NER, Relation Extraction (RE), Event Detection (ED), and Event Argument Extraction (EAE) tasks in Chinese and English. KnowCoder-**X** exhibits superior cross-lingual generalization across 29 unseen diverse languages, and surpasses ChatGPT by 30.17% and SoTA by 20.03%. Moreover, it achieved an impressive average improvement of 11.43% over the SoTA across 20 low-resource African languages. In the supervised evaluation, KnowCoder-**X** consistently ranks within the top-2 results across 40 (of 42) benchmarks. Notably, KnowCoder-**X** achieves SoTA across all Chinese IE benchmarks, effectively demonstrating the strength of knowledge transfer from English IE through our method. Our contributions can be summarized as:

- The code-based multilingual IE method of KnowCoder-**X** unifies the representation of schemas across different languages, thereby boosting cross-lingual transfer.
- We incorporate an IE cross-lingual alignment phase to improve cross-lingual transfer. This phase involves instruction tuning on a newly proposed translated instance prediction task.
- We propose a robust three-stage IE parallel data construction pipeline and construct a high-quality bilingual parallel NER dataset ParallelNER with $257,190$ samples to provide valuable resources to the community.

2 Related Work

Transfer Learning for cross-lingual IE Huang et al. (2017) introduced a zero-shot learning method for EE, facilitating generalization across unseen languages. CLaP (Huang et al., 2022) utilized generative language models for cross-lingual EAE. Prompt-XRE (Hsu et al., 2023) enhanced cross-lingual RE through prompt-based learning, reducing reliance on large annotated datasets. Zubillaga et al. (2024) emphasized the significance of typology in cross-lingual EE, particularly for low-resource languages like Basque. In contrast, KnowCoder-**X** focuses on leveraging the spontaneous alignment of LLMs to facilitate cross-lingual IE transfer on all IE tasks.

INSTRUCTION

```
# The following is an example of Chinese Named Entity Recognition (NER)
 in an object-oriented programming paradigm.
In this example, several entity classes are defined, and then objects o
f these classes are instantiated to correspond with the entities in the
 given sentence.
# Please study this example and complete the English NER task according
ly.
# Input (Chinese NER):
class Entity:
        """
        描述：实体（Entity）指的是 …
        """
        def __init__(self, name: str):
                self.name = name
…
sentence = "三种主要的学习范式是监督学习，无监督学习和强化学习。"
# Output (Chinese NER):
results = [
        BranchOfScience("监督学习"),
        BranchOfScience("无监督学习"),
        BranchOfScience("强化学习")
]
# Input (English NER):
class Entity:
        """
        Description: An entity is a …
        """
        def __init__(self, name: str):
                self.name = name
…
sentence = "The three major learning paradigms are supervised learning,
unsupervised learning and reinforcement learning."
```

COMPLETION

```
# Output (English NER):
results = [
        BranchOfScience("supervised learning"),
        BranchOfScience("unsupervised learning"),
        BranchOfScience("reinforcement learning")
]
```

Figure 1: An example of instruction-tuning data used in the IE cross-lingual alignment phase. The highlighted span in the completion and the two same color highlighted spans in the instruction must ensure semantic consistency and be exactly matched, respectively.

LLMs for Multilingual IE YAYI-UIE (Xiao et al., 2023) proposes a chat-enhanced instruction tuning framework for UIE. IEPILE (Gui et al., 2024) collects a comprehensive bilingual IE instruction corpus, and B^{2}NER (Yang et al., 2024) further designs a universal entity taxonomy. However, previous works overlook the mutual influence mechanisms between schemas in different languages and lack a unified representation.

Code-Based IE Code-based IE aims to formulate the IE as a code generation task. CodeIE (Li et al., 2023) first leverages the Code-LLM and recasts IE tasks into Python function completion tasks. Code4Struct (Wang et al., 2022) represents the schema in Python classes for the EAE task. Code4UIE (Guo et al., 2023) further constructs code-based schema for all the UIE tasks. Besides, KnowCoder (Li et al., 2024b) utilizes class inheritance, class methods, and type hints to further enrich the schema representation. GoLLIE (Sainz et al., 2024) incorporates guidelines within class comments. However, these works focus on the English IE task, and we are dedicated to exploring the significant role of code in multilingual IE.

3 Code-Based Multilingual IE

KnowCoder-**X** formulates the IE tasks across various languages using a unified code generation framework. In this section, we introduce the instruction and completion of the proposed code-based multilingual IE. The instruction consists of two parts (§3.1): (1) schema representation code across languages, including class definition and comments; (2) task prompts for completion. Subsequently, we present the completion format (§3.2).

3.1 Instruction Format

Class Definition The IE schema comprises multiple concepts, each of which consists of a name and a set of attributes. The three fundamental concepts of IE schemas `[Entity, Relation, Event]` are first defined as base classes. For each concept of a schema, a corresponding class is subsequently defined. Each concept inherits from its respective base class by default. Then, we define the attributes of each concept (such as argument roles) as input arguments of the constructor *__init__()*. Specially, for IE in non-English languages, we first create a mapping from the original schema to the English schema. The mapping standardizes the representation of the same ontology across different languages by utilizing the unified Python class. For example, the entity concept 사람 in Korean and 人物 in Chinese are both mapped to the `class` **`PER(Entity)`**, ensuring consistent semantic representation across linguistic boundaries to align the schema in different languages.

Class Comments We utilize class comments to provide clear definitions of concepts similar to GoLLIE (Sainz et al., 2024) and KnowCoder (Li et al., 2024b) for the understanding of concepts in different languages. Class comments comprise two components: 1) examples contain instances corresponding to the concept. 2) a description that explains the concept; For the examples part, we sample up to 10 instances for each concept by frequency. For the description part, we follow the annotation guidelines of datasets by default. We propose an LLM-based approach to generate descriptions for datasets where guidelines are unavailable. To enable the LLM to better understand the concepts and the annotation style of the dataset, we first sample 10 instances to prompt the LLM to summarize an initial description. To avoid providing excessive examples at once to hinder effective induction summarization, we then sample 20 instances for the LLM to iteratively refine the description based on each one. Specifically, for each instance, if it cannot be categorized into the corresponding concept according to the description, the LLM will adjust the inappropriate parts of the current description accordingly. We use the `GPT-4o-2024-08-06` as the LLM. Appendix A provides details on the description generation procedure.

Task Prompt The task prompt includes a docstring defining the task and a string variable *sentence* that holds the text. The prompt is then concatenated with the schema to form the instruction.

3.2 Completion Format

The completion represents the final output generated by the model, which begins after the assignment operation (i.e., *results* =). Each completion is a list, where each element is an instantiated object, representing the extracted structured knowledge instance. Additionally, we utilize the *exec()* function in Python to execute the instructions and completions of all code-based data in this work to ensure quality. We also use this function to execute the generated code to extract the results.

4 IE Cross-Lingual Alignment

KnowCoder-**X** adopts a two-phase instruction tuning framework. The first phase is the IE cross-lingual alignment phase, where we propose the task and data. Subsequently, we conduct instruction tuning on Chinese and English UIE tasks with 46 IE datasets to obtain KnowCoder-**X**. Appendix H shows examples of instruction tuning data. Appendix G shows the detailed dataset statistics. In this section, we introduce the IE cross-lingual alignment phase. First, we introduce the instruction and completion of the translated spans prediction task (§4.1). To obtain high-quality IE parallel data for this training phase, we propose a three-stage automatic pipeline for constructing IE parallel data (§4.2) and creating ParallelNER (§4.3).

4.1 Translated Instances Prediction Task

The translated instances prediction task is the training objective of this phase, aiming to perform IE in the target language. The instruction starts with a task description, prompting the LLM to predict translated instances by concatenating the input-output pairs of source-language IE data with the input of target-language IE data to form the instruction. The completion contains the output of the target language data. Figure 1 shows an example.

Algorithm 1: Sample Translation

Initialize : Source Sentence s^{src}
Source Spans I^{src}
Output : Target Sentence s^{tgt}
Target Spans I^{tgt}
Function JointTranslate(s^{src}, I^{src}):
  $prompt = \texttt{TemplateT}\ (s^{src}, I^{src})$;
  // Obtain Prompt for Joint Translation.
  $I^{tgt}, s^{tgt} = \texttt{LLM}(prompt)$;
  return I^{tgt}, s^{tgt} ;
Function SpanRephrase($I^{tgt}, I^{src}, s^{tgt}$):
  $Flag = False$;
  foreach (i_m^{tgt}, i_m^{src}) *in* (I^{tgt}, I^{src}) **do**
    if i_m^{tgt} *not in* s^{tgt} **then**
      $prompt = \texttt{TemplateI}$
      $(s^{tgt}, i_m^{src}, i_m^{tgt})$;
      // Obtain Prompt for Span Rephrase.
      $i_m^{cor} = \texttt{LLM}\ (prompt)$;
      if i_m^{cor} *in* s^{tgt} **then**
        $I^{tgt}.update(i_m^{tgt}, i_m^{cor})$;
      else
        $Flag = \text{True}$;
  return $I^{tgt}, Flag$;
Function SentenceRephrase(I^{tgt}, s^{src}):
  $prompt = \texttt{TemplateS}\ (I^{tgt}, s^{src})$;
  // Obtain Prompt for Sentence Rephrase.
  $s^{tgt} = \texttt{LLM}\ (prompt)$; **return** s^{tgt} ;
Function Main(s^{src}, I^{src}):
  $I^{tgt}, s^{tgt} = \texttt{JointTranslate}(s^{src}, I^{src})$;
  $I^{tgt}, Flag = \texttt{SpanRephrase}(I^{tgt}, I^{src}, s^{tgt})$;
  if *Flag == True* **then**
    $s^{tgt} = \texttt{SentenceRephrase}(I^{tgt}, s^{src})$;
  return s^{tgt}, I^{tgt}

4.2 IE Parallel Data Construction Pipeline

The construction of IE parallel data involves two key components: sample translation (i.e., label projection) and schema translation. The sample consists of sentences and spans, where spans refer to the annotated segments within the sentence that are relevant to the IE task (Kripke and Munitz, 1971; Chen and Yuille, 2004). Given the sentence s^{src} and span list $I^{src} = \{i_1^{src}, i_2^{src}, ..., i_n^{src}\}$ in the source language, label projection aims to obtain the sentence s^{tgt} and span list $I^{tgt} = \{i_1^{tgt}, i_2^{tgt}, ..., i_n^{tgt}\}$ in the target language. Schema translation can be carried out through machine translation and manual annotation. The primary challenge lies in the sample translation, which involves two key issues (Parekh et al., 2024): 1) **Inaccurate Translation**: Either the spans or the sentences are inaccurately translated; 2) **Missing Spans**: The translated spans may be missing from the translated sentence.

