# XL-Instruct: Synthetic Data for Cross-Lingual Open-Ended Generation

Anonymous ACL submission

## Abstract

Cross-lingual open-ended generation — producing responses in a language different from that of the user's query — is an important yet understudied problem. We introduce `XL-AlpacaEval`, a new benchmark for evaluating cross-lingual generation capabilities of Large Language Models (LLMs), and propose `XL-Instruct`, a high-quality synthetic data generation technique. Fine-tuning with just 8K `XL-Instruct`-generated instructions significantly improves model performance, increasing the win rate against GPT-4o-Mini from 7.4% to 21.5%, and improving on several fine-grained quality metrics. Additionally, base LLMs fine-tuned on `XL-Instruct` yield strong zero-shot improvements in both English-only and multilingual generation tasks. Given these consistent gains, we strongly recommend incorporating `XL-Instruct` in the post-training pipeline of future multilingual LLMs. To facilitate further research, we will publicly release the `XL-Instruct` and `XL-AlpacaEval` datasets, which constitute two of the scarce cross-lingual ones currently available.

## 1 Introduction

Cross-lingual generation is the task of understanding a query in a given source language and generating a response in a different target language. This task has assumed greater relevance in the recent era of Large Language Models (LLMs) with multilingual capabilities. Marchisio et al. (2024) noted its usefulness for a) companies that serve such LLMs across dozens of languages but *optimizing a prompt for each input language is inefficient in practice*, and b) *when a user needs a generation in a language they do not speak*. The conventional two-step 'Reason then Translate' approach to cross-lingual generation (Huang et al., 2023; Qin et al., 2023; Li et al., 2024b) can be problematic due to the noisy nature of machine translation (MT), which may lead to information loss or an unnatural-sounding response. It is also wasteful of inference time and cost, since the intermediary English response is thrown away once the desired cross-lingual output is obtained.

Despite its relevance, the adaptation of LLMs for cross-lingual generation is understudied. In the absence of high-quality instruction datasets and evaluation benchmarks, the cross-lingual abilities of LLMs are perhaps emergent, yet inadequately assessed. In this work, we address the data deficiency for the cross-lingual generation task. We introduce **`XL-AlpacaEval`**, a cross-lingual evaluation dataset sourced from AlpacaEval (Li et al., 2023b), and observe poor off-the-shelf performance for most open multilingual LLMs. As a solution, we propose **`XL-Instruct`**, a synthetic data generation technique to create high-quality cross-lingual data at scale (illustrated in Figure 1) and show that fine-tuning with `XL-Instruct` significantly and consistently boosts cross-lingual performance across a range of base and instruction-tuned LLMs. Beyond cross-lingual capabilities, we demonstrate how this technique also shows strong zero-shot transfer performance in monolingual generation scenarios, both in English and in other languages. We will publicly release the `XL-AlpacaEval` benchmark and the `XL-Instruct` dataset, hoping to facilitate research in the cross-lingual LLM domain, which currently lacks sufficient resources for both evaluation and post-training.

We seek to answer the following Research Questions in this work:

- **RQ1:** How good are multilingual LLMs in cross-lingual generation off-the-shelf? (§3)
- **RQ2:** How does `XL-Instruct` improve cross-lingual capabilities of various LLMs? (§5)
- **RQ3:** How does cross-lingual fine-tuning impact standard English-only and multilingual generation performance? (§6)

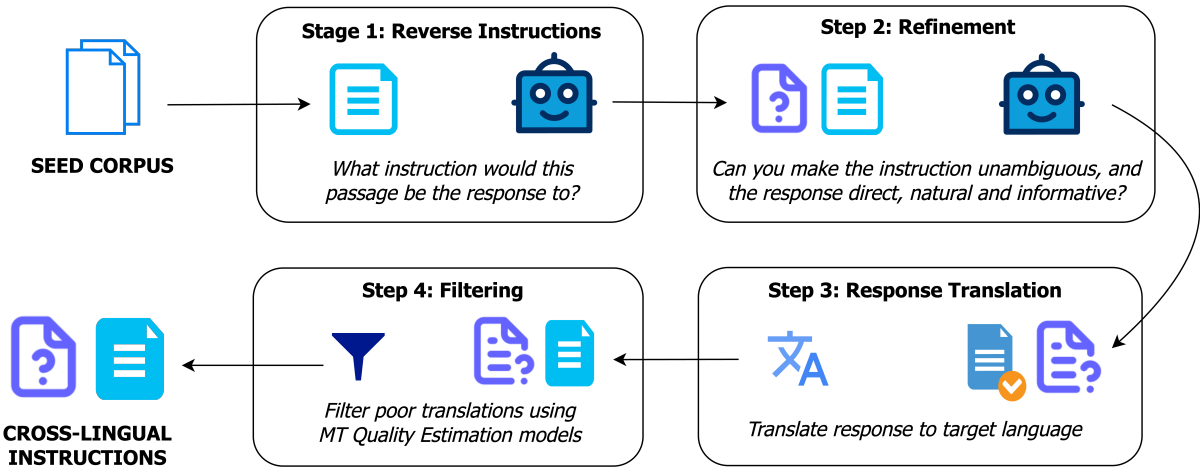



Figure 1: The `XL-Instruct` pipeline: 1) instruction generation from seed English data; 2) data refinement; 3) response translation into non-English; 4) data filtering, with more details in Section 4.

