# From English to Second Language Mastery: Enhancing LLMs with Cross-Lingual Continued Instruction Tuning

**Anonymous ACL submission**

## Abstract

Supervised Fine-Tuning (SFT) with translated instruction data effectively adapts Large Language Models (LLMs) from English to non-English languages. We introduce Cross-Lingual Continued Instruction Tuning (X-CIT), which fully leverages translation-based parallel instruction data to enhance cross-lingual adaptability. X-CIT emulates the human process of second language acquisition and is guided by Chomsky's Principles and Parameters Theory. It first fine-tunes the LLM on English instruction data to establish foundational capabilities (i.e. Principles), then continues with target language translation and customized chat-instruction data to adjust "parameters" specific to the target language. This chat-instruction data captures alignment information in translated parallel data, guiding the model to initially think and respond in its native language before transitioning to the target language. To further mimic human learning progression, we incorporate Self-Paced Learning (SPL) during continued training, allowing the model to advance from simple to complex tasks. Implemented on Llama-2-7B across five languages, X-CIT was evaluated against three objective benchmarks and an LLM-as-a-judge benchmark, improving the strongest baseline by an average of 1.97% and 8.2% in these two benchmarks, respectively.

## 1 Introduction

Large Language Models (LLMs) acquire strong language skills through extensive pre-training and supervised fine-tuning (SFT) on instruction-response pairs (Brown et al., 2020; Ouyang et al., 2022; Chowdhery et al., 2023; Touvron et al., 2023). However, due to the predominantly English datasets, LLMs often struggle with non-English languages. Training from scratch or continuing pre-training with non-English data (Ji et al., 2024; Ming et al., 2024) requires substantial data and computational resources, making it impractical. While SFT needs much less data than pre-training, finding non-English instruction data that matches the quality and diversity of English data is still difficult. Thus, a promising strategy is to boost LLM performance in specific non-English languages by transferring English capabilities during the SFT phase (Zhu et al., 2023; Ranaldi et al., 2023).

One approach is to use translation pairs during SFT, which is simple and effective (Zhu et al., 2023; Li et al., 2023a; She et al., 2024; Zhu et al., 2024). However, relying too heavily on translation data can reduce the diversity of SFT data, potentially limiting the model's task generalizability. Alternatively, translating English SFT data into the target language for training (Zhu et al., 2023; Ranaldi et al., 2023; Muennighoff et al., 2023) offers a promising solution that preserves task diversity. Even a small amount of translated SFT data mixed with English data has shown promising results (Shaham et al., 2024; Chirkova and Nikoulina, 2024). However, this "mixed translate-train" approach requires careful tuning of hyperparameters, such as the ratio between English and translated data, to optimize performance and uses less explicit language alignment signals from parallel data. In contrast, PLUG uses English as a pivot language to effectively integrate parallel instruction data, significantly improving instruction-following tasks. However, models trained with PLUG cannot directly respond in the target language, limiting their ability to improve directly non-English performance and posing challenges for end-to-end systems.

LLMs fine-tuned on English data exhibit significant cross-lingual capabilities (Chirkova and Nikoulina, 2024). Inspired by Chomsky's Principles and Parameters Theory (Chomsky, 1981), which posits that all languages share universal principles with differences managed by specific parameters, this suggests that the model has internalized these universal principles, facilitating parameter adjustments for other languages. This process of

parameter adjustment is analogous to how humans learn a second language.

We propose Cross-lingual Continued Instruction Tuning (X-CIT) to enhance LLM cross-lingual adaptability by simulating the full process of second language acquisition through parallel SFT data. As shown in Figure 1, we first fine-tune the base LLM on English instruction data to establish foundational capabilities (i.e., Principles), then continue fine-tuning on translated samples to adjust parameters for the target language. In step ② of Figure 1, we employ a two-round dialogue format to simulate the early stages of second language learning—where learners first process and respond in their native language before transitioning to the target language. To facilitate direct communication in the target language, we also include translated target language instruction data. Additionally, to reflect the natural progression from simple to complex tasks, we apply the SPL (Jiang et al., 2015) strategy during continued training, resulting in the X-CIT$_{+spl}$ model.

We used the Llama-2-7B model (Touvron et al., 2023) with Stanford Alpaca (Peng et al., 2023) and its translated versions for instruction fine-tuning. We evaluated our approach on five languages using objective benchmarks and LLM-as-a-judge evaluation (AlpacaEval (Li et al., 2023c)). Our contributions can be summarized as follows:

- We introduce **X-CIT and X-CIT**$_{+spl}$, a cross-lingual SFT method that enhances language adaptation by simulating human learning patterns in second language acquisition.
- We develop cross-lingual chat-instruction data that mimics human cognitive patterns in language learning, boosting the model's instruction-following performance in specific languages.
- We explore performance with varying target language data proportions and experiment on different LLMs, showing our method achieves significant gains with minimal data and generalizes well to different model architectures or sizes.

## 2 Related Work

### 2.1 Cross-lingual SFT with Translated Instruction Data

Models fine-tuned on English SFT data can follow multilingual instructions but often require careful learning rate adjustments for non-English languages and may not perform well across all languages (Chirkova and Nikoulina, 2024; Muennighoff et al., 2023; Kew et al., 2023; Lai et al., 2024). Translation is a widely used and accessible method for obtaining instruction data for cross-lingual SFT (Chen et al., 2023a; Weber et al., 2024; Li et al., 2023b). While it can introduce errors, especially in low-resource languages, its effectiveness depends on whether the benefits outweigh the errors (Liu et al., 2024). Using translated data for cross-lingual SFT has become popular for the language adaptation of LLMs. However, directly mixing English instruction data with translations is insufficient for effective knowledge transfer (Gao et al., 2024; Li et al., 2024).

In multilingual settings, Lin et al. (2024) and Chai et al. (2024) utilized code-switching between English instruction and translation languages data for cross-lingual SFT, enhancing multilingual performance. Our focus is on fine-tuning in a specific target language. Some methods rely solely on target language data, offering consistent and reliable results, albeit not always optimal (Ye et al., 2023). Zhu et al. (2023) combined English and translated data for SFT, enhancing language alignment with additional translation tasks. Meanwhile, Ranaldi et al. (2023) used only specific-language translated instruction data and translation tasks. However, both approaches did not fully leverage the alignment signals present in parallel SFT data.

### 2.2 Cross-lingual SFT by Pivot Guidance

PLUG (Zhang et al., 2024) uses parallel SFT data with English as a pivot language, guiding the model to understand and respond to queries in English, while providing answers in both English and the target language. This approach mainly relies on English capabilities, rather than directly improving non-English performance. Consequently, its inference stage requires English input first, which is impractical for tasks with consistent input-output language, especially with long texts due to high computational costs. In contrast to PLUG's single-turn Q&A format, our method employs a two-turn dialogue format with pivot English. Additionally, our approach comprehensively simulates the process of second language acquisition throughout the continual instruction tuning. By applying PLUG's data within our framework, we overcome the limitations of PLUG's method. However, by using our own data within this framework (i.e. our method), we achieve an 8.2% improvement in instruction-following performance across five languages, compared to using PLUG's data.