Traditional label projection methods (Dou and Neubig, 2021; Chen et al., 2023; Parekh et al., 2024), such as CLaP, typically first generate s^{tgt}, and then obtain I^{tgt} through label retrieval, word alignment, or contextual translation. However, these methods have a critical flaw: errors introduced during the generation of s^{tgt} are propagated into the process of obtaining I^{tgt}. For example, during the translation of s^{src} to get s^{tgt}, i_k^{src} may be mistranslated or merged, especially when i_k^{src} is a pronoun such as "his", leading to the failure in obtaining the correct i_k^{tgt}. To address the issue, we propose a three-stage pipeline that implements joint translation and multi-grained rephrasing. Appendix I shows all prompts used in the pipeline.

Stage 1: Joint Translation We propose the first stage to ensure semantic consistency between the translated sentence s^{tgt} and the span list I^{tgt}. Compared to previous methods, where s^{tgt} and I^{tgt} are obtained sequentially, this stage utilizes LLMs to simultaneously translate and generate both s^{tgt} and I^{tgt}, thereby reducing error accumulation through *parallel processing*. This stage provides s^{src} and I^{src} as a comprehensive context that introduces inner constraints on the two key issues, thus enabling a more accurate and natural translation. Specifically, we use a predefined template to format the sentence s^{src} along with spans I^{src} to serve as a prompt for the LLM to perform joint translation.

Stage 2: Span Rephrase Stage 2 and 3 tackle the missing spans issue by rephrasing. In Stage 2, i_k^{tgt} corresponding to i_k^{src} is identified from s^{tgt}, similar to the CLaP (Parekh et al., 2024), which allows the rephrasing of the previously translated ones to further address the missing spans issue that persists after Stage 1. Specifically, if $i_m^{tgt} \notin s^{tgt}$, we prompt the LLM to find the span $i_m^{tgt-cor}$ in s^{tgt} that semantically matches with i_m^{src}. If $i_m^{tgt-cor} \in s^{tgt}$, i_m^{tgt} will be rephrased as $i_m^{tgt-cor}$.

Stage 3: Sentence Rephrase However, due to the inherent issue of error accumulation in the traditional *sentence-then-span* approach employed by CLaP-like methods, we introduce Stage 3. In this stage, we rephrase the translated sentence that is a *span-then-sentence* strategy, thereby mitigating the error propagation of the *sentence-then-span* approach and addressing the missing spans issue that remains unresolved. Specifically, if not all $i_m^{tgt} \in s^{tgt}$, we prompt the LLM to rephrase s^{tgt} based on s^{src} to ensure all i_m^{tgt} are included.

This pipeline is capable of generating IE parallel data across any two languages, as demonstrated in Algorithm 1. By integrating three translation strategies, *parallel processing*, *sentence-then-span*, and *span-then-sentence*, we have achieved unprecedented performance on the label projection task, approaching an accuracy rate of nearly 100%. Following Parekh et al. (2024), we evaluate the pipeline on the label projection benchmark WikiANN (Pan et al., 2017), a multilingual IE dataset, and achieve an average 99% on the faithfulness evaluation in 10 English-centric bilingual settings. Appendix B shows the detailed settings and results.

4.3 ParallelNER

Since alignment training requires including bilingual IE samples within a single prompt, the excessive concepts in RE and EE result in insufficient context length of LLMs. Thus, we construct NER parallel data for the phase. NER serves as a shared and foundational task in the IE framework, enabling entity identification through spotting abilities to support all IE tasks (Lu et al., 2022).

Using the pipeline, we construct a high-quality English (en) ⇔ Chinese (zh) NER parallel dataset, ParallelNER. To ensure linguistic diversity, we use high-quality datasets of each language: WikiNeural (Tedeschi et al., 2021) (en⇒zh) and CLUENER2020 (Xu et al., 2020) (zh⇒en), resulting $\mathbf{257,190}$ samples. We use the `GPT-4o-mini` as the base LLM of the pipeline. If the process fails at any stage, such as occurring Missing Spans, we employ `GPT-4o-2024-08-06` to facilitate reprocessing. With our pipeline, the missing spans issue is rarely observed, with a rate of $15/10000$ in CLUENER2020 and $82/92720$ ($8.84‱$) in WikiNeural. For the 97 samples above, we conduct further manual annotation to ensure the quality. We utilize ParallelNER to construct the training data for this phase in both language directions (en⇔zh) to facilitate cross-lingual transfer. Appendix G.1 shows the statistics of ParallelNER.

5 Experimental Settings

5.1 Implementation Details

We use span-based offset Micro-F1 to evaluate the methods. For NER, an entity is considered correct if the span and type match a golden annotation. For RE, a relation is considered correct if its type, subject entity, and object entity match a golden annotation. For ED, an event is valid if its trigger and type match a golden annotation. For EAE, given event type and trigger, an argument is valid

Model	English	Chinese	*Cross-Lingual*									***Avg**$_{cross}$*	***Avg***
			German	Spanish	Dutch	Russian	Bengali	Persian	Hindi	Korean	Turkish		
ChatGPT†	37.20	18.80	37.10	34.70	35.70	27.40	23.30	25.90	27.30	30.00	31.90	30.37	29.94
YAYI-UIE (Xiao et al., 2023)	52.04	37.89	41.78	42.44	38.02	30.97	15.97	21.96	20.93	26.45	30.74	29.92	32.65
IEPILE (Gui et al., 2024)	53.19	39.26	39.87	42.41	35.69	28.69	12.26	23.93	18.98	27.56	26.93	28.48	31.71
GLiNER (Zaratiana et al., 2024)	41.70	24.30*	39.50	42.10	38.90	33.30	**25.90**	30.20	27.80	28.70	30.00	32.93	32.95
B^{2}NER (Yang et al., 2024)	54.80	45.40	36.60	46.00	43.00	33.90	-	-	-	-	-	-	-
KnowCoder-**X**	**56.37**	**47.53**	**49.87**	**54.17**	**48.81**	**41.53**	24.13	**30.83**	**29.67**	**33.56**	**43.19**	**39.53**$^{\uparrow 20.03\%}$	**41.79**
Supervised (Malmasi et al., 2022a)	62.70	53.10	64.60	58.70	62.60	59.70	39.70	52.30	47.80	55.80	46.80	54.22	54.89

Table 2: Results of cross-lingual evaluation in Multiconer22. ***Avg**$_{cross}$* represents the average performance across 9 unseen languages. † indicates that the score is reported by Lai et al. (2023). ‡ indicates that the score is from our implementation. * indicates that the result of GLiNER in Chinese is under the cross-lingual setting. Supervised denotes the supervised method, which is a strong baseline since the above methods are under the zero-shot setting.

if the span and its role match a golden annotation.

KnowCoder-**X** is fine-tuned based on Baichuan2-7B-Base (Baichuan, 2023) using LLaMA-Factory (Zheng et al., 2024). We apply LoRA (Hu et al., 2022) with a LoRA rank of 32 for parameter-efficient fine-tuning. The warmup ratio is set to 0.01, and the learning rate for both phases is 3×10^{-4}. The sequence length is limited to 4096, and the batch size is 256. During inference, the temperature is set to 0. All experiments are conducted on 8 x NVIDIA A8000 80G GPUs.

5.2 Datasets

We follow the IEPILE (Gui et al., 2024), YAYI-UIE (Xiao et al., 2023), and B^{2}NER (Yang et al., 2024) to conduct a comprehensive evaluation. Overall, for English IE, we evaluate performance across 28 benchmarks for NER, 10 benchmarks for RE, 6 benchmarks for ED, and 3 benchmarks for EAE. For Chinese IE, we evaluate performance on 4 benchmarks for NER, 5 benchmarks for RE, 4 benchmarks for ED, and 2 benchmarks for EAE.

Specifically, for cross-lingual evaluation, we evaluate performance on the multilingual NER benchmark Multiconer22 (Malmasi et al., 2022b) and the low-resource African language NER benchmark MasakhaNER 2.0 (Adelani et al., 2022). Moreover, we conduct the supervised evaluation on various benchmarks: 23 NER, 8 RE, 3 ED, and 3 EAE benchmarks in English, as well as 2 NER, 2 RE, 2 ED, and 2 EAE benchmarks in Chinese. Appendix G presents the detailed statistics and usage.

We only train on the supervised evaluation IE datasets of the three main baselines to ensure a fair comparison, without additional training data.

5.3 Baselines

To enable a fair and comprehensive comparison, we mainly use two types of baselines: Following B^{2}NER (Yang et al., 2024), we first compare KnowCoder-**X** with BERT-Base (Devlin, 2018). Additionally, we compare KnowCoder-**X** with the SoTA systems for English and Chinese IE based on LLMs, including IEPILE (Gui et al., 2024), YAYI-UIE (Xiao et al., 2023), and B^{2}NER (Yang et al., 2024), which are our main baselines and based on natural language prompts. The three baselines exhibit varying degrees of unfairness compared to KnowCoder-**X**. IEPILE and YAYI-UIE are fine-tuned on the larger backbone Baichuan2-13B, while B^{2}NER utilizes the more advanced backbone Intern2LM-7B (Cai et al., 2024).

For the cross-lingual evaluation, we further compare KnowCoder-**X** with ChatGPT (from Lai et al., 2023) and GLiNER (Zaratiana et al., 2024).