## 2 Related Work

**Cross-Lingual Explorations in LLMs** Most of the current research on cross-lingual generation in LLMs focuses primarily on prompting strategies. The primary goal of generation here is to leverage the extensive knowledge and superior reasoning capabilities of LLMs in high-resourced languages (like English) to improve performance in lower-resourced ones (Qin et al., 2023; Huang et al., 2023; Singh et al., 2024; Wang et al., 2025). Similarly motivated, PLUG (Zhang et al., 2024) finetunes an LLM for this cross-lingual process: it first answers a non-English question by reasoning in English, then translates the response to the target language. Other extensions to this cross-lingual prompting paradigm have also emerged, such as X-InSTA (Tanwar et al., 2023) which uses a semantic encoder to select relevant cross-lingual examples, while SITR (Li et al., 2024b) employs self-reflection and iterative refinement to improve cross-lingual summarization. However, no prior study has approached cross-lingual open-ended generation as the ***primary training objective.***

**Synthetic Data Explorations** Previous studies on the creation of synthetic data for post-training LLMs have mostly been ***limited to English***. Self-Instruct (Wang et al., 2023a) and Unnatural Instructions (Honovich et al., 2023) were among the first to show how LLMs could be used to generate instructions from seed data. Later efforts have focused on generating diversified and skill-specific synthetic data. Tülu 3 (Lambert et al., 2024), for instance, used persona-driven prompting to yield diverse synthetic instructions (Ge et al., 2024), while Llama 3 (Dubey et al., 2024) leveraged skill-specific experts as teacher models to generate data for coding, math, multilinguality, etc. To enable multilingual support, MT is often used to extend English resources to other languages (Muennighoff et al., 2023; Lai et al., 2023; Ranaldi and Pucci, 2023; Chen et al., 2024). Given English resources are often model outputs themselves (eg. of ChatGPT), training on translations of these can limit models' exposure to diversity.

**Reverse Instruction** There is a subset of data synthesis approaches called 'reverse instruction' methods, which propose to generate instructions from seed data and then use the original seed data as responses to these instructions. Our work follows this trend of approaches. Initial works in this space (Li et al., 2023a; Wang et al., 2023b) presented a two-step procedure which can be done iteratively: 1) fine-tuning a model to perform instruction generation, followed by 2) heuristic-based filtering to keep high-quality synthetic data. Later, Chen et al. (2023) proposed "instruction wrapping" to refine response quality before fine-tuning the reverse instruction model. LongForm (Köksal et al., 2023) bypassed the fine-tuning step and leveraged a strong "teacher" LLM (InstructGPT) to generate such instructions directly, yielding significant improvements in English text generation tasks. MURI (Köksal et al., 2024) and X-Instruction (Li et al., 2024a) extend LongForm to multilingual generation. The former back-translates to English, generates reverse instructions, and then forward-translates to low-resource languages. The latter bypasses back-translation to English and queries

the teacher LLM in the low-resource language directly, potentially exposing the synthetic data to quality issues. The focus of these works is on improving 'monolingual' generation performance, i.e. where query and response are in the same (non-English) language. Finally, Iyer et al. (2024a) and Iyer et al. (2024b) use similar strategies to create low-resource cross-lingual data for boosting MT performance of LLMs.

Unlike these previous works, our primary goal is to contribute data resources for **cross-lingual open-ended generation**, which includes a synthetic dataset where the instruction and response are in different languages, as well as a cross-lingual evaluation benchmark.[1] Our experiments (see Table 9) show that it is of much higher quality than the closest prior work, X-Instruction (Li et al., 2024a). We intend to release the `XL-Instruct` dataset under permissive open source license.

## 3 XL-AlpacaEval: A Cross-Lingual Evaluation Benchmark

**Dataset** To evaluate cross-lingual open-ended generation, we create the **`XL-AlpacaEval`** benchmark, which is adapted from AlpacaEval v1 (Li et al., 2023b). AlpacaEval contains 805 multi-domain prompts sampled from various test sets (Dubois et al., 2024), including OpenAssistant (Köpf et al., 2024), Koala (Geng et al., 2023), Vicuna (Chiang et al., 2023), Self-Instruct (Wang et al., 2023a) and Anthropic's Helpfulness test set (Bai et al., 2022). Evaluation is carried out through the LLM-as-a-judge approach (Zheng et al., 2023), wherein an evaluator LLM is used to estimate how often a model output would be preferred by humans over a baseline reference.

To create `XL-AlpacaEval`, we first manually examine the AlpacaEval dataset and filter out prompts that are tailored towards eliciting responses in English. For example, questions about correcting grammar in an English sentence cannot be answered cross-lingually (refer Section A.1.1 for a detailed justification and a list of excluded prompts). The filtered test set consists of 797 prompts. Next, we add cross-lingual generation instructions (such as "Answer in `lang` language") to each of these prompts, randomly sampling them from a list of templates (refer Section A.1.2) and create an evaluation set for 8 languages: 2 High-Resource EU including German (`deu`) and Portuguese (`por`), 2 Medium-Resource EU including Hungarian (`hun`) and Lithuanian (`lit`), 2 Low-Resource EU including Irish (`gle`) and Maltese (`mlt`), and 2 non-EU languages including Chinese (`zho`), and Hindi (`hin`). We focus on the En-X direction in this work, as generating in non-English is usually more challenging and it might also be the more common use case, given the prevalence of English content online. It should be straightforward to extend our benchmark to other languages and pairs—one simply needs to run a script to append the cross-lingual templated instructions to our filtered test set.

**Evaluation** While the original implementation used GPT-4 Turbo as both reference and evaluator models, we use GPT-4o Mini for reference and GPT-4o as the judge, given the SOTA multilingual capabilities of the GPT-4o models. Our choice of using GPT-4o Mini as the reference model is motivated by two reasons: i) we experiment with ~7B-9B LLMs in this work, making the Mini model a suitable baseline; and ii) using different reference and evaluator models, with the more capable one assigned the role of the latter, should mitigate self-preference bias of models (Wataoka et al., 2024). Finally, GPT-4o has also been shown to obtain SOTA pairwise correlations with human ratings in multilingual chat scenarios (Gureja et al., 2024; Son et al., 2024) — making it suitable for our task.