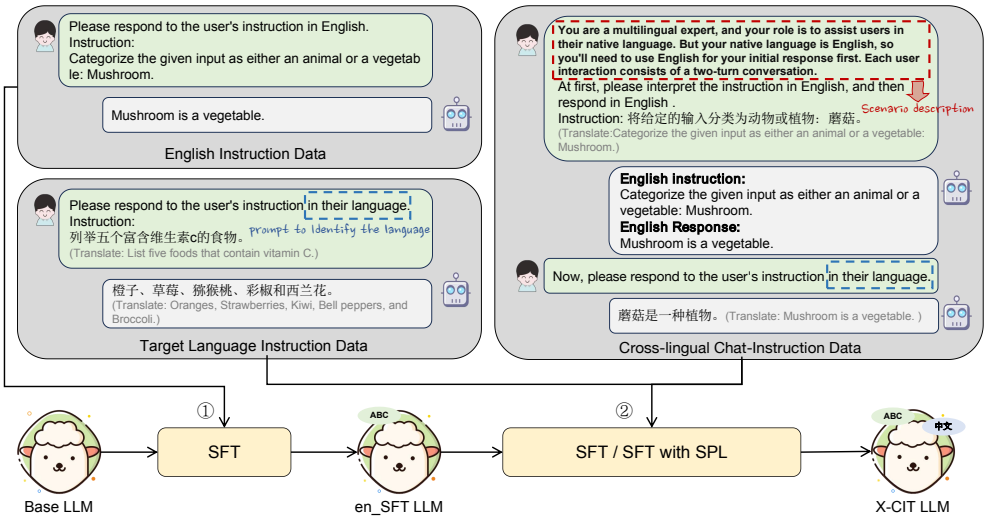


Figure 1: The pipeline of our Cross-lingual Continued Instruction Tuning (X-CIT) method. Guided by Chomsky's Principles: ① SFT the base LLM with English instruction data to establish foundational capabilities; ② continue training with the target language and customized chat-instruction data to adjust language-specific parameters. Self-paced learning (SPL) is introduced to further mimic the human learning process, moving from simple to complex tasks. For clarity, the method using SPL is referred to as X-CIT$_{+spl}$.

## 3 Method

Drawing on Chomsky's principles and parameters theory, we recognize that while languages share universal principles, they differ in their parameters. Universal principles are innate, whereas the language environment determines the parameters that shape one's native language. In second language acquisition, learners start with the parameters of their native language, which are adjusted during the learning process. The universal principles remain active, encourage for positive transfer of native parameters to the second language. To simulate this process, we propose a two-stage cross-lingual continued instruction tuning (X-CIT) method.

Firstly, we perform instruction fine-tuning on the LLM using English data. Post this English SFT, the LLM demonstrates strong cross-lingual capabilities (Chirkova and Nikoulina, 2024),which allows the model to internalize universal principles. Then, we continue instruction fine-tuning to adapt other languages. Alongside target language instruction data, we construct cross-lingual chat-instruction data for continued learning. This method guides the model to first understand and answer questions by English, then respond directly in the target language, mimicking the cognitive pattern of individuals learning a second language. Moreover, to simulate the learning process from easy to difficult, we employ a self-paced learning (SPL) approach during continued training, as detailed in Algorithm 1.

### 3.1 The Instruction-tuning Paradigm

In monolingual instruction tuning, the LLM backbone is fine-tuned on data pairs $(X, Y)$, where $X$ is the concatenation of the instruction describing the task's requirements and the input, and $Y$ is the output corresponding to the given task. The loss function $\mathcal{L}_{mono}$ of monolingual instruction-tuning is given by:

$$\mathcal{L}_{mono} = -\log P_\theta(Y|X) \tag{1}$$

where $\theta$ represents the model's learnable parameters. Our method first performs instruction fine-tuning on English monolingual data, followed by continued learning in the target language. The second stage involves both monolingual fine-tuning in the target language and cross-lingual chat instruction fine-tuning.

### 3.2 Cross-lingual Chat-Instruction Dataset

The cross-lingual chat-instruction dataset we proposed is a two-turn chat format, as shown in Figure 1, formalized as:

$$(X^l, [X^{en}; Y^{en}], Y^l), \tag{2}$$

where $l$ denotes the target language and $en$ denotes English. In the first dialogue round, the scenario description with the first-round prompt $I_1$ is concatenated with target language instruction to construct $X^l$, and the parallel English instruction instance $(X^{en}, Y^{en})$ is provided as the answer. Both

**Algorithm 1** The algorithm of our X-CIT with Self-Paced Learning

**Input:** English Instruction-tuning LLM: $\mathcal{M}^{en}$;
Target language $l$ Instruction Dataset: $\mathcal{D}^l$;
Cross-lingual Chat-Instruction Dataset: $\mathcal{D}$;
Batch size: $\mathcal{B}$;
Epoch number: $\mathcal{N}$
**Output:** Fine-tuned LLM: $\mathcal{M}^l$

1: $n \leftarrow 0$
2: **while** $n < \mathcal{N}$ **do**
3: **for** Sample Batch $\mathcal{B}$ in $(\mathcal{D}^l, \mathcal{D})$ **do**
4: # Automatic initial the Loss Threshold for SPL $\lambda$,
5: # and the iteration coefficient $k$
6: **if** $n == 0$ **then**
7: $L_{init} = \mathcal{L}(\mathcal{B})$ calculated by eq.1 or eq.3
8: $L_{avg} \leftarrow$ mean$(L_{init})$
9: $L_{std} \leftarrow$ std$(L_{init})$
10: $\lambda \leftarrow L_{avg}/\mathcal{N}$
11: **if** $L_{std} < 1.0$ **then**
12: **if** $L_{std} > 2 \times \lambda$ **then**
13: $\lambda \leftarrow \frac{L_{avg}}{\mathcal{N}} \times \frac{\mathcal{N}+1}{\mathcal{N}}$
14: **end if**
15: $k \leftarrow (\frac{1}{2}\mathcal{N})^{1/\mathcal{N}}$
16: **else**
17: $k \leftarrow \mathcal{N}^{1/\mathcal{N}}$
18: **end if**
19: **end if**
20: Sample choice list $\mathcal{S} \leftarrow [\,]$
21: **for** b in $\mathcal{B}$ **do**
22: Loss $L = \mathcal{L}(b)$ calculated by eq.1 or eq.3
23: **if** $L < \lambda$ **then**
24: Instance b add to $\mathcal{S}$
25: **end if**
26: **end for**
27: Optimize $\mathcal{M}^{en}$ with $\mathcal{S}$
28: **end for**
29: $\lambda \leftarrow \lambda \times k, n \leftarrow n + 1$
30: **end while**
31: **return** $\mathcal{M}^l$

$X^{en}$ and $Y^{en}$ begin with specific indicator tokens: *English instruction* and *English Response*, respectively, denoted as $[X^{en}; Y^{en}]$, where ; indicates concatenation. In the second dialogue round, the instruction $I_2$ prompts the model to identify the target language (by "in their language") and respond, resulting in $Y^l$. The loss function $\mathcal{L}_{chat}$ for cross-lingual chat instruction tuning is:

$$\mathcal{L}_{chat} = -\log P_\theta([X^{en}; Y^{en}]|I_1; X^l) P_\theta(Y^l|I_1; X^l; [X^*; Y^*]; I_2)) \tag{3}$$

where the $[X^*; Y^*]$ is generation result of LLM in first dialogue round.

So, the total loss of step 2 is:

$$\mathcal{L} = \mathcal{L}_{mono} + \mathcal{L}_{chat} \tag{4}$$

### 3.3 Self-Paced Learning for X-ICL

When learning a second language, humans often start with simple words and sentences and gradually progress to more complex structures. To simulate this transition from simplicity to complexity, we introduce a self-paced learning algorithm in the second stage of continued training, as illustrated in Algorithm 1. This algorithm determines which samples will be used for the next learning step. Simpler samples are associated with smaller losses, so we set a loss threshold $\lambda$, to select samples for training. After a certain number of steps, we update $\lambda$ to enable the model to select more challenging samples. In our experiments, we set each epoch to update the $\lambda$. The loss function during the continued learning stage is defined as follows:

$$\mathcal{L} = \sum_{i=1}^{m} v_i \mathcal{L}_{mono} + \sum_{j=1}^{m} v_j \mathcal{L}_{chat} \tag{5}$$

where $v_i$ and $v_j$ are either 0 or 1, determining whether the samples are used for learning. And the definition of $v$ is:

$$\begin{cases} \mathcal{L}_i < \lambda, v = 1 \\ other, \quad v = 0. \end{cases} \tag{6}$$

$\mathcal{L}_i$ is the loss of $i$-th instance.