6 Experimental Results

Our extensive evaluation of IE cross-lingual transfer can be divided into two main dimensions: the *cross-lingual evaluation* for external-language transfer in unseen languages (§6.1), and the *supervised evaluation* for internal-language transfer between English and Chinese (§6.2). We additionally conduct the zero-shot evaluation for internal language transfer. Appendix D shows the results.

6.1 Cross-Lingual Evaluation

To evaluate whether KnowCoder-**X** improves IE cross-lingual alignment, we conduct the cross-lingual evaluation on 9 unseen languages of Multiconer22. We have also supplemented the evaluation in Chinese and English. Table 2 shows results. KnowCoder-**X** significantly outperforms ChatGPT and SoTA in the cross-lingual setting, achieving a 30.17% and 20.03% improvement, respectively. For 4 unseen languages used in B^{2}NER that constitute more than 0.1% of the general LLM pretraining corpus (Touvron et al., 2023), KnowCoder-**X** outperforms B^{2}NER by 19.36%, with results comparable to B^{2}NER in both Chinese and English, demonstrating a significant im-

Dataset	BERT	YAYI-UIE	IEPILE	B²NER	KnowCoder-X
ACE 2005	87.30	81.78	81.86	83.04	**87.49**
AnatEM	85.82	76.54	87.21	89.18	**89.19**
BC2GM	80.90	82.05	80.73	81.95	**84.49**
BC4CHEMD	86.72	88.46	**90.45**	88.96	89.57
BC5CDR	85.28	83.67	88.07	**88.52**	88.46
Broad Twitter	58.61	**83.52**	-	82.16	82.36
CoNLL 2003	92.40	**96.77**	92.49	92.56	94.69
FabNER	64.20	72.63	77.07	78.82	**83.19**
FindVehicle	87.13	98.47	98.49	97.89	**99.47**
GENIA	73.30	75.21	76.66	76.43	**78.97**
HarveyNER	**82.26**	69.57	67.70	73.67	73.91
MIT Movie	88.78	70.14	88.23	**90.78**	89.50
MIT Restaurant	81.02	79.38	79.85	**83.71**	81.95
MultiNERD	91.25	88.42	94.60	93.98	**95.94**
NCBI	80.20	**87.29**	85.26	84.83	85.49
OntoNotes	**91.11**	87.04	87.55	84.31	87.91
PolyglotNER	**75.65**	70.85	-	61.96	64.47
TweetNER7	56.49	66.99	-	66.26	**67.98**
WikiANN	70.60	72.63	-	**85.07**	84.69
wikiNeural	82.78	87.63	-	**93.01**	87.79
Avg	80.09	80.95	-	83.85	**84.88**
MSRA	94.95	95.97	87.99	92.22	**96.01**
ResumeNER	95.93	-	93.92	95.90	**96.05**
Avg	94.72	-	90.96	94.06	**96.03**

Table 3: Results of supervised evaluation on NER. Results on Chinese benchmarks are from Li et al. (2020a).

Dataset	YAYI-UIE	IEPILE	KnowCoder-X
ADE corpus	84.14	83.73	**84.45**
CoNLL 2004	**79.73**	72.87	73.14
GIDS	72.36	74.71	**76.19**
kbp37	59.35	65.09	**72.94**
NYT	89.97	93.00	**96.08**
NYT11-HRL	**57.53**	53.19	56.79
SciERC	40.94	43.53	**44.93**
Semeval RE	61.02	58.47	**64.79**
Avg	68.13	68.07	**71.16**
CMeIE	-	49.16	**52.37**
DuIE 2.0	81.19	75.61	**82.85**
Avg	-	62.39	**67.26**

Table 4: Results of supervised evaluation on RE.

Task	Dataset	BERT	YAYI-UIE	IEPILE	KnowCoder-X
ED	ACE 2005	72.50	65.00	72.46	**73.57**
	CASIE	**68.98**	63.00	60.07	63.91
	PHEE	-	63.00	63.22	**67.03**
	Avg	-	63.67	65.25	**68.17**
	DuEE 1.0	82.18	85.00	86.73	**87.18**
	DuEE-Fin	84.53	82.50	83.54	**85.13**
	Avg	83.36	83.75	85.14	**86.16**
EAE	ACE 2005	59.90	62.71	63.90	**69.95**
	CASIE	60.37	64.23	56.07	**64.96**
	PHEE	-	**77.19**	70.85	76.24
	Avg	-	68.04	63.61	**70.38**
	DuEE 1.0	70.68	78.08	75.63	**82.12**
	DuEE-Fin	75.73	70.02	79.34	**81.09**
	Avg	73.21	74.05	77.49	**81.61**

Table 5: Results of supervised evaluation on EE.

provement in cross-lingual transfer. Surprisingly, despite no training data in 9 unseen languages, KnowCoder-**X** has achieved performance comparable to the supervised method in Spanish and Turkish that significantly outperforms the SoTA.

Appendix C presents the evaluation results on MasakhaNER 2.0 with 20 African low-resource languages. The results demonstrate that KnowCoder-**X** exhibits strong cross-lingual performance and can extend to a wide range of languages, particularly those with limited resources.

6.2 Supervised Evaluation

The results for NER, RE, EE (including ED and EAE) tasks are shown in Tables 3, 4, and 5 respectively. KnowCoder-**X** outperforms the SoTA baselines on most benchmarks for four tasks and ranks within the top-2 results across all RE and ED benchmarks. In the English IE, KnowCoder-**X** has achieved significant average improvements of 3.03 and 2.92 F1 points on the RE and ED tasks, with an improvement of 7.85 points on the kbp37 of RE. In the Chinese IE, KnowCoder-**X** has consistently achieved SoTA, with 4.12 points average improvement over the SoTA baseline on the EAE task. Moreover, it can be observed that the model not only achieves significant improvements in NER but also exhibits even greater enhancements in RE, ED, and EAE. This indicates that the alignment phase enhances the multilingual NER capability through cross-lingual alignment, which improves the performance of all tasks by leveraging the foundational spotting capabilities of NER.

To further demonstrate that KnowCoder-**X** improves multilingual IE across all languages through IE cross-lingual alignment, we also compared it comprehensively with SoTA monolingual (English) IE systems, which are fine-tuned on LLMs including InstructUIE (Wang et al., 2023), UniversalNER (Zhou et al., 2024), GoLLIE (Sainz et al., 2024), KnowCoder (Li et al., 2024b), GLiNER (Zaratiana et al., 2024), and GNER (Ding et al., 2024). Appendix E shows a detailed comparison with all baselines. Compared to 10 SoTA methods in the supervised setting, it achieves top-2 results on 14 (of 23) English NER benchmarks, indicating substantial enhancements. Specifically, when compared to other code-based IE systems, such as KnowCoder and GoLLIE, KnowCoder-**X** still achieves a notable improvement. This observation underscores the effectiveness of cross-lingual alignment of KnowCoder-**X**. This indicates that our cross-lingual alignment not only avoids conflicts between different languages but also enhances overall multilingual IE performance by leveraging cross-lingual alignment. Appendix F shows a comparison of the computational resources.

Dataset	Text-Based	Code-Based
ACE 2005 (Walker et al., 2006)	84.93	**86.07**
AnatEM (Pyysalo and Ananiadou, 2014)	89.63	**90.14**
BC2GM (Kocaman and Talby, 2020)	82.53	**83.17**
BC5CDR (Li et al., 2016)	90.03	**90.17**
CoNLL 2003 (Sang and Meulder, 2003)	94.67	**94.93**
WNUT 2017 (Derczynski et al., 2017)	**66.14**	66.03
ResumeNER (Zhang and Yang, 2018)	94.31	**95.37**
MSRA (Levow, 2006)	92.97	**94.03**
Avg	86.90	**87.49**

Table 6: Ablation study on code-based multilingual IE.

Model	**NER**	**RE**	**ED**	**EAE**	***Avg***
KnowCoder-**X** (w/o align)	85.06	68.67	74.12	74.00	75.46
KnowCoder-**X**	**85.63**	**70.38**	**75.37**	**74.87**	**76.56**

Table 7: Ablation study on IE alignment training.

6.3 Ablation Study

We conduct comprehensive ablation experiments to investigate whether the code-based multilingual IE and IE cross-lingual alignment phase of KnowCoder-**X** contribute to the performance.

Code-Based Multilingual IE Table 6 shows results. We evaluate the text-based (Wang et al., 2023) and code-based multilingual IE method in the supervised setting. We use 6 English NER datasets, ACE 2005, AnatEM, BC2GM, BC5CDR, CoNLL 2003, WNUT 2017, and 2 Chinese NER datasets, MSRA, ResumeNER to conduct instruction tuning on the Baichuan2-7B for supervised evaluation. We remove class comments in our schemas to ensure a fair comparison. The code-based method outperforms by 0.59 points, with improvements exceeding 1.00 points in 3 datasets, particularly in the Chinese datasets MSRA and ResumeNER. The results show that the code can mitigate schema differences between languages, thereby enhancing the cross-lingual alignment to improve the performance of multilingual IE.

IE Alignment Training To show whether the IE cross-lingual alignment phase contributes to the performance, we further conduct an ablation study by removing the cross-lingual alignment phase of KnowCoder-**X** and denoting it as KnowCoder-**X** `(w/o align phase)`. Table 7 shows the average results in the supervised setting on 4 tasks across 45 benchmarks. Compared to KnowCoder-**X** `(w/o align phase)`, the results indicate that KnowCoder-**X** achieves a substantial average improvement of 1.1 points across four tasks. This suggests that the IE cross-lingual alignment phase enhances the overall performance of multilingual IE, demonstrating its contribution to facilitating cross-lingual transfer within the multilingual IE framework.

Language	Text-Based	Code-Based	Code-Based (w/ com.)
German	35.16	38.69	**48.02**
Spanish	43.19	48.93	**52.16**
Dutch	39.16	42.19	**46.79**
Russian	33.12	36.19	**39.68**
Bengali	18.23	17.59	**23.95**
Persian	23.45	24.69	**28.54**
Hindi	21.71	23.91	**28.13**
Korean	28.31	28.65	**31.42**
Turkish	32.36	35.16	**41.16**
Avg	30.52	32.89	**37.76**

Table 8: Analysis on cross-lingual transfer.