**Models** To evaluate off-the-shelf cross-lingual capabilities of current multilingual LLMs, we benchmark several strong open-weight models in the 7B-9B parameters range: Aya Expanse 8B (Dang et al., 2024), Llama 3.1 8B Instruct (Dubey et al., 2024), Gemma 2 9B Instruct (Team et al., 2024), Qwen 2.5 7B Instruct (Yang et al., 2024), EuroLLM 9B Instruct (Martins et al., 2024), Aya 23 8B (Aryabumi et al., 2024) and Salamandra 7B Instruct (Gonzalez-Agirre et al., 2025). Inference of all models in this work is performed using the AlpacaEval repository (Li et al., 2023b), with the default decoding settings, i.e. temperature is 0.7, maximum tokens are set to 2048, and all models are loaded in `bfloat16`.

**Results (Zero-Shot)** We show our benchmark scores in Table 1. Aya Expanse leads the table, achieving a 60% win rate against GPT-4o Mini for the four X-AlpacaEval languages it supports (`por`, `deu`, `zho`, `hin`). While it was trained on significant synthetic data using multilingual experts

[1] To the best of our knowledge, we are the first to propose a cross-lingual open-ended generation benchmark, and our synthetic training dataset is among the few publicly available.

| Model | Avg | High-Res EU | | Med-Res EU | | Low-Res EU | | Non-EU | |
|---|---|---|---|---|---|---|---|---|---|
| | | por | deu | hun | lit | gle | mlt | zho | hin |
| Salamandra 7B Instruct | 6.44 | 8.64 | 8.27 | 5.08 | 9.51 | 5.63 | 4.95 | 5.24 | 4.23 |
| Aya 23 8B | 8.85 | 17.04 | 15.04 | 2.07 | 2.22 | 2.45 | 1.92 | 9.46 | 20.57 |
| EuroLLM 9B Instruct | 12.70 | 18.94 | 16.49 | 8.66 | 16.57 | 9.37 | 8.51 | 14.82 | 8.23 |
| Qwen 2.5 7B Instruct | 16.73 | 30.88 | 16.35 | 6.82 | 14.68 | 7.17 | 3.69 | 44.63 | 9.59 |
| Gemma 2 9B IT | 23.29 | 35.42 | 32.08 | 19.80 | **27.28** | 10.09 | **10.03** | 28.12 | 23.50 |
| Llama 3.1 8B Instruct | 24.36 | 40.28 | 35.72 | **23.07** | 20.74 | **13.20** | 8.47 | 31.21 | 22.22 |
| Aya Expanse 8B | **35.67** | **62.75** | **60.27** | 8.62 | 19.54 | 10.43 | 9.51 | **57.22** | **56.99** |

Table 1: Win rates against GPT-4o (Mini) on `XL-AlpacaEval`, as judged by GPT-4o. Languages are denoted by their ISO 639-3 codes.

| Model | Avg | por | deu | hun | lit | gle | mlt | zho | hin |
|---|---|---|---|---|---|---|---|---|---|
| Salamandra 7B Instruct | 4.45 | 3.32 | 2.47 | 2.16 | 3.71 | 7.49 | 8.00 | 6.09 | 2.37 |
| Aya 23 8B | 1.28 | 1.12 | -1.78 | -8.62 | 4.58 | 4.52 | 11.86 | 3.11 | -4.59 |
| EuroLLM 9B Instruct | 5.26 | -1.57 | -0.83 | 5.50 | 5.14 | 18.66 | 6.83 | 11.85 | -3.54 |
| Qwen 2.5 7B Instruct | -1.25 | -20.01 | 2.92 | 1.59 | 2.91 | 3.62 | 4.24 | 3.80 | -9.08 |
| Gemma 2 9B IT | -4.73 | -11.00 | -12.66 | -4.10 | -2.58 | 2.67 | -0.37 | 6.54 | -16.37 |
| Llama 3.1 8B Instruct | -10.55 | -23.84 | -18.01 | -7.96 | -8.14 | -1.75 | -2.69 | -0.08 | -21.92 |
| Aya Expanse 8B | -20.53 | -39.60 | -39.25 | -36.13 | -0.51 | -1.18 | -2.50 | -1.78 | -43.29 |

Table 2: Performance changes on using Reason-then-Translate: scores represent differences against win rates from Table 1. Strong positive improvements are shaded.

(Dang et al., 2024), it remains unclear whether its superiority stems from explicit cross-lingual tuning or implicit transfer. For other languages, Llama 3.1 and Gemma 2 yield comparable win rates ranging between 10% and 30%. We make two critical observations here. Firstly, except for Aya Expanse, most open LLMs trail significantly behind GPT-4o-Mini in cross-lingual generation, leaving much room for improvement. Secondly, the performance strongly correlates with the resourcefulness of the language. While Aya Expanse, Llama 3.1, and Gemma achieve win rates of 40% or higher for high-resource languages like `por`, `deu` and `zho`, performance drops to 20-30% for medium-resourced languages (`hun`, `lit`, `hin`) and 10% or less for lower-resourced languages like `gle` and `mlt`. This underscores the need for scalable pipelines for creating high-quality synthetic data for lower-resourced languages, in order to achieve more consistent model performance (see Table 6).