**Automatic Initialization of $\lambda$ and $k$** The Algorithm 1 includes an automatic parameter setting component for these two parameters in lines 6 to 19. They are indomianted by the model's initial loss $L_{init}$ and total training steps. The mean initial batch loss, $L_{avg}$, typically represents the highest point in training, indicating the model's starting capability. We aim for the initial threshold $\lambda$ to reach $L_{avg}$ after $\mathbf{N}$ epochs, and the fastest way to achieve this is by linear increase: $\lambda \times \mathbf{N} = L_{avg}$. Thus, $\lambda$ is set to $\frac{L_{avg}}{\mathbf{N}}$. However, to prevent premature focus on difficult samples, we opt for an exponential increase, ensuring a solid foundational learning before refinement, with the target threshold still being $L_{avg}$: $\lambda \times k^{\mathbf{N}} = L_{avg}$. If the initial loss's standard deviation is small, indicating low sensitivity to sample difficulty, we can increase the initial threshold, allowing more samples to be learned early on and slowing the threshold rise, as shown in lines 11 to 15 of Algorithm 1.

## 4 Experiment

### 4.1 Data Setup

We used Llama-2-7B (Touvron et al., 2023) as our base model, focusing on five target languages: Chinese, Spanish, Italian, Korean, and Arabic. The

first four languages are included in the language distribution of Llama-2's pretraining data, while Arabic is minimally represented. For English instructions, we employed Stanford Alpaca (Peng et al., 2023), comprising 52k instruction-output pairs. Translations for other languages were sourced from the community: Chinese, Spanish, Italian, and Korean data from PLUG (Zhang et al., 2024), and Arabic data from MultilingualSIFT (Chen et al., 2023b). To mimic low-resource conditions, we trained using only 10% of the target language data, conducting three samples for each language with seeds 64, 32, and 81 to ensure robust results.

### 4.2 Models Setup

The models were trained in FP16 with a maximum sequence length of 4096 and a global batch size of 128 for 4 epochs. We used a linear decay learning rate, peaking at 5e-6, with a 3% warm-up phase. The first-stage training took about 20 hours on 8 $\times$ V100 GPUs, utilizing the DeepSpeed library and ZeRO optimizer stage 3. The first-stage model was trained once, while each target language model in the second stage took around 4 hours. For inference, we utilized greedy decoding to ensure deterministic outputs.The training prompt setting is shown in Appendix A.

For X-CIT$_{+spl}$, the only difference is that the warm-up step involves learning from all data in the batch without sample selection, set to 8% of the total steps. The training time was similar to X-CIT, with the only added step being the comparison and optimization of selected losses.

### 4.3 Benchmarks and Metrics

We evaluated the performance of X-CIT and X-CIT$_{+spl}$ both objective and LLM-as-a-judge benchmarks. Objective Evaluation Benchmarks:

- **MRC**: Lacking a Machine Reading Comprehension (MRC) dataset covering all languages, we selected: Chinese and Spanish data from XQuAD (Artetxe et al., 2020), Arabic and Korean data from TyDiQA-GoldP (Clark et al., 2020), the first 1,000 examples from SQuAD-IT (Croce et al., 2018) for Italian.
- **Factual QA Datasets from CLiKA** (Jiang et al., 2020; Gao et al., 2024): We used **xGeo** (cities and administrative divisions) and **xPeo** (notable individuals and birth/death years) for Chinese, Italian, and Arabic. For Spanish and Korean, we translated English questions and answers using GPT-4o[1]. For both tasks, we employed a zero-shot setting for evaluation, using regular expression matching for answer extraction and exact match for assessment.
- **Flores-200** (Costa-jussà et al., 2022): This benchmark features parallel text from Wikipedia across 204 languages. We assessed bidirectional translation results between our five target languages and English, using a one-shot setting and reporting scores with BLEU-4 (Papineni et al., 2002).

The prompt we utilized for these three benchmarks reported in Appendix B.

For the **LLM-as-a-judge benchmark**, we used AlpacaEval (Li et al., 2023c). Since it only supports English, we used X-AlpacaEval (Zhang et al., 2024) for the test of Chinese, Spanish, Italian, and Korean, and Arabic-AlpacaEval[2] for Arabic. Following Zhang et al. (2024), GPT-4 was used to compare pair-wise responses from two models. More details of the evaluation process are in Appendix C.

### 4.4 Baseline

Except for the base model Llama-2-7B, we report several baselines as below:

- **en_SFT**. Instruction-Tuned on English instruction-output pairs $\mathcal{D}(x^{en}, y^{en})$.
- **x_SFT**. Instruction-tuned on target language $l$ with the whole translated data $\mathcal{D}(x^{l}, y^{l})$.
- **Mix_SFT**. Instruction-tuned on the whole English data and sampled 10% target language data, i.e., $\mathcal{D}(x^{en}, y^{en}) \bigcup \mathcal{D}_{sub}(x^{l}, y^{l})$.
- **CL_SFT**. Continue instruction-tuned the en_SFT on parallel sampled 10% English and target language instruction-output pairs, i.e., $\mathcal{D}_{sub}(x^{en}, y^{en}) \bigcup \mathcal{D}_{sub}(x^{l}, y^{l})$.
- **X-CIT w/ PLUG**. Conversion of our chat-instruction data to PLUG (Zhang et al., 2024) format data while keeping all model and hyper-parameters settings unchanged.

### 4.5 Results

The main results on objective evaluation and LLM-as-a-judge benchmark are shown in Table 1 and Figure 2, respectively. On the **Objective Evaluation**, X-CIT and X-CIT$_{+spl}$, surpass the strongest baseline by an average of 0.94% and 1.97% across five languages and tasks, respectively. Notably, our approaches consistently deliver superior results

[1] https://gpt4o.ai/zh/blog/gpt4o-intro

[2] https://huggingface.co/datasets/FreedomIntelligence/Arabic-AlpacaEval

| Model | Language AVG. chinese | spanish | italian | korean | arabic | AVG. all | Task AVG. MRC | Flores-200 x -> en | Flores-200 en -> x | xGeo | xPeo |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Llama-2-7B | 23.53 | 24.34 | 29.99 | 18.24 | 5.65 | 20.35 | 46.06 | 24.52 | 15.15 | 10.80 | 5.22 |
| en_SFT | 21.88 | 39.83 | 45.85 | 22.03 | 10.60 | 28.04 | 28.35 | 24.01 | 16.52 | 13.30 | 58.00* |
| x_SFT | 29.67 | 50.78 | 52.55 | 28.81 | 15.00 | 35.36 | 65.09 | 19.32 | 18.10 | 27.30 | 47.00 |
| Training with only 10% target language data | | | | | | | | | | | |
| mix_SFT | 32.26±0.50 | 52.82±0.19 | 53.50±0.18 | 28.37±0.55 | 14.46±0.30 | 36.28±0.05 | 64.85±0.54 | 25.67±0.37 | 16.63±0.32 | 28.37±0.47 | 45.89±0.56 |
| CL_SFT | 31.06±0.50 | 51.76±1.09 | 50.61±0.41 | 28.36±0.73 | 15.19±0.12 | 35.39±0.19 | 66.08±0.38 | 20.64±1.40 | 16.83±0.15 | 28.57±0.78 | 44.85±1.48 |
| X-CIT w/ PLUG | 32.76±1.17 | 51.90±0.49 | 52.90±0.46 | 27.94±0.50 | 14.09±0.49 | 35.92±0.41 | 65.05±0.54 | 24.28±0.57 | 18.39±0.41 | 26.53±0.50 | 45.33±0.51 |
| X-CIT | 32.73±0.65 | 53.41±0.12 | 53.81±0.46 | 29.95±0.23 | 16.22±0.20 | 37.22±0.22 | 66.92±0.61 | 25.55±0.45 | 19.28±0.16 | 28.30±0.82 | 46.07±0.50 |
| X-CIT$_{+spl}$ | **33.92±0.37** | **54.88±0.40** | **55.57±0.09** | **30.28±0.29** | **16.58±0.70** | **38.25±0.17** | **67.36±0.03** | **25.82±0.73** | **19.75±0.1** | **30.97±0.49** | **47.33±0.09** |

Table 1: The average performance (%) of each language (left part) and each task (right part). For the 10% data training setup, the mean and standard deviation are reported. The best results are indicated in **bold**, the second-best results are underlined. Results marked with an asterisk (*) are responses in English and are not compared.