6.4 Analysis on Cross-Lingual Transfer

To further investigate the impact of code-based unified schema representation in cross-lingual IE, we conduct cross-lingual evaluation finetuned on all NER datasets under three settings: text-based, code-based, and code-based (with comments). Table 8 shows the results. It indicates that adopting the code-based multilingual IE method can enhance cross-lingual performance by 2.37%. This improvement can be attributed to the standardized representation of the same ontology across different languages provided by the code-based multilingual IE method. Additionally, the integration of guideline information further improves cross-lingual performance by 4.85%, particularly in low-resource languages such as Bengali and Persian. The code-based multilingual IE effectively facilitates the integration of guideline information, significantly enhancing cross-lingual IE performance.

7 Conclusions

We introduced KnowCoder-**X**, a code-based multilingual IE model that significantly enhances cross-lingual transfer through unified schema representation and IE cross-lingual alignment training. KnowCoder-**X** utilizes Python classes to represent schemas uniformly across languages, ensuring semantic consistency in multilingual IE. Additionally, the IE cross-lingual alignment phase leverages high-quality IE parallel datasets ParallelNER, which are generated by our proposed LLM-based pipeline, to boost multilingual generalization. Comprehensive experiments demonstrate that KnowCoder-**X** achieves SoTA in Chinese and English IE under various settings, with remarkable cross-lingual capability.

Limitation

Expanding training to include languages beyond English and Chinese remains an area for further exploration. Moreover, incorporating additional languages would provide valuable insights into the cross-lingual alignment mechanisms. Additionally, addressing domain specificity will be a key focus of future research. Specialized vocabulary and domain-specific language use present significant challenges for cross-lingual transfer, making it essential to overcome these obstacles in upcoming work. Introducing parallel data from other IE tasks during the cross-lingual alignment phase also offers promising avenues for further investigation.

Acknowledgments

This work is partially funded by the National Natural Science Foundation of China under grants 62306299 and 62441229, the Lenovo-CAS Joint Lab Youth Scientist Project, the project under Grants No. JCKY2022130C039, and the Strategic Priority Research Program of the CAS under Grants No. XDB0680102. We thank anonymous reviewers for their insightful comments and suggestions.

References

David Ifeoluwa Adelani, Graham Neubig, Sebastian Ruder, Shruti Rijhwani, Michael Beukman, Chester Palen-Michel, Constantine Lignos, Jesujoba O Alabi, Shamsuddeen H Muhammad, Peter Nabende, et al. 2022. Masakhaner 2.0: Africa-centric transfer learning for named entity recognition. *arXiv preprint arXiv:2210.12391*.

Rami Al-Rfou, Vivek Kulkarni, Bryan Perozzi, and Steven Skiena. 2014. POLYGLOT-NER: massive multilingual named entity recognition. *CoRR*, abs/1410.3791.

Baichuan. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. 2024. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*.

Chih-Yao Chen and Cheng-Te Li. 2021. ZS-BERT: towards zero-shot relation extraction with attribute representation learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3470–3479. Association for Computational Linguistics.

Pei Chen, Haotian Xu, Cheng Zhang, and Ruihong Huang. 2022. Crossroads, buildings and neighborhoods: A dataset for fine-grained location recognition. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 3329–3339. Association for Computational Linguistics.

Xiangrong Chen and Alan L Yuille. 2004. Detecting and reading text in natural scenes. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 2, pages II–II. IEEE.

Yang Chen, Chao Jiang, Alan Ritter, and Wei Xu. 2023. Frustratingly easy label projection for cross-lingual transfer. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5775–5796, Toronto, Canada. Association for Computational Linguistics.

Leon Derczynski, Kalina Bontcheva, and Ian Roberts. 2016. Broad Twitter corpus: A diverse named entity recognition resource. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1169–1179, Osaka, Japan. The COLING 2016 Organizing Committee.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yuyang Ding, Juntao Li, Pinzheng Wang, Zecheng Tang, Bowen Yan, and Min Zhang. 2024. Rethinking negative instances for generative named entity recognition. *Preprint*, arXiv:2402.16602.

Rezarta Islamaj Dogan, Robert Leaman, and Zhiyong Lu. 2014. NCBI disease corpus: A resource for disease name recognition and concept normalization. *J. Biomed. Informatics*, 47:1–10.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European*

Chapter of the Association for Computational Linguistics: Main Volume, pages 2112–2128, Online. Association for Computational Linguistics.

Seth Ebner, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. 2020. Multi-sentence argument linking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Changjiang Gao, Hongda Hu, Peng Hu, Jiajun Chen, Jixing Li, and Shujian Huang. 2024. Multilingual pretraining and instruction tuning improve cross-lingual knowledge alignment, but only shallowly. *arXiv preprint arXiv:2404.04659*.

Runwei Guan, Ka Lok Man, Feifan Chen, Shanliang Yao, Rongsheng Hu, Xiaohui Zhu, Jeremy Smith, Eng Gee Lim, and Yutao Yue. 2024. Findvehicle and vehiclefinder: a ner dataset for natural language-based vehicle retrieval and a keyword-based cross-modal vehicle retrieval system. *Multimedia Tools and Applications*, 83(8):24841–24874.

Tongfeng Guan, Hongying Zan, Xiabing Zhou, Hongfei Xu, and Kunli Zhang. 2020. Cmeie: Construction and evaluation of chinese medical information extraction dataset. In *Natural Language Processing and Chinese Computing - 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14-18, 2020, Proceedings, Part I*, volume 12430 of *Lecture Notes in Computer Science*, pages 270–282. Springer.

Honghao Gui, Lin Yuan, Hongbin Ye, Ningyu Zhang, Mengshu Sun, Lei Liang, and Huajun Chen. 2024. IEPile: Unearthing large scale schema-conditioned information extraction corpus. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 127–146, Bangkok, Thailand. Association for Computational Linguistics.

Yucan Guo, Zixuan Li, Xiaolong Jin, Yantao Liu, Yutao Zeng, Wenxuan Liu, Xiang Li, Pan Yang, Long Bai, Jiafeng Guo, et al. 2023. Retrieval-augmented code generation for universal information extraction. *arXiv preprint arXiv:2311.02962*.

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, and Luca Toldo. 2012. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports. *Journal of Biomedical Informatics*, 45(5):885–892. Text Mining and Natural Language Processing in Pharmacogenomics.

Cuiyun Han, Jinchuan Zhang, Xinyu Li, Guojin Xu, Weihua Peng, and Zengfeng Zeng. 2022. Duee-fin: A large-scale dataset for document-level event extraction. In *Natural Language Processing and Chinese Computing - 11th CCF International Conference, NLPCC 2022, Guilin, China, September 24-25, 2022, Proceedings, Part I*, volume 13551 of *Lecture Notes in Computer Science*, pages 172–183. Springer.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4803–4809. 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.

Leonhard Hennig, Philippe Thomas, and Sebastian Möller. 2023. Multitacred: A multilingual version of the tac relation extraction dataset. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3785–3801.

Eduard H. Hovy, Mitchell P. Marcus, Martha Palmer, Lance A. Ramshaw, and Ralph M. Weischedel. 2006. Ontonotes: The 90% solution. In *North American Chapter of the Association for Computational Linguistics*.

Chiaming Hsu, Changtong Zan, Liang Ding, Longyue Wang, Xiaoting Wang, Weifeng Liu, Fu Lin, and Wenbin Hu. 2023. Prompt-learning for cross-lingual relation extraction. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–9. IEEE.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Kuan-Hao Huang, I Hsu, Premkumar Natarajan, Kai-Wei Chang, Nanyun Peng, et al. 2022. Multilingual generative language models for zero-shot cross-lingual event argument extraction. *arXiv preprint arXiv:2203.08308*.

Lifu Huang, Heng Ji, Kyunghyun Cho, and Clare R Voss. 2017. Zero-shot transfer learning for event extraction. *arXiv preprint arXiv:1707.01066*.

Sharmistha Jat, Siddhesh Khandelwal, and Partha Talukdar. 2018. Improving distantly supervised relation extraction using word and entity based attention. *Preprint*, arXiv:1804.06987.

Jin-Dong Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. 2003. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics (Oxford, England)*, 19 Suppl 1:i180–2.

Veysel Kocaman and David Talby. 2020. Biomedical named entity recognition at scale. In *Pattern Recognition. ICPR International Workshops and Challenges - Virtual Event, January 10-15, 2021, Proceedings, Part I*, volume 12661 of *Lecture Notes in Computer Science*, pages 635–646. Springer.

Keshav Kolluru, Muqeeth Mohammed, Shubham Mittal, Soumen Chakrabarti, et al. 2022. Alignment-augmented consistent translation for multilingual open information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2502–2517.

Saul Kripke and Milton K Munitz. 1971. Identity and necessity. *1971*, pages 135–164.

Aman Kumar and Binil Starly. 2021. "fabner": information extraction from manufacturing process science domain literature using named entity recognition. *Journal of Intelligent Manufacturing*, 33:2393 – 2407.

Viet Dac Lai, Nghia Trung Ngo, Amir Pouran Ben Veyseh, Hieu Man, Franck Dernoncourt, Trung Bui, and Thien Huu Nguyen. 2023. Chatgpt beyond english: Towards a comprehensive evaluation of large language models in multilingual learning. *ArXiv*, abs/2304.05613.

Meisin Lee, Lay-Ki Soon, Eu-Gene Siew, and Ly Fie Sugianto. 2021. An annotated commodity news corpus for event extraction. *Preprint*, arXiv:2105.08214.

Gina-Anne Levow. 2006. The third international chinese language processing bakeoff: Word segmentation and named entity recognition. In *Proceedings of the Fifth Workshop on Chinese Language Processing, SIGHAN@COLING/ACL 2006, Sydney, Australia, July 22-23, 2006*, pages 108–117. Association for Computational Linguistics.

Jiahuan Li, Shujian Huang, Aarron Ching, Xinyu Dai, and Jiajun Chen. 2024a. Prealign: Boosting cross-lingual transfer by early establishment of multilingual alignment. *arXiv preprint arXiv:2407.16222*.

Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. Biocreative V CDR task corpus: a resource for chemical disease relation extraction. *Database J. Biol. Databases Curation*, 2016.

Peng Li, Tianxiang Sun, Qiong Tang, Hang Yan, Yuanbin Wu, Xuanjing Huang, and Xipeng Qiu. 2023. Codeie: Large code generation models are better few-shot information extractors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 15339–15353. Association for Computational Linguistics.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 894–908. Association for Computational Linguistics.

Xiaonan Li, Hang Yan, Xipeng Qiu, and Xuanjing Huang. 2020a. FLAT: Chinese NER using flat-lattice transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6836–6842, Online. Association for Computational Linguistics.

Xinyu Li, Fayuan Li, Lu Pan, Yuguang Chen, Weihua Peng, Quan Wang, Yajuan Lyu, and Yong Zhu. 2020b. Duee: A large-scale dataset for chinese event extraction in real-world scenarios. In *Natural Language Processing and Chinese Computing - 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14-18, 2020, Proceedings, Part II*, volume 12431 of *Lecture Notes in Computer Science*, pages 534–545. Springer.

Zixuan Li, Yutao Zeng, Yuxin Zuo, Weicheng Ren, Wenxuan Liu, Miao Su, Yucan Guo, Yantao Liu, Xiang Li, Zhilei Hu, et al. 2024b. Knowcoder: Coding structured knowledge into llms for universal information extraction. *arXiv preprint arXiv:2403.07969*.

Wenxuan Liu, Zixuan Li, Long Bai, Yuxin Zuo, Daozhu Xu, Xiaolong Jin, Jiafeng Guo, and Xueqi Cheng. 2025. Towards event extraction with massive types: Llm-based collaborative annotation and partitioning extraction. *arXiv preprint arXiv:2503.02628*.

Yijin Liu, Fandong Meng, Jinchao Zhang, Jinan Xu, Yufeng Chen, and Jie Zhou. 2019. GCDT: A global context enhanced deep transition architecture for sequence labeling. *CoRR*, abs/1906.02437.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. Crossner: Evaluating cross-domain named entity recognition. 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 13452–13460. AAAI Press.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. Association for Computational Linguistics.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018a. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. *Preprint*, arXiv:1808.09602.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018b. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3219–3232. Association for Computational Linguistics.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022a. Multiconer: A large-scale multilingual dataset for complex named entity recognition. In *International Conference on Computational Linguistics*.

Shervin Malmasi, Anjie Fang, Besnik Fetahu, Sudipta Kar, and Oleg Rokhlenko. 2022b. MultiCoNER: A large-scale multilingual dataset for complex named entity recognition. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3798–3809, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Tanmay Parekh, I-Hung Hsu, Kuan-Hao Huang, Kai-Wei Chang, and Nanyun Peng. 2024. Contextual label projection for cross-lingual structured prediction. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5738–5757, Mexico City, Mexico. Association for Computational Linguistics.

Nanyun Peng and Mark Dredze. 2015. Named entity recognition for chinese social media with jointly trained embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 548–554. The Association for Computational Linguistics.

Sameer S. Pradhan and Nianwen Xue. 2009. Ontonotes: The 90% solution. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, May 31 - June 5, 2009, Boulder, Colorado, USA, Tutorial Abstracts*, pages 11–12. The Association for Computational Linguistics.

Sampo Pyysalo and Sophia Ananiadou. 2014. Anatomical entity mention recognition at literature scale. *Bioinform.*, 30(6):868–875.

Jirui Qi, Raquel Fernández, and Arianna Bisazza. 2023. Cross-lingual consistency of factual knowledge in multilingual language models. pages 10650–10666, Singapore.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *ECML/PKDD*.

Alan Rios, Bruno Cabral, Daniela Claro, Rerisson Cavalcante, and Marlo Souza. 2024. Transalign: An automated corpus generation through cross-linguistic data alignment for open information extraction. In *Proceedings of the 16th International Conference on Computational Processing of Portuguese*, pages 196–206.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, pages 1–8, Boston, Massachusetts, USA. Association for Computational Linguistics.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2024. GoLLIE: Annotation guidelines improve zero-shot information-extraction. In *The Twelfth International Conference on Learning Representations*.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147. ACL.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. CASIE: extracting cybersecurity event information from text. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8749–8757. AAAI Press.

Zhaoyue Sun, Jiazheng Li, Gabriele Pergola, Byron C. Wallace, Bino John, Nigel Greene, Joseph Kim, and Yulan He. 2022. PHEE: A dataset for pharmacovigilance event extraction from text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 5571–5587. Association for Computational Linguistics.

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu, and Minlie Huang. 2018. A hierarchical framework for relation extraction with reinforcement learning. In *AAAI Conference on Artificial Intelligence*.

Simone Tedeschi, Valentino Maiorca, Niccolò Campolungo, Francesco Cecconi, and Roberto Navigli. 2021. Wikineural: Combined neural and knowledge-based silver data creation for multilingual ner. In *Findings of the association for computational linguistics: EMNLP 2021*, pages 2521–2533.

Simone Tedeschi and Roberto Navigli. 2022. MultiNERD: A multilingual, multi-genre and fine-grained dataset for named entity recognition (and disambiguation). In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 801–812, Seattle, United States. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Asahi Ushio, Leonardo Neves, Vitor Silva, Francesco Barbieri, and Jose Camacho-Collados. 2022. Named entity recognition in twitter: A dataset and analysis on short-term temporal shifts. *Preprint*, arXiv:2210.03797.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus.

Bin Wang, Zhengyuan Liu, Xin Huang, Fangkai Jiao, Yang Ding, AiTi Aw, and Nancy F. Chen. 2024. Seaeval for multilingual foundation models: From cross-lingual alignment to cultural reasoning. *Preprint*, arXiv:2309.04766.

Haitao Wang, Zhengqiu He, Jin Ma, Wenliang Chen, and Min Zhang. 2019. Ipre: a dataset for inter-personal relationship extraction. In *Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, October 9–14, 2019, Proceedings, Part II 8*, pages 103–115. Springer.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. 2023. Instructuie: Multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*.

Xingyao Wang, Sha Li, and Heng Ji. 2022. Code4struct: Code generation for few-shot event structure prediction. *arXiv preprint arXiv:2210.12810*.

Xinglin Xiao, Yijie Wang, Nan Xu, Yuqi Wang, Hanxuan Yang, Minzheng Wang, Yin Luo, Lei Wang, Wenji Mao, and Daniel Zeng. 2023. Yayi-uie: A chat-enhanced instruction tuning framework for universal information extraction. *arXiv preprint arXiv:2312.15548*.

Liang Xu, Qianqian Dong, Cong Yu, Yin Tian, Weitang Liu, Lu Li, and Xuanwei Zhang. 2020. Cluener2020: Fine-grained name entity recognition for chinese. *arXiv preprint arXiv:2001.04351*.

Yuming Yang, Wantong Zhao, Caishuang Huang, Junjie Ye, Xiao Wang, Huiyuan Zheng, Yang Nan, Yuran Wang, Xueying Xu, Kaixin Huang, et al. 2024. Beyond boundaries: Learning a universal entity taxonomy across datasets and languages for open named entity recognition. *arXiv preprint arXiv:2406.11192*.

Urchade Zaratiana, Nadi Tomeh, Pierre Holat, and Thierry Charnois. 2024. GLiNER: Generalist model for named entity recognition using bidirectional transformer. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5364–5376, Mexico City, Mexico. Association for Computational Linguistics.

Renzo M. Rivera Zavala, Paloma Martínez, and Isabel Segura-Bedmar. 2018. A hybrid bi-lstm-crf model to recognition of disabilities from biomedical texts. In *IberEval@SEPLN*.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *Preprint*, arXiv:1508.01006.

Yue Zhang and Jie Yang. 2018. Chinese NER using lattice LSTM. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1554–1564, Melbourne, Australia. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Yu Gu, Muhao Chen, and Hoifung Poon. 2024. Universalner: Targeted distillation from large language models for open named entity recognition. In *The Twelfth International Conference on Learning Representations*.

Yang Zhou, Yubo Chen, Jun Zhao, Yin Wu, Jiexin Xu, and Jinlong Li. 2021. What the role is vs. what plays the role: Semi-supervised event argument extraction via dual question answering. 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 14638–14646. AAAI Press.

Wenhao Zhu, Yunzhe Lv, Qingxiu Dong, Fei Yuan, Jingjing Xu, Shujian Huang, Lingpeng Kong, Jiajun Chen, and Lei Li. 2023. Extrapolating large language models to non-english by aligning languages. *arXiv preprint arXiv:2308.04948*.

Mikel Zubillaga, Oscar Sainz, Ainara Estarrona, Oier Lopez de Lacalle, and Eneko Agirre. 2024. Event extraction in basque: Typologically motivated cross-lingual transfer-learning analysis. *arXiv preprint arXiv:2404.06392*.

A Description Generation

We provide a detailed explanation of the description generation process using the NER task as an example. The generation process consists of two main phases, i.e., *Description Initialization* and *Description Polish*:

- *Description Initialization:* In this phase, we sample some entities to instruct the LLMs to summarize an initial description of these entities. The details are as follows: 1) Randomly sample 10 entities from the training set. 2) Format the 10 entities into the description initialization prompt template as shown in Tables 20, instructing the LLM to summarize and generate an initial description.

- *Description Polish:* Based on the description generated by the first phase, this phase further instructs the LLM to polish the description based on a set of entities iteratively.

Description Initialization often struggles to accurately represent and capture the characteristics of each entity. To address this, we introduce *Description Polish*, a method designed to refine the details of the description, thereby enabling it to summarize the entities within a given entity type more effectively.

B Pipeline Evaluation

We evaluate our pipeline on the label projection benchmark WikiANN (Pan et al., 2017), which is a multilingual NER dataset. We use 10 languages, including Bengali (bn), German (de), Spanish (es), Persian (fa), Hindi (hi), Korean (ko), Dutch (nl), Russian (ru), Turkish (tr), and Chinese (zh), and randomly sample $1,000$ samples for each language. We compare our pipeline with CLaP (Parekh et al., 2024), Awesome-Align (Dou and Neubig, 2021), and EasyProject (Chen et al., 2023). We mainly evaluate faithfulness following the setting of CLaP, measured as the percentage of instances where translated labels appear in the translated sentence, as accuracy evaluation requires costly native speaker rankings across methods, incurring high costs.