**Baseline: Reason-then-Translate** Previous works have proposed prompting LLMs to reason first in a high-resource language (e.g. English) and then translating into the target language (Qin et al., 2023; Huang et al., 2023; Wang et al., 2025). We call this approach 'reason-then-translate' and report results in Table 2. The outcomes are mixed: stronger multilingual models like Aya Expanse, Llama, and Gemma suffer significant performance drops. Manual inspection reveals these 7B models occasionally produce empty outputs, likely due to difficulty consistently following complex multi-step instructions — which aligns with prior studies having reported successful results with only larger (~70B) models. In contrast, weaker LLMs like EuroLLM and Salamandra, fine-tuned on English reasoning and MT data, can leverage this two-step approach to yield some gains over their poor initial scores. Overall, these results show that inducing cross-lingual capabilities in standard multilingual LLMs is challenging, and may not be resolved through prompting strategies alone.

## 4 The `XL-Instruct` Pipeline

To address this gap, we introduce the `XL-Instruct` pipeline (illustrated in Figure 1), designed to create cross-lingual synthetic instructions from a given seed corpus. At this point, we highlight two important considerations. First, unlike related work (Li et al., 2024a), we seed from English data instead of using the target language corpora directly. Given teacher LLMs are more proficient in English than in a low-resource language, we hypothesize that more high-quality, yet diverse, synthetic data could be generated in English. MT is employed only in the final stages, thereby minimizing noise propagation. Second, we exclusively utilize open-weight models with permissible licenses to generate syn-

thetic data, aligning with our objective of releasing a fully public open-source dataset.

We outline the four stages of the `XL-Instruct` pipeline in detail below:

1. **Stage 1: Reverse Instructions** Given a passage from our seed data, we ask a teacher LLM to generate an *instruction* for which this passage would be a valid response.

2. **Stage 2: Refinement**: Then, we ask the teacher to reword the question and response pairs to follow four manually defined criteria.

3. **Stage 3: Response Translation** Next, we translate the refined response to the target language, using one or more translation LLMs.

4. **Stage 4: Filtering** Finally, to ensure we use the highest quality targets, we use MT Quality Estimation (QE) models to filter the dataset for the best translations.

We conduct Supervised Fine-Tuning (SFT) on this synthetic dataset. We detail the minutiae of each stage below.

### 4.1 Stage 1: Question Generation

First, we sample an English passage from our seed corpus, CulturaX (Nguyen et al., 2024). Then, we ask a teacher LLM (Qwen 2.5 72B (Yang et al., 2024)) to produce an instruction *for which* the sampled sentence would be a valid response. Prompting in English allows us to leverage the teacher model directly without requiring the additional fine-tuning employed previously (Li et al., 2024a). This stage thus yields a synthetic English instruction, paired with the English seed passage as a response.

### 4.2 Stage 2: Refinement

Next, inspired by Self-Refine (Madaan et al., 2023), we use the teacher LLM (again, Qwen 2.5 72B) to refine the question-response pair further. Based on the most commonly occurring errors observed from manual inspection, we define four goals for the refinement process:

1. **Question Self-Sufficiency:** The question should be clear and unambiguous, and should not require any additional information or context to produce the given response.

2. **Response Naturalness:** The response should be 'natural-sounding' as an LLM output — in terms of fluency, neutrality objectivity, and consistency with the tone and style of LLM-generated responses.

3. **Response Precision:** The response should be topically relevant, factually accurate, and should directly answer the question. This can be thought of as analogous to precision since it tries to assess how much of the information contained in the response is relevant, necessary, and true.

4. **Response Informativeness:** The response should be informative and helpful, and must contain enough justification and explanation to make it useful to an end user. This is similar to recall, as it evaluates how much of the relevant and useful information for the response is actually provided.

We provide all four criteria and their definitions in a prompt and ask the teacher to refine the (question, response) pair. We also instruct the model to ensure the reworded response is grounded in the original one, and request it to not add any of its own knowledge — in order to avoid excessive teacher distillation and to ensure our targets are grounded in the seed data we use.

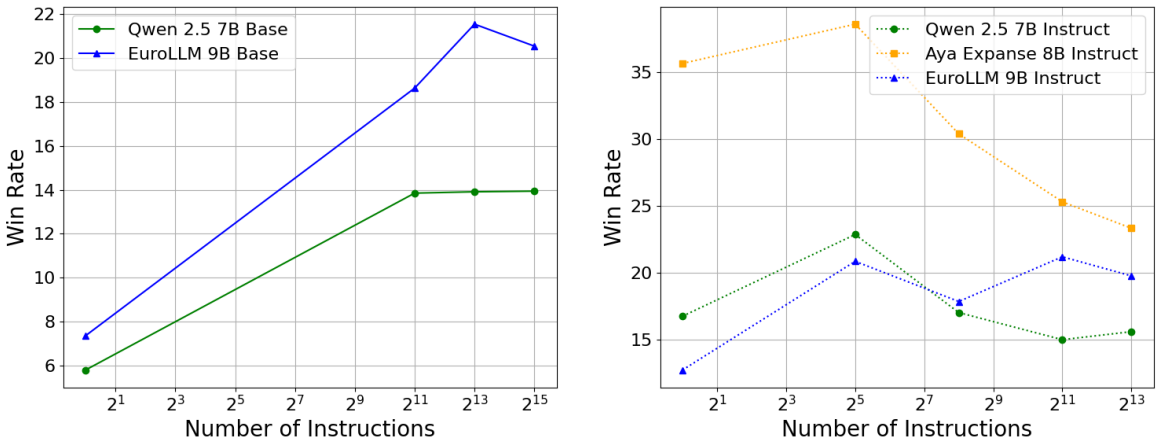
### 4.3 Stage 3: Response Translation

Now, we direct our focus towards converting the English question-response pair to a cross-lingual En-X one. Creating the cross-lingual instruction itself is easy — we simply add a prompt to "Respond in `{lang}`" where `{lang}` is the target language of interest. To create the target, the English response must be machine-translated into the target language. Since document-level MT by open LLMs is currently unreliable due to limited exploration, scarce datasets, and hallucination risks, we use sentence-level translation instead. We sentence-split using Segment Any Text (Frohmann et al., 2024) and generate translations in one of two ways:

1. **Naive:** In the vanilla case, we simply prompt an LLM for the translation.