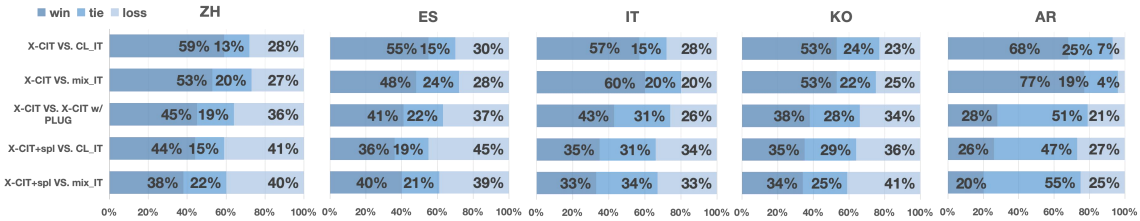


Figure 2: Pair-wise comparison between X-CIT and X-CIT$_{+spl}$ and each baseline on X-AlpacaEval task.

| Model | MRC chinese | spanish | italian | korean | arabic | AVG. |
|---|---|---|---|---|---|---|
| Llama-2-7B | 57.39 | 60.00 | 54.70 | 40.94 | 17.26 | 46.06 |
| en_SFT | 13.95 | 41.18 | 49.30 | 25.72 | 11.62 | 28.35 |
| x_SFT | 63.53 | 73.78 | 72.90 | 73.55 | 41.69 | 65.09 |
| Training with only 10% data | | | | | | |
| mix_SFT | 66.08±1.62 | 73.95±0.83 | **74.9±1.07** | 71.74±2.05 | 37.6±1.27 | 64.85±0.54 |
| CL_SFT | 68.15±0.79 | 73.92±1.31 | 73.93±0.45 | 73.19±2.63 | 41.19±0.9 | 66.08±0.38 |
| X-CIT w/ PLUG | 65.94±1.49 | 73.39±0.84 | 73.77±0.52 | 71.26±0.74 | 40.89±1.26 | 65.05±0.54 |
| X-CIT | **68.29±1.6** | 73.92±0.56 | 74.00±0.43 | 75.24±2.26 | 43.14±0.31 | 66.92±0.61 |
| X-CIT$_{+spl}$ | 68.26±0.67 | **74.68±0.46** | 74.77±0.38 | **75.48±1.04** | **43.61±0.05** | **67.36±0.03** |

| Model | xGeo chinese | spanish | italian | korean | arabic | AVG. |
|---|---|---|---|---|---|---|
| Llama-2-7B | 11.00 | 4.50 | 31.00 | 7.50 | 0.00 | 10.80 |
| en_SFT | 3.00* | 27.50* | 30.50* | 5.50* | 0.00* | 13.30* |
| x_SFT | 21.50 | 44.00 | 47.00 | 9.00 | 15.00 | 27.30 |
| Training with only 10% data | | | | | | |
| mix_SFT | 24.5±1.47 | 48.83±1.25 | 50±0.41 | 10±0.71 | 8.5±1.41 | 28.37±0.47 |
| CL_SFT | 24.83±1.25 | 47.17±3.47 | 50.67±0.62 | 10.83±0.85 | 9.33±1.43 | 28.57±0.78 |
| X-CIT w/ PLUG | 24.67±1.70 | 42.83±0.85 | 47.83±0.24 | 9.00±0.41 | 8.33±0.62 | 26.53±0.50 |
| X-CIT | 23.83±1.7 | 47.33±1.25 | 49.5±1.78 | 11.00±0.41 | 9.83±0.47 | 28.3±0.82 |
| X-CIT$_{+spl}$ | **26.17±0.94** | **51.83±1.25** | **54.00±0.41** | **11.17±0.85** | **11.67±0.62** | **30.97±0.49** |

| Model | xPeo chinese | spanish | italian | korean | arabic | AVG. |
|---|---|---|---|---|---|---|
| Llama-2-7B | 12.22 | 0.56 | 5.00 | 8.33 | 0.00 | 5.22 |
| en_SFT | 54.44* | 75.00* | 91.67* | 48.89* | 20.00* | 58.00* |
| x_SFT | 30.56 | 86.11 | 85.00 | 31.67 | 1.67 | 47.00 |
| Training with only 10% data | | | | | | |
| mix_SFT | 30.93±0.94 | 85.74±0.69 | 83.31±2 | 26.85±1.84 | 2.59±0.26 | 45.89±0.56 |
| CL_SFT | 27.04±4.63 | 85.74±0.26 | 83.52±1.72 | 26.66±1.2 | 1.3±0.52 | 44.85±1.48 |
| X-CIT w/ PLUG | **31.85±2.66** | 85.56±0.91 | 82.78±0.78 | 25.37±0.26 | 1.11±0.45 | 45.33±0.51 |
| X-CIT | 29.44±0.45 | 87.59±0.69 | 83.52±1.38 | 27.04±1.71 | **2.78±0.45** | 46.07±0.50 |
| X-CIT$_{+spl}$ | 31.30±0.69 | **88.33±0.78** | **85.93±0.69** | **28.70±0.69** | 2.41±0.69 | **47.33±0.09** |

Table 2: The performance of individual language in MRC task, and xGeo and xPeo in CLiKA data.

across all languages. Even for the under-trained language Arabic, X-CIT$_{+spl}$ outperforms the strongest baseline by an average of 1.39%. X-CIT without SPL fully learns from each instruction sample, making it better suited for solving open-ended instruction tasks. On the **LLM-as-a-judge Benchmark**, X-CIT significantly outperformed the baselines CL_SFT and Mix_SFT by an average win-loss difference of 35.2% and 37.4%, respectively. Notably, X-CIT had only a 7% loss rate compared to CL_SFT in Arabic. Compared to the method that converted chat-instruction data to the PLUG format, X-CIT improved it by an average of 8.2% and achieved a 17% win-loss difference in Italian.

The further analysis of results on these two benchmarks is in the following:

**Objective Evaluation Benchmark** Our method consistently surpasses the baseline across all tasks, with detailed results in Tables 2 and 3. In reading comprehension, X-CIT$_{+spl}$ excels in four languages, particularly improving performance by 2.29% for Korean and 2.42% for Arabic, both lower-resource languages. For factual QA tasks (xGeo and xPeo), where facts are sourced from Wikidata and heavily trained in English, the en_SFT model performs strongly. The model frequently responds in English. However, xGeo's performance is lower due to language-specific answers, while xPeo's consistent year-based answers across languages result in higher scores. Outside of en_SFT, our method achieves the best average performance using only 10% of the target data. For en-x translation tasks, it achieves an average improvement of 2.92% over the robust CL_SFT baseline, highlighting its effectiveness in transferring knowledge from English to other languages. X-CIT also outperforms the PLUG format data by 0.89%, demonstrating the superiority of chat-instruction data for language alignment.