Table 9 shows the result. We achieve 99% in faithfulness on average, outperforming the current SoTA Awesome-Align by 3% points on average. We attribute the achievement to the latter two stages, which rephrase the span with minimal disturbance to the sentence. Besides, other baselines experience significant performance drops in certain languages, such as EasyProject in Russia. In contrast, our pipeline achieves scores exceeding 93% across all languages, which strongly demonstrates the robustness and reliability of our pipeline. The results further substantiate the stability of our pipeline, demonstrating its capability to adapt effectively across different linguistic contexts.

Language	Awesome-Align	EasyProject	CLaP	KnowCoder-**X**
bn	92	98	93	**99**
de	99	97	79	**99**
es	99	99	84	**100**
fa	96	99	72	**100**
hi	93	36	90	**99**
ko	96	93	64	**99**
nl	99	100	85	**100**
ru	97	**99**	66	93
tr	98	98	94	**99**
zh	92	92	60	**99**
Avg	96	91	79	**99**

Table 9: Results of label projection evaluation.

C Cross-Lingual Evaluation on Low Resource Languages

Table 10 shows results on the MasakhaNER 2.0. KnowCoder-**X** achieves the best performance across a wider variety of low-resource languages. Compared to IEPILE and GLiNER, KnowCoder-**X** achieves improvements of 11.41% and 27.11%, respectively. These findings highlight the remarkable cross-lingual performance of KnowCoder-**X** on low-resource languages. The results demonstrate that KnowCoder-**X** exhibits excellent cross-lingual generalization for IE, which can be attributed to the enhancement of IE cross-lingual alignment.

D Zero-Shot Evaluation

Datasets For the zero-shot evaluation, among NER benchmarks, we use 5 English benchmarks from CrossNER (Liu et al., 2021) and 2 Chinese benchmarks, Weibo (Peng and Dredze, 2015) and Boson[1]. Among RE benchmarks, we adopt 2 English benchmarks FewRel (Han et al., 2018), Wiki-ZSL (Chen and Li, 2021) and 3 Chinese benchmarks, COAE2016[2], IPRE (Wang et al., 2019), and SKE2020[3]. Among ED benchmarks, we use 3 English benchmarks CrudeOil News (Lee et al., 2021),

[1] https://github.com/InsaneLife/
[2] https://github.com/Sewens/COAE2016
[3] https://aistudio.baidu.com/datasetdetail/177191

Language	YAYI-UIE	IEPILE	GLiNER	KnowCoder-X
Bamanankan	31.19	29.00	28.85	**32.97**
Ghomala	32.03	32.36	**34.71**	32.97
Éwé	63.01	57.83	64.64	**66.05**
Fon	30.05	29.73	30.58	**31.86**
Hausa	47.68	43.06	42.04	**48.35**
Igbo	37.14	37.54	36.08	**38.75**
Kinyarwanda	40.84	43.87	45.04	**47.56**
Luganda	46.74	53.13	48.59	**58.04**
Luo	44.04	48.02	39.04	**49.89**
Mossi	26.05	26.42	30.96	**31.71**
Chichewa	**53.72**	52.93	48.52	51.94
Nigerian-Pidgin	66.71	67.50	39.96	**81.91**
Shona	43.47	43.87	34.37	**46.58**
Swahili	64.62	55.50	58.06	**69.42**
Setswana	51.45	**54.05**	39.83	**56.97**
Twi	35.62	45.69	28.54	**51.10**
Wolof	34.75	35.31	36.27	**38.70**
Xhosa	36.57	36.99	29.25	**45.73**
Yoruba	28.88	26.79	5.06	**34.73**
Zulu	**38.07**	34.61	28.34	36.50
Avg	42.63	42.71	37.44	**47.59**$^{\uparrow 11.43\%}$

Table 10: Results of cross-lingual evaluation on MasakhaNER 2.0.

Task	Dataset	GPT-4	YAYI-UIE	IEPILE	KnowCoder-X
NER	AI	-	52.40	-	**65.04**
	Literature	-	45.99	-	**59.68**
	Music	-	51.20	-	**62.97**
	Politics	-	51.82	-	**63.65**
	Science	-	50.53	-	**62.29**
	Avg	58.49	50.39	55.55	**62.73**
RE	FewRel	22.43	36.09	41.28	**42.18**
	Wiki-ZSL	23.76	41.07	37.61	**42.13**
ED	CrudeOil News	26.13	12.45	**36.61**	33.47
	WikiEvents	5.24	10.97	9.12	**12.59**
	RAMS	10.14	18.87	20.19	**20.71**
Avg	-	38.02	37.14	42.26	**46.47**

Table 11: Results of zero-shot evaluation on English Benchmarks. Results of GPT-4 are from Gui et al. (2024)

Task	Dataset	GPT-4	YAYI-UIE	IEPILE	KnowCoder-X
NER	Boson	48.15	49.25	55.77	**59.58**
	Weibo	29.80	38.03	36.46	**44.01**†
RE	COAE2016	41.15	19.97	47.43	**48.79**
	IPRE	18.15	22.97	29.76	**32.43**
	SKE2020	56.77	70.80	72.50	**72.91**
ED	CCF Law	42.12	12.87	63.53	**68.90**
	FewEC	74.25	81.28	83.59	**84.87**
Avg	-	44.34	42.17	55.58	**58.78**

Table 12: Results of zero-shot evaluation on Chinese Benchmarks. † donates the result of Weibo using four types following the setups of the two baseline models, and the result for the eight types is 38.80.

RAMS (Ebner et al., 2020), WikiEvents (Li et al., 2021) and 2 Chinese benchmarks FewFC (Zhou et al., 2021) and CCF law[4].

Results Tables 11 and 12 show the zero-shot performance in English and Chinese, respectively. KnowCoder-**X** achieved SoTA performance across all benchmarks in Chinese tasks, demonstrating substantial and consistent improvements in generalization ability. In English tasks, KnowCoder-**X** exhibited a noteworthy average increase of 4.24 points on the NER task. Similarly, in Chinese tasks, KnowCoder-**X** surpassed existing SoTA models on all benchmarks, achieving an average improvement of 3.29 points. Moreover, the trends demonstrated in both Chinese and English show that the most significant improvement occurs in NER. We hypothesize that the NER alignment training in the first phase is more challenging to generalize to other IE tasks in out-of-domain scenarios, which leaves the introduction of RE and EE alignment training to future work.

E Full Comparison of Supervised Evaluation

We conduct a comprehensive supervised comparison in English between KnowCoder-**X** and all current SoTA methods, including monolingual IE methods and multilingual IE methods under the supervised setting in Tables 13, 14, and 15. In the NER task, we demonstrate top-2 results across 14 (of 23) benchmarks; in the RE task, we achieve top-2 results across 5 (of 8) benchmarks. Moreover, particularly in the ED and EAE tasks, we rank among the top-2 results across all benchmarks, further substantiating the significant efficacy of our method in multilingual IE.

Additionally, in the comparison of three benchmarks DIANN (Zavala et al., 2018), Ontonotes 5 (Hovy et al., 2006), and WNUT 2017 (Derczynski et al., 2017), utilized by other English code-based IE work (Sainz et al., 2024; Li et al., 2024b), KnowCoder-**X** outperforms the other baselines, which further demonstrated that cross-lingual alignment can enhance monolingual IE.

F Computational Resource Comparison

In the Universal IE Training Phase, we only train on datasets of the supervised setting, which is the same as the other baselines and necessary for fair comparison. Therefore, the additional computational resources of KnowCoder-**X** are mainly from the IE Cross-Lingual Alignment Phase. First, this is one of our core contributions, which necessitates additional computational resources. Second, this phase of training is highly efficient. In this phase, we train the model for one epoch with 257k samples,

[4] https://aistudio.baidu.com/projectdetail/4201483

Dataset	BERT	InsturctUIE	IEPILE	YAYI-UIE	GoLLIE	UniversalNER	GLiNER	KnowCoder	GNER	B²NER	KnowCoder-X
ACE 2005	87.30	79.94	81.86	81.78	**89.10**	86.69	82.80	86.10	-	83.04	87.49
AnatEM	85.82	85.82	87.21	76.54	-	88.65	88.90	86.40	**90.24**	89.18	89.19
BC2GM	80.90	80.69	80.73	82.05	-	82.42	83.70	82.00	83.18	81.95	**84.49**
BC4CHEMD	86.72	87.62	**90.45**	88.46	-	89.21	87.90	-	89.40	88.96	89.57
BC5CDR	85.28	89.02	88.07	83.67	**91.90**	89.34	88.70	89.30	90.27	88.52	88.46
Broad Twitter	58.61	80.27	83.52	-	-	81.25	82.50	78.30	**83.74**	82.16	82.36
CoNLL 2003	92.40	91.53	92.49	**96.77**	92.90	93.30	92.60	94.10	93.60	92.56	94.69
FabNER	64.20	78.38	77.07	72.63	-	81.87	77.80	82.90	**85.39**	78.82	83.19
FindVehicle	87.13	87.56	98.49	98.47	-	98.30	95.70	99.40	98.62	97.89	**99.47**
GENIA	73.30	75.71	76.66	75.21	-	77.54	78.90	76.70	-	76.43	**78.97**
HarveyNER	**80.20**	74.49	67.70	69.57	-	74.21	-	-	74.73	73.67	73.91
MIT Movie	88.78	89.58	88.23	70.14	-	90.17	87.90	89.70	90.23	**90.78**	90.24
MIT Restaurant	81.02	82.59	79.85	79.38	-	82.35	83.60	81.30	81.73	**83.71**	81.95
MultiNERD	91.25	90.26	94.60	88.42	-	93.73	93.80	**96.10**	94.30	93.98	95.94
NCBI	80.20	86.21	85.26	87.29	-	86.96	87.80	83.80	**89.27**	84.83	85.49
OntoNotes	**91.11**	88.64	87.04	87.55	-	89.91	-	-	90.69	84.31	87.91
Polyglot	**75.65**	53.30	70.85	-	-	65.70	61.50	-	67.52	61.96	64.47
TweetNER7	56.49	65.90	66.99	-	-	65.80	51.40	-	66.87	66.26	**67.98**
WikiANN	70.60	64.47	72.63	-	-	84.91	83.70	**87.70**	86.87	85.07	84.69
WikiNeural	82.78	88.27	87.63	-	-	93.28	91.30	-	**93.71**	93.01	87.79
OntoNotes 5	84.28	-	-	-	83.40	-	-	88.20	-	-	**88.35**
DIANN	82.05	-	-	-	84.10	-	-	94.70	-	-	**94.71**
WNUT 2017	36.16	-	-	-	53.70	-	-	66.40	-	-	**68.72**

Table 13: Full comparison of supervised evaluation on NER. The results of BERT on DIANN, OntoNotes 5, and WNUT 2017 are from our implementation.