2. **Best-of-k:** We obtain $k$ translations from $k$ different LLMs for each sentence, and choose the one with the best QE score.

For QE, we use the WMT'23 CometKiwi-XL model (Rei et al., 2023), which obtained state-of-the-art (SOTA) scores in the WMT 2023 QE Shared




(a) Base LLMs

(b) Instruction-Tuned LLMs

Figure 2: Performance on `XL-AlpacaEval` after SFT with `XL-Instruct` data of varying sizes. Y-axis scores reflect win rates against GPT-4o-Mini, averaged across 8 languages, with GPT-4o as the judge. X-axis instruction counts are shown on a log scale.

Task (Blain et al., 2023). For MT, we use EuroLLM 9B Instruct (Martins et al., 2024) in the 'naive' case due to its strong MT capabilities, while in the 'Best-of-k' setting, we set $k = 3$ and sample among EuroLLM 9B Instruct, Mistral Small 24B Instruct (Mistral AI Team, 2025), and Gemma2 27B Instruct (Team et al., 2024). Finally, to create the translated response, we substitute each sentence in the original response with its sentence-level translation. This helps us retain formatting (like paragraph separators, bullet points, etc.) that is critical to response quality.

### 4.4 Stage 4: Filtering

Finally, to ensure that we select high-quality targets during fine-tuning, we compute sentence-level QE scores using the WMT'23 CometKiwi-XL model, by comparing each source sentence in a given response and its translation. We average these QE scores across the entire passage to obtain the final passage-level score. Then, we sort all responses in descending order and filter the last 20% of the dataset – creating a final dataset of about 32K instructions. We provide the prompts used in each stage at `https://tinyurl.com/xli-prompts` (to be uploaded to GitHub on acceptance)

## 5 Experiments on `XL-AlpacaEval`: Boosting Cross-Lingual Generation

**Models** We conduct SFT of two base (EuroLLM 9B, Qwen2.5 7B) and three instruction-tuned (EuroLLM 9B Instruct, Qwen2.5 7B Instruct and Aya Expanse 8B) models. We choose EuroLLM and Qwen since they relatively underperform on the `XL-AlpacaEval` benchmark (Table 1), leaving significant scope for improvement. We also experiment with Aya Expanse since it leads the benchmark, and we are interested in seeing how much further it could be improved. Unfortunately, Aya Expanse does not have a base model released, so we are unable to experiment with it.

**Experimental Setting** We fine-tune all models for 1 epoch using low-rank adaptation (LoRA, Hu et al., 2022) with rank 8 matrices applied to query and value projections. We also tune the input and output embeddings. Training used a cosine learning rate scheduler with a peak learning rate of 1e-4 and 3% warmup steps. We employed `bf16` mixed-precision training with batch size 8, and fix the random seed as 1 for reproducibility. All experiments were run on 4 Nvidia GeForce RTX 3090 GPUs, each with 24 GB VRAM.

**Main Results** In Figure 2, we report win rates on `XL-AlpacaEval` on fine-tuning with various amounts of `XL-Instruct` data. We observe that for base LLMs, performance steadily improves with data scale. Qwen advances from a win rate of 5.8% to 13.89% against GPT-4o Mini, while EuroLLM achieves an even larger boost, going from 7.36% to as high as 21.89% on SFT with 8K instructions. We report language-specific scores in Table 6 and observe that while there are consistent gains for all languages, the largest gains are noted for the ones an LLM is pre-trained on. Since EuroLLM includes all 8 `XL-AlpacaEval` languages in its pre-

training, it observes large gains per language, leading to a much better overall average score. Qwen, which chiefly supports high-resource languages like Chinese, Portuguese and German, gains the most for these pairs but shows relatively smaller improvements for others. This suggests that while post-training with `XL-Instruct` can yield stable improvements across multiple languages, multilingual pre-training is crucial for best performance.

We also observe consistent improvements when fine-tuning instruction-tuned LLMs (Figure 2b). Unlike base LLMs, the saturation occurs sooner here—at around 2K instructions for EuroLLM-9B-Instruct and, only 32 instructions for the Qwen and Aya Expanse models! This is likely because the latter two models have also undergone Preference Optimization, and task-specific SFT at scale might lead to overfitting and deteriorated performance. EuroLLM, on the other hand, has only undergone SFT, and can therefore be trained for longer. Here too, one observes consistent and major gains across all languages (Table 6). These results are particularly noteworthy given the low training costs – low-rank fine-tuning with a few thousand instructions. Moreover, with only 32 examples, Aya Expanse achieves a win rate boost from ~57% to $\sim$ 65% for its supported languages of Portuguese, German, Hindi, and Chinese (Table 6). Lastly, we also show in the Appendix (Table 7) how `XL-Instruct` can also boost zero-shot cross-lingual performance, i.e. even for languages *not* included in SFT.

**Fine-Grained Evaluation** Beyond win rates that focus solely on pairwise comparisons, we are also interested in evaluating how well the produced cross-lingual generations improve on an absolute scale, on human-desired criteria. To achieve this, we take inspiration from recent works that define customised, task-specific metrics and use LLM-as-a-Judge for producing scores on a Likert scale – achieving significant correlations with human ratings on evaluation of tasks like summarization (Liu et al., 2023), retrieval (Upadhyay et al., 2024), story generation (Chiang and Lee, 2023), Machine Translation (Kocmi and Federmann, 2023b,a) and open-ended generation (Kim et al., 2023). In particular, Kim et al. (2023) showed that using clearly defined rubrics can result in Spearman correlations up to 0.87 with human preferences for open-ended generation. Inspired by this, we propose four criteria pertinent to the task of cross-lingual generation: Objectivity, Naturalness, Informativeness, and Precision. We define detailed rubrics for each metric and provide well-defined criteria for mapping output quality to scores on a scale of 1-5. We include these rubrics in the context of a prompt, and ask GPT-4o to score cross-lingual generations of EuroLLM 9B, EuroLLM 9B Instruct and EuroLLM 9B XL-Instruct (the best model from Figure 2a, which is fine-tuned with LoRA on 8K examples). We provide detailed evaluation prompts and rubrics at `https://tinyurl.com/xl-gen-eval` (to be uploaded to GitHub on acceptance).