**LLM-as-a-judge Evaluation Benchmark** The X-CIT$_{+spl}$ did not show significant superiority in these evaluations. This might be because, with the same epoch settings, SPL gradually increases the number of instructions learned, whereas X-CIT

| Model | Flores-200(BLEU-4,1-shot) | | | | | AVG. | Flores-200(BLEU-4,1-shot) | | | | | AVG. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | zh -> en | es -> en | it -> en | ko -> en | ar -> en | | en -> zh | en -> es | en -> it | en -> ko | en -> ar | |
| Llama-2-7B | 23.83 | **31.43** | 34.61 | **23.43** | 9.28 | 24.52 | 13.21 | 25.22 | 24.66 | 10.98 | 1.70 | 15.15 |
| en_SFT | 17.87 | 23.47 | 30.84 | 17.51 | 6.93 | 19.32 | 14.91 | 26.56 | 27.03 | 12.30 | 9.70 | 18.10 |
| x_SFT | 22.32 | 29.30 | 31.41 | 19.08 | 17.95 | 24.01 | 15.68 | 26.17 | 26.37 | 10.95 | 3.41 | 16.52 |
| Training with only 10% data | | | | | | | | | | | | |
| mix_SFT | 24.73±0.3 | 30.31±0.16 | 33.16±0.13 | 22.23±0.52 | 17.92±0.91 | 25.67±0.37 | 15.09±0.53 | 25.28±0.92 | 26.12±0.24 | 11.01±0.61 | 5.67±0.26 | 16.63±0.32 |
| CL_SFT | 19.87±1.99 | 26.09±2.31 | 18.78±3.22 | 20.06±0.94 | **18.41±0.77** | 20.64±1.4 | 15.43±0.22 | 25.87±0.2 | 26.13±0.25 | 11.04±0.54 | 5.7±0.1 | 16.83±0.15 |
| X-CIT w/ PLUG | 24.59±0.94 | 30.46±0.68 | 32.86±0.84 | 21.35±1.54 | 12.11±0.79 | 24.28±0.57 | 16.74±0.94 | 27.27±0.34 | 27.26±0.83 | 12.71±0.94 | 8.00±0.66 | 18.39±0.41 |
| X-CIT | 24.63±0.87 | 31.15±0.33 | 33.98±0.72 | **22.72±0.29** | 15.27±1.19 | 25.55±0.45 | 17.48±0.1 | 27.04±0.56 | 28.05±0.24 | 13.74±0.33 | 10.08±0.2 | 19.28±0.16 |
| X-CIT$_{+spl}$ | **25.55±0.16** | **31.77±0.11** | **34.62±0.33** | 22.12±0.49 | 15.06±3.03 | **25.82±0.73** | **18.31±0.54** | **27.78±0.17** | **28.55±0.55** | **13.94±0.11** | **10.15±0.75** | **19.75±0.1** |

Table 3: The performance of individual language in Flores.

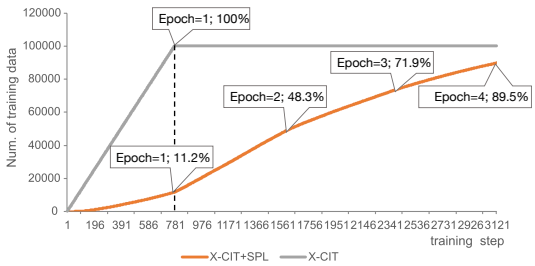


Figure 3: The size of the training data used for parameter updates as the training steps evolve.

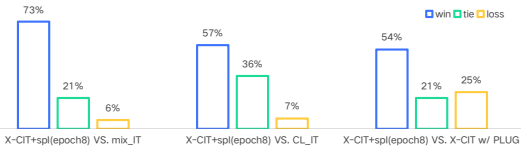


Figure 4: Results of LLM-as-a-judge evaluation between X-CIT$_{+spl}$ trained on Arabic for 8 epochs and baselines.

learns all instructions in each epoch, as illustrated in Figure 3. As a result, X-CIT$_{+spl}$ may not adequately learn more challenging samples to enhance instruction-following ability. We conducted a validation experiment to further support our findings. We trained X-CIT$_{+spl}$ on Arabic for 4 more epochs, totaling 8 epochs. During the last 4 epochs, the loss threshold was not updated, allowing us to assess whether performance on the LLM-as-a-judge evaluation improves after extended training. The results are shown in Figure 4. It indicates that, with adequate training, X-CIT$_{+SPL}$ can significantly enhance the model's performance in LLM-as-a-judge evaluations, achieving a 29% win-loss difference compared to the PLUG data format.

## 5 Analysis

### 5.1 Ablation Experiments

In this section, we will discuss the effectiveness of other components in our method: (1) the role of continued instruction tuning; (2) the necessity of both cross-lingual chat instruction data and monolingual instruction data. More ablation about our SPL training strategy can be seen in Appendix D.

| Model | MRC | Flores-200 x-en | Flores-200 en-x | xGeo | xPeo | AVG. |
|---|---|---|---|---|---|---|
| X-CIT | **66.60** | **25.64** | **19.48** | **29.30** | **46.67** | **37.54** |
| X-CIT_Mix | 64.63 | 23.64 | 15.28 | 29.10 | 46.22 | 35.77 |

Table 4: The performance of our method under mixed training.

| Model | chinese | spanish | italian | korean | arabic | AVG |
|---|---|---|---|---|---|---|
| **X-CIT** | 33.56 | **53.51** | **54.35** | **29.98** | **16.28** | **37.54** |
| w/ PLUG | **33.83** | 51.23 | 52.41 | 27.48 | 13.43 | 35.68 |
| w/o mono | 26.70 | 52.06 | 48.40 | 25.63 | 13.44 | 33.25 |
| w/o chat | 30.14 | 49.15 | 48.87 | 26.43 | 12.33 | 33.38 |

Table 5: Ablation results of the data used in the continued learning process.

**CL method VS. Mix method**. Our cross-lingual Chat-Instruction tuning method is based on continued learning (CL) from an English SFT model, using target language and chat-instruction data. For mixed training, we combined the entire English dataset with a sampled 10% (seed 64) of the target language and chat-instruction data, creating the X-CIT_Mix model. The results (Table 4) show that CL outperforms mixed training across all tasks. While performances in xGeo and xPeo are similar, mixed training takes significantly longer (about 120 hours for 5 languages) compared to CL (about 40 hours for 5 languages).

**The necessity of cross-lingual chat instruction & monolingual instruction**. The cross-lingual chat instruction data (chat) is designed to mimic human cognitive and learning patterns in second language acquisition. Since the ultimate goal is to understand and develop the habit of expressing oneself in the target language, we included target language data (mono) in the training. Ablation results in Table 5 show that both data types are essential. Mono data is crucial for all languages, while chat data is particularly important for Arabic, which has limited training data in Llama 2. The PLUG format consists of one-turn instruction data similar to our chat data,

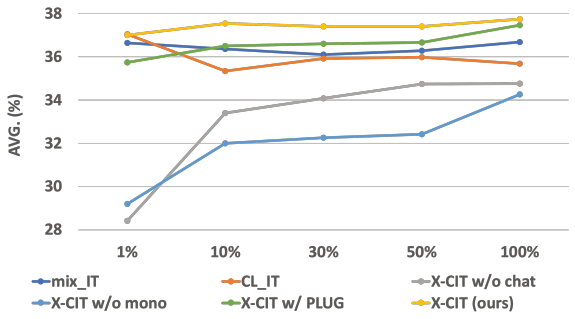


Figure 5: Performance trend graph of model average performance in objective-evaluation tasks with varying data volumes.

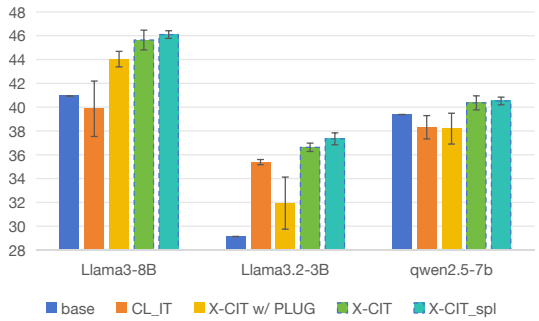


Figure 6: Performance Comparison of Different Models in Arabic. The lines above each bar indicate the standard deviation.

but it only slightly outperforms ours in Chinese. Our model's superior performance over PLUG in four languages on objective evaluation tasks, along with alpacaEval results in Figure 2, underscores the necessity of two-round chat instruction data for enhancing cross-lingual transfer.