Dataset	YAYI-UIE	IEPILE	InsturctUIE	KnowCoder	KnowCoder-X
ADE corpus	84.14	83.73	82.31	84.30	**84.45**
CoNLL 2004	**79.73**	72.87	78.48	73.30	73.14
GIDS	72.36	74.71	76.90	**78.00**	76.19
kbp37	59.35	65.09	36.14	**73.20**	72.94
NYT	89.97	93.00	91.00	93.70	**96.08**
NYT11-HRL	**57.53**	53.19	56.06	-	56.79
SciERC	40.94	43.53	37.40	40.00	**44.93**
Semeval RE	61.02	58.47	**73.23**	66.30	64.79

Table 14: Full comparison of supervised evaluation on RE.

Task	Dataset	BERT	YAYI-UIE	IEPILE	KnowCoder	KnowCoder-X
ED	ACE 2005	72.50	65.00	72.46	**74.2**	73.57
	CASIE	**68.98**	63.00	60.07	-	63.91
	PHEE	-	63.00	63.22	-	**67.03**
EAE	ACE 2005	59.90	62.71	63.90	**70.30**	69.95
	CASIE	60.37	64.23	56.07	-	**64.96**
	PHEE	-	**77.19**	70.85	-	76.24

Table 15: Full comparison of supervised evaluation on ED and EAE.

which is considered a reasonable computational cost. For comparison, the additional computational resources required for the Schema Understanding Training of KnowCoder is one epoch with 926k samples (∼4x). Meanwhile, KnowCoder even has a large-scale pre-training phase.

G Data Statistics

G.1 ParallelNER

In this section, we mainly introduce the statistics of ParallelNER, which is constructed from two datasets WikiNeural (Tedeschi et al., 2021) and CLUENER2020 (Xu et al., 2020). The detailed statistics are shown in Table 16.

G.2 Dataset Statistics

In this work, we conduct evaluations on 64 datasets, comprising 34 datasets for the NER task, 15 datasets for the RE task, 10 datasets for the ED task, and 5 datasets for the EAE task. To fairly compare with code-based methods, KnowCoder and GoLLIE in demonstrating the effectiveness of our cross-lingual alignment, we also used PileNER (Zhou et al., 2024) for training. The detailed statistics are shown in Tables 17, 18, and 19, respectively.

H Examples

We present the examples of instruction-tuning data for the IE cross-lingual alignment phase and the multi-lingual IE training phase in Figure 2 and Figure 3, respectively.

I Prompts

IE Parallel Data Construction Pipeline Using en⇒zh as an example, we introduce the prompts we used in the IE parallel data construction pipeline including three stages *Joint Translation*, *Span Rephrase*, and *Sentence Rephrase* in Tables 22, 23, and 24.

#Source	#Origin Language	#Types	#Instances	#Train Num	#Valid Num	#Test Num	#Total Num
WikiNeural	En	3	149,011	92,720	11,590	11,597	115,907
CLUENER2020	Zh	10	19,720	10,000	1,343	1,345	12,688

Table 16: Statistics of ParallelNER datasets.

#Dataset	#Types	#Major Domain	#Train	#Test	#Train Num	#Valid Num	#Test Num
ACE2005 (Walker et al., 2006)	7	News	✓	✓	7,299	964	1,060
AnatEM (Pyysalo and Ananiadou, 2014)	1	Biomedical	✓	✓	5,861	2,081	3,830
BC2GM (Kocaman and Talby, 2020)	1	Biomedical	✓	✓	12,500	2,500	5,000
BC4CHEMD (Kocaman and Talby, 2020)	1	Biomedical	✓	✓	30,488	30,468	26,204
BC5CDR (Li et al., 2016)	2	Biomedical	✓	✓	4,560	4,581	4,797
Broad Twitter (Derczynski et al., 2016)	3	Social Media	✓	✓	5334	2,000	2,001
CoNLL2003 (Sang and Meulder, 2003)	3	News	✓	✓	14,041	3,250	3,453
FabNER (Kumar and Starly, 2021)	12	Science	✓	✓	9,435	2,182	2,064
FindVehicle (Guan et al., 2024)	21	Traffic	✓	✓	21,547	20,777	20,769
GENIA (Kim et al., 2003)	5	Biomedical	✓	✓	15,023	1,669	1,854
HarveyNER (Chen et al., 2022)	4	Social Media	✓	✓	3,553	1,270	1,260
MIT Movie (Liu et al., 2019)	12	Social Media	✓	✓	9,774	2,442	2,442
MIT Restaurant (Liu et al., 2019)	8	Social Media	✓	✓	7,659	1,520	1,520
MultiNERD (Tedeschi and Navigli, 2022)	16	Wikipedia	✓	✓	134,144	10,000	10,000
NCBI (Dogan et al., 2014)	1	Biomedical	✓	✓	5,432	923	940
OntoNotes (Pradhan and Xue, 2009)	18	General	✓	✓	107,032	14,110	10,838
PolygtNER (Al-Rfou et al., 2014)	3	Wikipedia	✓	✓	393,941	-	10,000
TweetNER7 (Ushio et al., 2022)	7	Social Media	✓	✓	1,325	7,111	576
WikiANN (Pan et al., 2017)	3	Wikipedia	✓	✓	20,000	-	10,000
WikiNeural (Tedeschi et al., 2021)	3	Wikipedia	✓	✓	92,720	11,590	11,597
DIANN (Zavala et al., 2018)	1	Science	✓	✓	3,900	975	1,334
OntoNotes5 (Hovy et al., 2006)	18	General	✓	✓	54994	7997	7782
WNUT 2017 (Derczynski et al., 2017)	6	General	✓	✓	3,394	1,008	1,287
PileNER (Zhou et al., 2024)	15,578	General	✓		45,883	-	-
CrossNER_AI (Liu et al., 2021)	13	AI		✓	100	350	431
CrossNER_literature (Liu et al., 2021)	11	Literary		✓	100	400	416
CrossNER_music (Liu et al., 2021)	12	Musical		✓	100	380	465
CrossNER_politics (Liu et al., 2021)	8	Political		✓	199	540	650
CrossNER_science (Liu et al., 2021)	16	Scientific		✓	200	450	543
MSRA (Levow, 2006)	3	News	✓	✓	46,364	4,500	3,442
ResumeNER (Zhang and Yang, 2018)	8	News	✓	✓	3,821	-	477
Weibo (Peng and Dredze, 2015)	8	Social media		✓	1,350	-	270
Boson[5]	6	News		✓	-	-	191
Multiconer(Malmasi et al., 2022b)	6	News		✓	-	-	51,793
MasakhaNER 2.0(Adelani et al., 2022)	4	News		✓	-	-	30,538

Table 17: Statistics of NER datasets.

#Dataset	#Types	#Major Domain	#Train	#Test	#Train Num	#Valid Num	#Test Num
ADE corpus (Gurulingappa et al., 2012)	5	Biomedical	✓	✓	3417	427	428
CoNLL 2004 (Roth and Yih, 2004)	5	News	✓	✓	922	231	288
GIDS (Jat et al., 2018)	4	News	✓	✓	8526	1417	4307
NYT (Riedel et al., 2010)	24	News	✓	✓	56,196	5000	5000
NYT11 (Takanobu et al., 2018)	12	News	✓	✓	60765	-	362
kbp37 (Zhang and Wang, 2015)	18	News	✓	✓	15917	1724	3405
SciERC (Luan et al., 2018a)	7	Scientific	✓	✓	1861	275	551
semeval RE (Hendrickx et al., 2010)	10	Scientific	✓	✓	6507	1493	2717
FewRel (Han et al., 2018)	100	Wikipedia		✓	-	-	17291
Wiki-ZSL (Chen and Li, 2021)	83	Wikipedia		✓	-	-	23113
DuIE2.0 (Luan et al., 2018b)	48	News	✓	✓	171126	20652	-
CMeIE (Guan et al., 2020)	53	Biomedical	✓	✓	14339	3585	-
COAE2016[6]	9	General		✓	-	-	971
IPRE (Wang et al., 2019)	35	General		✓	-	-	3340
SKE2020[7]	49	News		✓	-	-	3601

Table 18: Statistics of RE datasets. Since the test set for DuIE2.0 is not open-sourced, we use the valid set as our evaluation data.