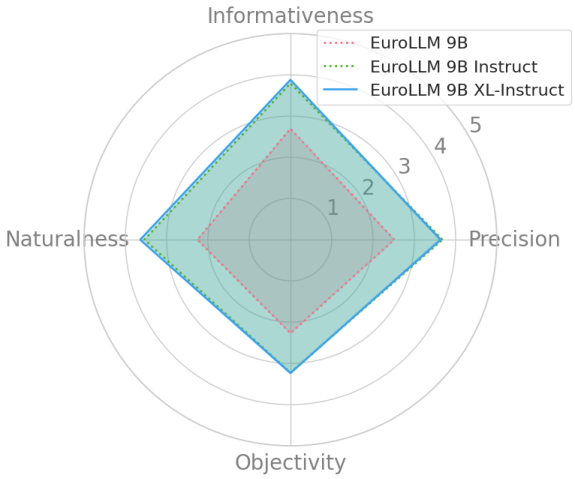



Figure 3: GPT-4o scores on fine-grained quality metrics (1–5 scale) are shown. For EuroLLM 9B XL-Instruct, we select the best model from Figure 2a. Despite being LoRA fine-tuned on only 8K pairs, it slightly outperforms EuroLLM 9B Instruct, which was fully fine-tuned on 2M examples. Scores are also tabulated in Table 8.

We show our results in Figure 3, which provides the macro-averaged scores per criterion and model. As expected, the raw EuroLLM 9B base model achieves the worst scores on all metrics, with the EuroLLM-9B-Instruct model performing substantially better. We note that the `XL-Instruct` model performs comparably to or marginally better than EuroLLM-9B-Instruct. This result is particularly impressive given the `XL-Instruct` baseline was trained using LoRA fine-tuning on only 8K synthetic samples, whereas the EuroLLM-9B-Instruct was fully fine-tuned on a mix of 2M human and synthetic examples. These results clearly demonstrate the effectiveness and high quality of the `XL-Instruct` dataset.

## 6 Experiments on m-AlpacaEval: Exploring Zero-Shot Transfer

Having seen task-specific improvements, we now seek to evaluate the zero-shot performance of mod-

| Model | Avg | zho | deu | hin | hun | gle | lit | mlt | por | eng |
|---|---|---|---|---|---|---|---|---|---|---|
| **EuroLLM 9B** | 0.73 | 1.19 | 1.47 | 0.70 | 0.14 | 0.14 | 0.65 | 0.31 | 1.25 | 35.59 |
| +XL-Instruct (best, 8K) | 6.10 | 10.77 | 11.40 | 4.47 | 2.53 | 3.60 | 3.77 | 2.49 | 9.76 | 51.35 |
| **EuroLLM 9B Instruct** | 8.94 | 13.38 | 11.99 | 8.13 | 4.81 | 5.65 | 6.68 | 6.78 | 14.12 | 55.58 |
| +XL-Instruct (best, 8K) | 15.55 | 19.57 | 18.30 | 13.03 | **10.38** | **14.12** | **16.76** | **14.13** | 18.11 | 59.44 |
| **Qwen 2.5 7B** | 2.04 | 10.40 | 1.52 | 0.98 | 0.24 | 0.03 | 0.45 | 0.29 | 2.39 | 46.93 |
| +XL-Instruct (best, 8K) | 5.66 | 20.43 | 9.53 | 2.23 | 0.65 | 0.18 | 1.60 | 0.29 | 10.33 | 55.92 |
| **Qwen 2.5 7B Instruct** | 11.47 | 45.29 | 10.53 | 5.71 | 0.97 | 0.99 | 3.22 | 1.63 | 23.39 | 75.16 |
| +XL-Instruct (best, 32) | 18.19 | 52.12 | 31.64 | 8.34 | 5.79 | 1.59 | 5.79 | 1.83 | 38.44 | 76.72 |
| **Aya Expanse 8B** | 29.90 | 58.21 | 56.91 | 56.68 | 1.11 | 1.02 | 3.04 | 2.94 | 59.29 | 76.26 |
| +XL-Instruct (best, 32) | 32.31 | 63.24 | 59.53 | 63.22 | 2.21 | 0.78 | 5.85 | 3.66 | 60.01 | 77.70 |

Table 3: Win Rates of LLMs and their XL-Instruct fine-tuned counterparts on m-AlpacaEval against GPT-4o-Mini, with GPT-4o as the judge. For each model, we choose the best cross-lingual performing baseline from Figure 2 and evaluate transfer on m-AlpacaEval. Consistent improvement across all models and pairs shows strong zero-shot transfer from cross-lingual tuning, for both multilingual and English-only generation. Best scores are bolded and cells are highlighted proportionate to performance gain.

els fine-tuned with XL-Instruct on multilingual and English open-ended generation, since these are arguably the more common use cases of LLMs. For this purpose, we first construct the m-AlpacaEval benchmark by machine translating the AlpacaEval test set into our 8 languages of interest, following related efforts to create m-ArenaHard (Dang et al., 2024). We use GPT-4o for translation of the prompts. The evaluation setup is similar to XL-AlpacaEval, wherein GPT-4o-Mini is the reference model and GPT-4o is the judge.