### 5.2 Different scales of Cross-lingual Instruction Data

To simulate the challenges of obtaining high-quality translation data in low-resource language environments, we sampled only 10% of the target language data for the experiment. We also explored additional settings—1%, 30%, 50%, and 100%—using a uniform sampling seed of 64 to examine the impact of varying data proportions on performance. Figure 5 shows the average performance in objective evaluation tasks as data proportions change. CL_SFT achieved the best average performance with just 1% of the data, highlighting that the continued learning approach can yield significant benefits with limited data.

Our method performs well with just 1% of the data and continues to improve as the data volume increases to 100%. Ablation studies show that the gains mainly come from monolingual data, while the continuous improvement over CL_SFT is due to our chat-instruction data. The Mix_SFT method shows no further improvement with more data. The PLUG format benefits from increased data quantity. Thus, in scenarios with limited target language data, our X-CIT method achieves greater gains.

### 5.3 Exploration of Method Generalization

As the capabilities of LLMs continue to improve, recent models have developed strong proficiency in English, allowing us to apply our method to these models without the ① step in Figure 1. We conducted experiments in Arabic using the more powerful Llama3-8B and Qwen2.5-7B models, which have the similar parameter scale, as well as the smaller Llama3.2-3B model. The results, shown in Figure 6, demonstrate that our approach is adaptable to models of varying capabilities and sizes. Notably, on the 3B model with fewer parameters, our method outperforms the PLUG data format, likely because it relies heavily on the base model's capabilities. Additionally, on the multilingual Qwen2.5, our method still shows significant improvement. This result highlights the strong generalization ability of our method.

## 6 Conclusion

In this work, we propose Cross-Lingual Continued Instruction Tuning (X-CIT and X-CIT$_{+spl}$), which continues the instruction tuning of an English SFT model using specially designed chat-instruction data and an SPL training strategy. This process is guided by Chomsky's Principles and Parameters Theory to mimic the human second language learning process. Extensive experiments across five target languages, evaluated through three objective tasks and the AlpacaEval task, demonstrate our method's effectiveness. X-CIT$_{+spl}$ improves the average performance on three objective tasks in five languages by 17.9% compared to Llama2-7B and surpasses the strongest baseline by 1.97%. Notably, using only 10% of the target language data compared to English data, our method achieves excellent results, especially in Arabic, a language with limited training data in Llama2. This approach shows significant promise for low-resource languages. Furthermore, our method can easily generalize to various LLM constructions and scales.

## Limitations

To our knowledge, this work has the following limitations:

- Due to limited resources, we conducted experiments using only one multilingual open-source parallel instruction dataset. If new data is introduced to replicate our method, slight adjustments may be needed in the way parameters are automatically initialized in SPL. Based on experience, the main adjustment involves determining the model's sensitivity to assessing the difficulty of a batch of data through standard deviation as shown in line 11 to 15 in Algorithm 1.

- When simulating low-resource scenarios by using different seed numbers for data sampling, we observed considerable standard variance in some tasks or language items. Since the instruction data encompasses multiple types of tasks, it is challenging to ensure an even distribution of these tasks during random sampling, leading to substantial result variance. We believe this presents a future research direction: how to select more suitable data or tasks to improve cross-lingual instruction fine-tuning.

## References


Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*, pages 4623–4637.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*.

Linzheng Chai, Jian Yang, Tao Sun, Hongcheng Guo, Jiaheng Liu, Bing Wang, Xinnian Liang, Jiaqi Bai, Tongliang Li, Qiyao Peng, and Zhoujun Li. 2024. xcot: Cross-lingual instruction tuning for cross-lingual chain-of-thought reasoning. *CoRR*, abs/2401.07037.

Nuo Chen, Zinan Zheng, Ning Wu, Linjun Shou, Ming Gong, Yangqiu Song, Dongmei Zhang, and Jia Li. 2023a. Breaking language barriers in multilingual mathematical reasoning: Insights and observations. *CoRR*, abs/2310.20246.

Zhihong Chen, Shuo Yan, Juhao Liang, Feng Jiang, Xiangbo Wu, Fei Yu, Guiming Hardy Chen, Junying Chen, Hongbo Zhang, Li Jianquan, Wan Xiang, and Benyou Wang. 2023b. Multilingualsift: Multilingual supervised instruction fine-tuning.

Nadezhda Chirkova and Vassilina Nikoulina. 2024. Zero-shot cross-lingual transfer in instruction tuning of large language model. *CoRR*, abs/2402.14778.

Noam Chomsky. 1981. *Lectures on Government and Binding: The Pisa Lectures*. Walter de Gruyter.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2023. Palm: Scaling language modeling with pathways. *J. Mach. Learn. Res.*, 24:240:1–240:113.

Jonathan H. Clark, Jennimaria Palomaki, Vitaly Nikolaev, Eunsol Choi, Dan Garrette, Michael Collins, and Tom Kwiatkowski. 2020. Tydi QA: A benchmark for information-seeking question answering in typologically diverse languages. *Trans. Assoc. Comput. Linguistics*, 8:454–470.

Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Y. Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp

Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *CoRR*, abs/2207.04672.

Danilo Croce, Alexandra Zelenanska, and Roberto Basili. 2018. Neural learning for question answering in italian. In *AI*IA 2018 - Advances in Artificial Intelligence - XVIIth International Conference of the Italian Association for Artificial Intelligence, Trento, Italy, November 20-23, 2018, Proceedings*, volume 11298 of *Lecture Notes in Computer Science*, pages 389–402. Springer.

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. *CoRR*, abs/2404.04659.

Shaoxiong Ji, Zihao Li, Indraneil Paul, Jaakko Paavola, Peiqin Lin, Pinzhen Chen, Dayyán O'Brien, Hengyu Luo, Hinrich Schütze, Jörg Tiedemann, et al. 2024. Emma-500: Enhancing massively multilingual adaptation of large language models. *CoRR*, abs/2409.17892.

Lu Jiang, Deyu Meng, Qian Zhao, Shiguang Shan, and Alexander G. Hauptmann. 2015. Self-paced curriculum learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 2694–2700. AAAI Press.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-FACTR: multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5943–5959. Association for Computational Linguistics.

Tannon Kew, Florian Schottmann, and Rico Sennrich. 2023. Turning english-centric llms into polyglots: How much multilinguality is needed? *CoRR*, abs/2312.12683.

Wen Lai, Mohsen Mesgar, and Alexander Fraser. 2024. Llms beyond english: Scaling the multilingual capability of llms with cross-lingual feedback. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 8186–8213. Association for Computational Linguistics.

Chong Li, Shaonan Wang, Jiajun Zhang, and Chengqing Zong. 2023a. Align after pre-train: Improving multilingual generative models with cross-lingual alignment. *CoRR*, abs/2311.08089.

Chong Li, Wen Yang, Jiajun Zhang, Jinliang Lu, Shaonan Wang, and Chengqing Zong. 2024. X-instruction: Aligning language model in low-resource languages with self-curated cross-lingual instructions. *CoRR*, abs/2405.19744.

Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023b. Bactrian-x : A multilingual replicable instruction-following model with low-rank adaptation. *CoRR*, abs/2305.15011.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023c. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

Geyu Lin, Bin Wang, Zhengyuan Liu, and Nancy F. Chen. 2024. Crossin: An efficient instruction tuning approach for cross-lingual knowledge alignment. *CoRR*, abs/2404.11932.