#Dataset	#Types	#Major Domain	#Train	#Test	#Train Num	#Valid Num	#Test Num
ACE05 (Walker et al., 2006)	33	News	✓	✓	19216	901	676
CASIE (Satyapanich et al., 2020)	5	Cybersecurity	✓	✓	3732	777	1492
PHEE (Sun et al., 2022)	2	Biomedical	✓	✓	2897	960	968
CrudeOilNews (Lee et al., 2021)	18	Oil News		✓	-	-	356
RAMS (Ebner et al., 2020)	106	News		✓	-	-	887
WikiEvents (Li et al., 2021)	31	Wikipedia		✓	-	-	249
DuEE1.0 (Li et al., 2020b)	65	News	✓	✓	11908	1492	-
DuEE-fin (Han et al., 2022)	13	Finance	✓	✓	7015	1171	-
CCF_law[8]	9	Law		✓	-	-	971
FewFC (Zhou et al., 2021)	5	Finance		✓	-	-	2879

Table 19: Statistics of EE datasets. Since the test sets for DuEE1.0 and DuEE-fin are not open-sourced, we use the valid sets as our evaluation data.

```
----------------------------------------Instruction-----------------------------------------

# The following is an example of Chinese Named Entity Recognition (NER) in an object-oriented programming paradigm. In this example, several entity classes are defined, and then objects of these classes are instantiated to correspond with the entities in the given sentence.

# Please study this example and complete the English NER task accordingly.

# Input (Chinese NER):
class Entity:
    """
    描述：实体（Entity）指的是现实世界的对象，如人，地点，组织，产品等，可以用适当的名称表示；它可以是抽象的，也可以是物理存在的。Entity类是所有实体类别的基类。
    """
    def __init__(self, name: str):
        self.name = name

class Location(Entity):
    """
    描述：地理实体，如城市，国家等物理地点。
    例子：“美国”，“德国”，“伦敦”，“日本”，“纽约”，“伊利诺伊州基尔迪尔”，“萨维诺内”，“金布雷斯”，“美国”，“加利福尼亚州”。
    """
    pass

sentence = "哥伦比亚已经开始在军事技术方面进行创新，为其自身军队以及世界其他军队提供支持；特别是在设计和制造个人防弹保护产品，军事装备，军事机器人，炸弹，模拟器和雷达方面。"

# Output (Chinese NER):
results = [
    Location("哥伦比亚")
]

# Input (English NER):
class Entity:
    """
    Description: An entity is a real-world object, such as a person, place, organization, product, etc., which can be represented by an appropriate name; it can be abstract or physically existent. The Entity class is the base class of all entity classes.
    """
    def __init__(self, name: str):
        self.name = name

class Location(Entity):
    """
    Description: Geographical entities such as cities, countries or any identifiable physical place.
    Examples: "United States", "Germany", "London", "Japan", "New York", "Kildeer , Illinois", "Savignone", "Kinbrace", "US", "CA".
    """
    pass

sentence = "Colombia has begun to innovate in military technology for its army and other armies of the world ; especially in the design and creation of personal ballistic protection products , military hardware , military robots , bombs , simulators and radar ."

----------------------------------------Completion-----------------------------------------

# Output (English NER):
results = [
    Location("Colombia")
]
```

Figure 2: An example of instruction-tuning data for the IE cross-lingual alignment phase.

```
----------------------------------------Instruction-----------------------------------------

class Entity:
    """
    Description: An entity is a real-world object, such as a person, place, organization, product, etc., which can be represented by an appropriate name; it can be abstract or physically existent. The Entity class is the base class of all entities.
    """
    def __init__(self, name: str):
        self.name = name

class Organization(Entity):
    """
    Description: Each organization or set of organizations mentioned in a document gives rise to an entity of type Organization. An Organization must have some formally established association. Typical examples are businesses, government units, and sports teams. Industrial sectors are also treated as Organizations.
    Examples: "Reuters", "U.N.", "NEW YORK", "CHICAGO", "OSCE", "Bloomsbury Minerals Economics", "Eurostat", "Philip Morris", "AD", "3".
    """
    pass

class Person(Entity):
    """
    Description: First, middle and last names of people, animals and fictional characters aliases.
    Examples: "Clinton", "Dole", "Arafat", "Yeltsin", "Dutroux", "Mariaan de Swardt", "Ronald Waterreus", "MITCHELL", "Das", "BAN".
    """
    pass

class Location(Entity):
    """
    Description: Locations defined on a geographical or political basis which are mentioned in a document. These include, for example, China, the U.S., the Hudson River, Mt. Everest. However, ("Chinese", and "Syrian") are not entities.
    Examples: "U.S.", "Germany", "Britain", "Australia", "England", "RIO DE JANEIRO", "Acatepec", "Yemen", "IA", "IL".
    """
    pass

"""
This is an object-oriented programming task: For DATASET "CoNLL2003-NER", some Entity Classes are defined above. Please instantiate all the corresponding Entity Objects in the following sentence.
"""

sentence = "Coste said he had approached the player two months ago about a comeback ."

----------------------------------------Completion------------------------------------------

results = [
    Person("Coste")
]
```

Figure 3: An example of instruction-tuning data for the multilingual IE training phase.

PROMPT FOR DESCRIPTION INITIALIZATION.

Writing Entity Descriptions

Introduction
This guide provides a step-by-step process for writing a clear, concise, and accurate description of an entity type based on a provided list of examples. The objective is to generalize the shared characteristics of the examples without referencing any specific instance, giving a broad and comprehensive understanding of the entity type.

Prerequisites
Before you begin, make sure you have the following:
- **Entity Type**: The name of the entity type that requires a description.
- **Entity List**: A set of examples representing this entity type.
- **Basic Description**: While not mandatory, familiarity with the general concept of the entity type could be beneficial.

Step-by-Step Instructions
Step 1: Begin with the Required Phrase
Each description should start with:
[Entity Type] refers to
This ensures consistency across all descriptions. Replace **[Entity Type]** with the actual type name.

Step 2: Generalize the Shared Characteristics
- Review the **Entity List** to identify common traits among all examples.
- Avoid referring to specific examples directly. Generalize to cover the entire group.
- Example: If the list includes various vehicles (cars, trucks), the description should focus on common traits such as modes of transportation designed for movement.

Step 3: Provide Comprehensive Coverage
- The description should encapsulate all critical aspects represented in the example list, accounting for any outliers or unusual cases.
- Example: If the list includes motorized vehicles and non-motorized bicycles, ensure the description covers both.

Step 4: Output the Description
- After completing the description, present it without referencing explicit examples. It should summarize the entity type in a single, generalized statement.
- If you want to revise the description, output the modified description in the format.

Conclusion
By following these steps, you will create an accurate, clear, and generalized description of an entity type. Start with the required phrase, focus on generalization, and keep the language simple yet precise.

[Input and Output]

Table 20: Prompt for Description Initialization.

PROMPT FOR DESCRIPTION POLISH.

Evaluating and Revising Entity Description

Introduction
This guide provides a systematic approach to evaluate whether a given description accurately represents the characteristics of an entity type. If the description is accurate and complete, no revision is necessary. However, if inaccuracies or omissions exist, revisions are required to ensure clarity and consistency in classifying entities.

Step-by-Step Instructions
Step 1: Analyze the Entity Type Description
- Carefully review the **entity type description** provided.
- Example: For the entity type “Animal,” the description may include “living organisms that move, breathe, and consume organic matter.”

Step 2: Analyze the Entity
- Review the specific entity’s characteristics, noting its unique features.
- Example: If the entity is “dog,” note traits like “mammal, four-legged, domesticated, etc.”

Step 3: Evaluate the Description’s Accuracy and Completeness
- Compare the entity type description with the entity’s characteristics.
  - Does the description fully encompass the defining features of the entity?
  - Are any characteristics missing or misrepresented?
- Check for completeness:
  - Does the description cover all essential traits necessary for classification?
- Verify accuracy:
  - Are the described attributes factually correct?

Step 4: Revise the Description (if necessary)
- If the description is incomplete or inaccurate, revise it to reflect the entity’s correct characteristics.
- Ensure the revised description is clear, precise, and free from ambiguities.

Conclusion
Following these steps will ensure each entity’s description is both accurate and comprehensive. This process maintains clarity and consistency in classifying entities under their respective types.

Input
Entity Type: {entity_type}
Entity Example List: {entity_example_list}

Example Template for Output
Entity Type: {entity_type}
Entity Example List: {entity_example_list}
Entity Type Description: "{entity_type} refers to..."(in the language of the Entity list)

Table 21: Prompt for Description Polish.

PROMPT FOR JOINT TRANSLATION.

Translate the sentence and spans from English to Chinese.

Please follow these guidelines:
1. Translate each span considering the context of the sentence.
2. Ensure the number of spans after translation matches the original number of spans.
3. When outputting spans, ensure only to output the translation of each span.

The following is a few examples:

[English]
"sentence": "The EU rejected Germany's call for a boycott of British lamb."
"spans": ["EU"]
[Chinese]
"sentence": "欧盟拒绝德国呼吁抵制英国羊肉。"
"spans": ["欧盟"]

[English]
"sentence": "FM involves 2 - 4.7% of the general population."
"spans": []
[Chinese]
"sentence": "FM 影响了2 - 4.7% 的普通人群。"
"spans": []

[English]
"sentence": "4000 guests from home and abroad attended the opening ceremony."
"spans": ["home", "abroad"]
[Chinese]
"sentence": "4000名来自国内和国外的嘉宾出席了开幕式。"
"spans": ["国内", "国外"]

Please translate the following sentence and spans:
[English]
"sentence": "{src_sentence}"
"spans": [{src_spans}]
[Chinese]

Table 22: Prompt for Joint Translation.

PROMPT FOR SPAN REPHRASE.

Please find the Chinese span corresponding to the English span in the Chinese sentence.

Please follow these guidelines:
1. Only find the span in the Chinese sentence that corresponds to the English span.
2. Ensure that the Chinese span must be semantically consistent with the English span.

The following is an example:

[English]
"sentence": "Siemens invested 800 million US dollars to complete the electric power plant project."
"spans": ["US"]
[Chinese]
"sentence": "西门子投资了8亿美元完成了电力厂项目。"
"spans": ["美"]

Please find the corresponding span in the Chinese sentence:
[English]
"sentence": "{src_sentence}"
"spans": [{src_span}]
[Chinese]
"sentence": "{tgt_sentence}"
"spans":

Table 23: Prompt for Span Rephrase.

PROMPT FOR SENTENCE REPHRASE.

Please translate the following sentence from English to Chinese.

Please follow these guidelines:
1. Ensure that the translation includes the following spans: [{tgt_lang_spans}].
2. If the target sentence is semantically inconsistent with the source sentence, return "modification failure".

[English]
"sentence": "{src_sentence}"
[Chinese]
"sentence":

Table 24: Prompt for Sentence Rephrase.