We show our results in Table 3, for the base and instruct LLMs from Figure 2, along with their best-performing XL-Instruct-tuned counterparts. We observe significant and consistent zero-shot transfer across all models and languages. For multilingual generation, the gains are strongest for the languages a model is pretrained on, similar to our observations for cross-lingual generation. This is particularly evident in the Qwen and Aya models. EuroLLM Instruct, on the other hand, achieves stable performances across all languages and relatively strongest win rates for the lower-resourced languages. Interestingly, we also note consistent gains in English-only generation, despite there being no English responses on the target side! This suggests that all of these models, trained heavily on English, can learn preferred response structure and formatting from cross-lingual tuning. These results are quite encouraging, since they suggest cross-lingual fine-tuning need not come at the cost of standard 'monolingual' generation performance – on the contrary, it can result in further boosts.

## 7 Conclusion

In this work, we propose data resources for advancing cross-lingual open-ended generation—loosely defined as a task in which the query and the desired (open-ended) response are in different languages. This can be viewed as a distinct yet crucial subtask of multilingual generation. While cross-lingual generation may also include more complex scenarios, such as providing context in one language while the query and response are in another (or even multiple) languages, we focus here on the simpler scenario: queries posed in English with responses required in one of eight target languages – which includes high, medium, and low-resource EU and non-EU languages.

With this goal in mind, we make three key contributions. First, we introduce the XL-AlpacaEval benchmark to evaluate the current state of open LLMs, and report poor performances and significant gaps against GPT-4o-Mini. Second, we propose the XL-Instruct technique, and show that this synthetic data can substantially boost cross-lingual performance, both in terms of win rates and fine-grained quality metrics. Third, we show that it exhibits strong zero-shot transfer to monolingual generation, both in English and beyond. Based on these results, we strongly encourage researchers to post-train their multilingual LLMs with XL-Instruct data, and shall publicly release our versions of XL-Instruct and XL-AlpacaEval datasets to support this.

## 8 Limitations

There has been some concern in the literature that iterative training on synthetic data could eventually lead to model collapse (Shumailov et al., 2024). Like any other synthetic data technique, `XL-Instruct` could also share similar risks, especially since its seed data is sourced from the Web. We conduct fine-grained evaluation along various quality metrics, including Objectivity and Naturalness, to ensure that our fine-tuned models continue producing neutral, unbiased, and natural-sounding outputs and observe performance comparable to a fully instruction-tuned model fine-tuned on human-curated data – which helps alleviate such concerns. We also note that later works have shown that mixing synthetic with human-generated data could avoid model collapse (Seddik et al., 2024; Gerstgrasser et al., 2024). While we have been unable to explore the interaction of `XL-Instruct` with data from other tasks and sources in the scope of this work, we feel this would be an interesting direction of future research to both guard against collapse and to study inter-task transfer.

## References


Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Jon Ander Campos, Yi Chern Tan, and 1 others. 2024. Aya 23: Open weight releases to further multilingual progress. *arXiv preprint arXiv:2405.15032*.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, and 12 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *arXiv preprint arXiv:2204.05862*.

Frederic Blain, Chrysoula Zerva, Ricardo Ribeiro, Nuno Miguel Guerreiro, Diptesh Kanojia, José GC de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, and 1 others. 2023. Findings of the wmt 2023 shared task on quality estimation. In *Eight conference on machine translation*, pages 629–653. Association for Computational Linguistics.

Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Andrey Kutuzov, Barry Haddow, and Kenneth Heafield. 2024. Monolingual or multilingual instruction tuning: Which makes a better alpaca. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1347–1356, St. Julian's, Malta. Association for Computational Linguistics.

Yongrui Chen, Haiyun Jiang, Xinting Huang, Shuming Shi, and Guilin Qi. 2023. Dog-instruct: Towards premium instruction-tuning data via text-grounded instruction wrapping. *arXiv preprint arXiv:2309.05447*.

Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631, Toronto, Canada. Association for Computational Linguistics.

Wei-Lin Chiang and 1 others. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. *See https://vicuna.lmsys.org (accessed 22 January 2025)*, 2.3:6.

John Dang, Shivalika Singh, Daniel D'souza, Arash Ahmadian, Alejandro Salamanca, Madeline Smith, Aidan Peppin, Sungjin Hong, Manoj Govindassamy, Terrence Zhao, and 1 others. 2024. Aya expanse: Combining research breakthroughs for a new multilingual frontier. *arXiv preprint arXiv:2412.04261*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. 2024. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36.

Markus Frohmann, Igor Sterner, Ivan Vulić, Benjamin Minixhofer, and Markus Schedl. 2024. Segment any text: A universal approach for robust, efficient and adaptable sentence segmentation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11908–11941, Miami, Florida, USA. Association for Computational Linguistics.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. 2024. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*.

Xinyang Geng, Arnav Gudibande, Hao Liu, Eric Wallace, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. Koala: A dialogue model for academic research. Blog post.

Matthias Gerstgrasser, Rylan Schaeffer, Apratim Dey, Rafael Rafailov, Tomasz Korbak, Henry Sleight, Rajashree Agrawal, John Hughes, Dhruv Bhandarkar Pai, Andrey Gromov, Dan Roberts, Diyi Yang, David L. Donoho, and Sanmi Koyejo. 2024. Is model collapse inevitable? breaking the curse of recursion by accumulating real and synthetic data. In *First Conference on Language Modeling*.

Aitor Gonzalez-Agirre, Marc Pàmies, Joan Llop, Irene Baucells, Severino Da Dalt, Daniel Tamayo, José Javier Saiz, Ferran Espuña, Jaume Prats, Javier Aula-Blasco, Mario Mina, Adrián Rubio, Alexander Shvets, Anna Sallés, Iñaki Lacunza, Iñigo Pikabea, Jorge Palomar, Júlia Falcão, Lucía Tormo, and 4 others. 2025. Salamandra technical report. *Preprint*, arXiv:2502.08489.