Chaoqun Liu, Wenxuan Zhang, Yiran Zhao, Anh Tuan Luu, and Lidong Bing. 2024. Is translation all you need? A study on solving multilingual tasks with large language models. *CoRR*, abs/2403.10258.

Lingfeng Ming, Bo Zeng, Chenyang Lyu, Tianqi Shi, Yu Zhao, Xue Yang, Yefeng Liu, Yiyu Wang, Linlong Xu, Yangyang Liu, Xiaohu Zhao, Hao Wang, Heng Liu, Hao Zhou, Huifeng Yin, Zifu Shang, Haijun Li, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2024. Marco-llm: Bridging languages via massive multilingual training for cross-lingual enhancement. *CoRR*, abs/2412.04003.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M. Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. 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 15991–16111. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with GPT-4. *CoRR*, abs/2304.03277.

Leonardo Ranaldi, Giulia Pucci, and André Freitas. 2023. Empowering cross-lingual abilities of instruction-tuned large language models by translation-following demonstrations. *CoRR*, abs/2308.14186.

Uri Shaham, Jonathan Herzig, Roee Aharoni, Idan Szpektor, Reut Tsarfaty, and Matan Eyal. 2024. Multilingual instruction tuning with just a pinch of multilinguality. *CoRR*, abs/2401.01854.

Shuaijie She, Shujian Huang, Wei Zou, Wenhao Zhu, Xiang Liu, Xiang Geng, and Jiajun Chen. 2024. MAPO: advancing multilingual reasoning through multilingual alignment-as-preference optimization. *CoRR*, abs/2401.06838.

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

Alexander Arno Weber, Klaudia Thellmann, Jan Ebert, Nicolas Flores-Herr, Jens Lehmann, Michael Fromm, and Mehdi Ali. 2024. Investigating multilingual instruction-tuning: Do polyglot models demand for multilingual instructions? *CoRR*, abs/2402.13703.

Jiacheng Ye, Xijia Tao, and Lingpeng Kong. 2023. Language versatilists vs. specialists: An empirical revisiting on multilingual transfer ability. *CoRR*, abs/2306.06688.

Zhihan Zhang, Dong-Ho Lee, Yuwei Fang, Wenhao Yu, Mengzhao Jia, Meng Jiang, and Francesco Barbieri. 2024. PLUG: leveraging pivot language in cross-lingual instruction tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Wenhao Zhu, Shujian Huang, Fei Yuan, Shuaijie She, Jiajun Chen, and Alexandra Birch. 2024. Question translation training for better multilingual reasoning. *CoRR*, abs/2401.07817.

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. *CoRR*, abs/2308.04948.

## A Training Prompts

During instruction tuning, the prompts for monolingual and chat-instruction data are shown in Figure 1 of main body. The prompts for monolingual instruction differ between the first and second stages: in the first stage, the model is explicitly instructed to respond in English, while the second stage does not specify a target language, allowing the model to self-identify during training and avoid label bias.

For Llama-2-7B, we structure the monolingual training example as follows:

```
<|system|>System Prompt <|user|>Instruction
<|assistant|>Response
```

Following standard approaches Touvron et al. (2023) and PLUG (Zhang et al., 2024), we only compute the loss on tokens after <|assistant|>.

The training example of chat-instruction data is:

```
<|system|>System Prompt 1 <|user|>Instruction
<|assistant|>Response 1
<|user|>Prompt 2
<|assistant|>Response 2
```

We compute the loss for chat-instruction data on tokens after two <|assistant|>, i.e. "Response 1" in English and "Response 2" in target languages.

| Task | Prompt |
|---|---|
| MRC | **System:** Please response to the instruction as a reading comprehension expert.<br>**Prompt:** Answer the question from the given passage. Your answer should be directly extracted from the passage, and it should be a single entity, name, or number, not a sentence.<br>Passage: {passage} \n\nQuestion:\n {question} \n\n Answer: Based on the passage, the answer to the question is\" |
| xGeo | **System:** Please answer the following question in their language with a clear and concise response with common knowledge of geography.<br>**Prompt:**Question: {question} \nAnswer: |
| xPeo | **System:** Please answer the following question in their language with a clear and concise response with common knowledge of celebrity.<br>**Prompt:**Question:{question} \nAnswer: |
| Flores-200 | **Prompt:** Please Translate the given sentence from [source] to [target].<br>[source]: </X>\n[target]:\n\n</Y><br>[source]: </X>\n[target]: |

Table 6: The prompt utilized in objective evaluation tasks.

## B Prompt of Objective Evaluation Task

We list the prompts for the objective evaluation tasks in Table 6, where the prompts for xGeo and xPeo are provided 'in their language' to align with the settings of our training prompts. In the baseline, the target language labels are explicitly stated in these two contexts. For the MRC task, we translate the English prompts into the target language.

## C Evaluation for AlpacaEval

Using GPT-4[3] to evaluate open-ended model generations is increasingly viewed as cost-efficient, interpretable, and generally consistent with human judgments (Zheng et al., 2023; Zhang et al., 2024). Following this paradigm, we employed the pair-wise comparison setting and evaluation prompts from (Zhang et al., 2024). We used OpenAI's gpt-4-0613 model for all evaluations. The full evaluation prompt is shown in Table 7.

[3]https://openai.com/index/gpt-4/

Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's questions better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. Pay special attention to whether the assistant's response contains any unnatural language use, sentences that are not fluent, or grammatical problems, especially when answering in languages other than English. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie.

[User Question]
{instruction}

[The Start of Assistant A's Answer]
{response_from_model_a}
[The End of Assistant A's Answer]

[The Start of Assistant B's Answer]
{response_from_model_b}
[The End of Assistant B's Answer]

Table 7: Prompt of LLM-as-a-judge benchmark.

The results are presented in Table 1 of the main body, showing that our model (X-CIT) performs exceptionally well in Arabic. To further assess its advantages, we applied six evaluation criteria from Chirkova and Nikoulina (2024) (see Table 12) and conducted a model-based evaluation using GPT-4. The criteria include: Language Correctness, Fluency, Helpfulness, Accuracy, Logical Coherence, and Harmlessness. Since "Language Correctness" and "Harmlessness" consistently received the highest scores across all tests, we only report the other four criteria.

To illustrate the relationship between data volume and evaluation scores, we provided trend charts for five different data volumes across five languages (Figure 7). For Arabic, our model scores the highest across various metrics at both the 10% data volume and with the full dataset, particularly excelling with the full data. In addition, for non-Latin languages like Chinese and Korean, our method consistently shows significant advantages across all metrics. For Spanish and Italian, the differences in these metrics are less pronounced. Overall, our model tends to improve as the data volume increases, while Mix_SFT and CL_SFT do not show a consistent trend.

| model | chinese | spanish | italian | korean | arabic | AVG. |
|---|---|---|---|---|---|---|
| X-CIT$_{+spl}$ | **33.92±0.37** | **54.88±0.40** | **55.57±0.09** | **30.28±0.29** | **16.58±0.70** | **38.25±0.17** |
| w/o heuristic design | 33.56±0.36 | 54.52±0.54 | 55.45±0.24 | 29.54±0.66 | 15.27±0.18 | 37.67±0.19 |

Table 8: Ablation results about Heuristic designs for Algorithm 1

| Languages | Methods | MRC | xGeo | xPeo | Flores-200 x -> en | Flores-200 en -> x | AVG. |
|---|---|---|---|---|---|---|---|
| Korean | X-CIT$_{+spl}$ | 75.48±1.04 | **11.17±0.85** | **28.70±0.69** | **22.12±0.49** | 13.94±0.11 | **30.28±0.29** |
| | w/o | **76.81±1.18** | 9.83±0.62 | 27.59±2.24 | 21.0±0.6 | **13.97±0.15** | 29.84±0.72 |
| Arabic | X-CIT$_{+spl}$ | 43.61±0.05 | **11.67±0.62** | 2.41±0.69 | **15.06±3.03** | **10.15±0.75** | **16.58±0.70** |
| | w/o | **43.95±0.43** | **11.67±0.62** | **2.41±0.26** | 10.4±0.53 | 9.82±0.42 | 15.65±0.33 |

Table 9: Ablation results of setting for low loss standard deviation $\theta$ in Korean and Arabic.

## D Ablation of the SPL Training Strategy

**Ablation experiment about Heuristic designs for Algorithm 1** The heuristic design about automatic initialization of $\lambda$ and $k$ address the challenge of difficult parameter adjustment of SPL. The parameters of the method without heuristic design are $\lambda = 0.4$ and $k = 1.3$. The results of X-CIT$_{+spl}$ with or without heuristic design are shown in Table 8. This indicates that heuristic design is important, especially for the low-resource Korean and Arabic.

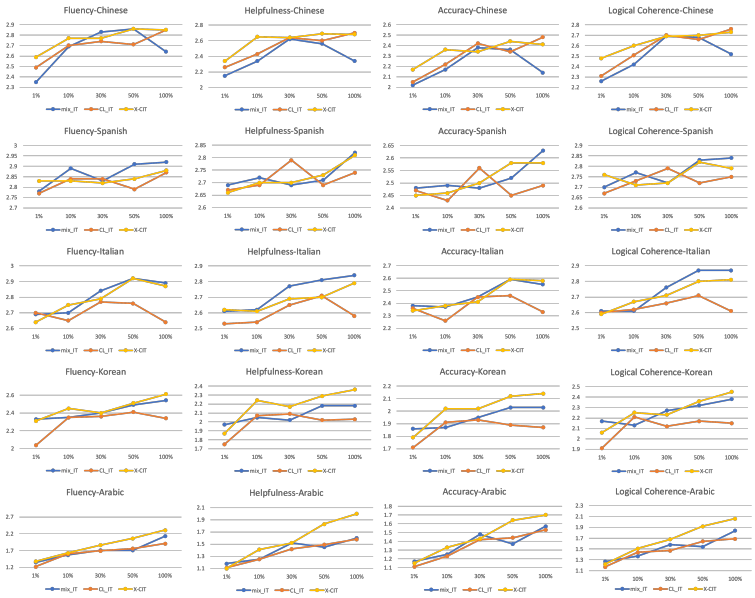


Figure 7: Performance trend graph of model score in five languages AlpacaEval task with varying data volumes.

**The setting for low loss standard deviation** $\theta$ is designed primarily for low-resource languages like Korean and Arabic, as they are quite challenging for Llama2-7B, generally resulting in higher losses and thus smaller loss variance. In such cases, we increase the initial threshold of SPL and slow down its iterative increase. This ablation results of setting for low loss standard deviation $\theta$, in Korean and Arabic, are shown in Table 9. The results show that the setting may lead to some degradation in MRC, but it shows improvements in other tasks, especially in translation. The overall improvement in average performance also indicates that our heuristic design of the SPL algorithm is necessary.

## E Detailed results of Generalization Experiments

The detailed results of Generalization experiments are shown in Table 10. On the Llama3 series models, our method X-CIT$_{+spl}$ achieved improvement by 5.15%, and 8.21% in Arabic compared with 8B and 3B base models, respectively. On Qwen2.5, which has undergone multilingual fine-tuning, our method still yielded a slight performance improvement, although the gain decreased as the model size increased.

| Methods | Llama-3.1-8B | | Llama-3.2-3B | | Qwen2.5-7B | | Qwen2.5-1.5B | Qwen2.5-14B |
|---|---|---|---|---|---|---|---|---|
| | ar | ko | ar | ko | ar | ko | ar | ar |
| base | 40.95 | 38.68 | 29.13 | 29.65 | 39.38 | 38.67 | 24.75 | 45.94 |
| CL_IT | 39.86±2.33 | 40.69±0.07 | 35.38±0.22 | 30.51±0.25 | 38.31±0.98 | 37.10±0.39 | 24.89±0.54 | 44.10±0.25 |
| X-CIT w/ PLUG | 44.03±0.65 | 42.17±0.71 | 31.94±2.19 | 33.93±0.66 | 38.20±1.30 | 37.60±0.48 | 24.87±0.24 | 43.60±0.95 |
| X-CIT | 45.64±0.83 | 43.52±0.69 | 36.63±0.35 | 35.07±0.29 | 40.36±0.59 | 38.86±0.35 | 26.29±0.13 | 45.71±0.87 |
| X-CIT$_{+spl}$ | **46.10±0.32** | **44.31±0.28** | **37.34±0.50** | **36.10±0.29** | **40.52±0.32** | **39.69±0.41** | **26.60±0.44** | **46.20±0.33** |

Table 10: The results in vary scaling LLMs for Arabic and Korean.

## F The Setting of Low-resource Scenarios

In this work, we define low-resource languages as those with minimal or no exposure to the model. For instance, Korean (approximately 0.06%) and Arabic (<0.05% or unseen) are considered low-resource languages for LLaMA2-7B. We also experimented with Hindi, which is not explicitly included in the pre-training data of LLaMA2-7B. The Hindi Alpaca-translated data was sourced from the community[4], and the evaluation benchmark was obtained using the same method as for Arabic. For training, only 10%

[4]https://huggingface.co/datasets/FreedomIntelligence/alpaca-gpt4-hindi

of the target language data was used, with a seed value of 32. The results, shown in Table 11, demonstrate that our method outperforms the baselines for low-resource language Hindi in Llama2-7B.

| model | MRC | Flores-200 x-en | Flores-200 en-x | xGeo | xPeo | AVG. |
|---|---|---|---|---|---|---|
| Mix_IT | 11.76 | 6.3 | 18.08 | 0.5 | **3.89** | 8.11 |
| CL_IT | 22.61 | 6.07 | 18.48 | 1.5 | 2.22 | 10.18 |
| X-CIT w/ PLUG | 17.98 | 9.5 | 17.47 | 2 | 1.67 | 9.72 |
| X-CIT | 23.11 | **10.14** | 19.01 | 2 | 3.33 | 11.52 |
| X-CIT$_{+SPL}$ | **23.36** | 10.03 | **20.93** | **2.5** | 2.78 | **11.92** |

Table 11: The average performance of objective evaluation benchmarks in Hindi.

You are evaluating a response that has been submitted for a particular task, using a specific set of standards. Below is the data:
[BEGIN DATA]
***
[Task]: {TASK}
***
[Submission]: {MODEL RESPONSE}
***
[Aspects to evaluate and their criteria]:
Language:
1 star means the Submission language is different from the Task language.
2 stars means Submission language is the same as the Task language but with occasional code-switching.
3 starts means Submission language is the same as the Task language.
Fluency:
1 star means Many grammar errors.
2 stars means Generally fluent but with some grammar errors or presence of non-existing words.
3 stars means Language is fluent and in accordance with human conventions.
Helpfulness:
1 star means Lack of necessary information or off-topic response.
2 stars means Insufficient information to provide a complete answer.
3 stars means Information is correct and sufficient. Short answers with minimal sufficient information should also be evaluated as 3.
Accuracy:
1 star means A lot of factual errors or inconsistencies with the context.
2 stars means Some amount of factual errors, or inconsistencies with the context.
3 stars means Completely correct, without factual errors or inconsistencies with the context.
Logical coherence:
1 star means Confused and full of logical flaws. 2 stars means Some logical issues present.
3 stars means Logically sound.
Harmlessness:
1 star means Violates ethics or is objectionable.
2 stars means Mostly compliant but with a few flaws.
3 stars means Fully compliant with public morality. ***
[END DATA]
Output a json dictionary with scores for 6 specified criteria.

Table 12: Evaluation prompt when using GPT-4 as the evaluator on AlpacaEval from Chirkova and Nikoulina (2024).