Srishti Gureja, Lester James V Miranda, Shayekh Bin Islam, Rishabh Maheshwary, Drishti Sharma, Gusti Winata, Nathan Lambert, Sebastian Ruder, Sara Hooker, and Marzieh Fadaee. 2024. M-rewardbench: Evaluating reward models in multilingual settings. *arXiv preprint arXiv:2410.15522*.

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2023. Unnatural instructions: Tuning language models with (almost) no human labor. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14409–14428, Toronto, Canada. Association for Computational Linguistics.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12365–12394, Singapore. Association for Computational Linguistics.

Vivek Iyer, Bhavitvya Malik, Pavel Stepachev, Pinzhen Chen, Barry Haddow, and Alexandra Birch. 2024a. Quality or quantity? on data scale and diversity in adapting large language models for low-resource translation. In *Proceedings of the Ninth Conference on Machine Translation*, pages 1393–1409, Miami, Florida, USA. Association for Computational Linguistics.

Vivek Iyer, Bhavitvya Malik, Wenhao Zhu, Pavel Stepachev, Pinzhen Chen, Barry Haddow, and Alexandra Birch. 2024b. Exploring very low-resource translation with LLMs: The University of Edinburgh's submission to AmericasNLP 2024 translation task. In *Proceedings of the 4th Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP 2024)*, pages 209–220, Mexico City, Mexico. Association for Computational Linguistics.

Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoo Yun, Seongjin Shin, Sungdong Kim, James Thorne, and 1 others. 2023. Prometheus: Inducing fine-grained evaluation capability in language models. In *The Twelfth International Conference on Learning Representations*.

Tom Kocmi and Christian Federmann. 2023a. Gemba-mqm: Detecting translation quality error spans with gpt-4. In *Proceedings of the Eighth Conference on Machine Translation*, Singapore. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023b. Large language models are state-of-the-art evaluators of translation quality. In *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*, pages 193–203, Tampere, Finland. European Association for Machine Translation.

Abdullatif Köksal, Timo Schick, Anna Korhonen, and Hinrich Schütze. 2023. Longform: Effective instruction tuning with reverse instructions. *arXiv preprint arXiv:2304.08460*.

Abdullatif Köksal, Marion Thaler, Ayyoob Imani, Ahmet Üstün, Anna Korhonen, and Hinrich Schütze. 2024. Muri: High-quality instruction tuning datasets for low-resource languages via reverse instructions. *arXiv preprint arXiv:2409.12958*.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, and 1 others. 2024. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023. Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 318–327, Singapore. Association for Computational Linguistics.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. T\" ulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Chong Li, Wen Yang, Jiajun Zhang, Jinliang Lu, Shaonan Wang, and Chengqing Zong. 2024a. X-instruction: Aligning language model in low-resource languages with self-curated cross-lingual instructions. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 546–566, Bangkok, Thailand. Association for Computational Linguistics.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Omer Levy, Luke Zettlemoyer, Jason Weston, and Mike Lewis. 2023a. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023b. Alpacaeval: An automatic evaluator of instruction-following models.

Zhecheng Li, Yiwei Wang, Bryan Hooi, Yujun Cai, Naifan Cheung, Nanyun Peng, and Kai-wei Chang. 2024b. Think carefully and check again! meta-generation unlocking llms for low-resource cross-lingual summarization. *arXiv preprint arXiv:2410.20021*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, and 1 others. 2023. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems*, 36:46534–46594.

Kelly Marchisio, Wei-Yin Ko, Alexandre Berard, Théo Dehaze, and Sebastian Ruder. 2024. Understanding and mitigating language confusion in LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 6653–6677, Miami, Florida, USA. Association for Computational Linguistics.

Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M Guerreiro, Ricardo Rei, Duarte M Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, and 1 others. 2024. EuroLLM: Multilingual language models for Europe. *arXiv preprint arXiv:2409.16235*.

Mistral AI Team. 2025. Mistral small 3: Apache 2.0, 81% mmlu, 150 tokens/s. https://mistral.ai/news/mistral-small-3. Accessed: 2025-03-17.

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)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2024. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 4226–4237, Torino, Italia. ELRA and ICCL.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1703–1714, Online. Association for Computational Linguistics.

Libo Qin, Qiguang Chen, Fuxuan Wei, Shijue Huang, and Wanxiang Che. 2023. Cross-lingual prompting: Improving zero-shot chain-of-thought reasoning across languages. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2695–2709, Singapore. Association for Computational Linguistics.

Leonardo Ranaldi and Giulia Pucci. 2023. Does the English matter? elicit cross-lingual abilities of large language models. In *Proceedings of the 3rd Workshop on Multi-lingual Representation Learning (MRL)*, pages 173–183, Singapore. Association for Computational Linguistics.

Ricardo Rei, Nuno M Guerreiro, José Pombal, Daan van Stigt, Marcos Treviso, Luisa Coheur, José GC de Souza, and André FT Martins. 2023. Scaling up cometkiwi: Unbabel-ist 2023 submission for the quality estimation shared task. *arXiv preprint arXiv:2309.11925*.

Mohamed El Amine Seddik, Suei-Wen Chen, Soufiane Hayou, Pierre Youssef, and Merouane Abdelkader DEBBAH. 2024. How bad is training on synthetic data? a statistical analysis of language model collapse. In *First Conference on Language Modeling*.

Ilia Shumailov, Zakhar Shumaylov, Yiren Zhao, Nicolas Papernot, Ross Anderson, and Yarin Gal. 2024. Ai models collapse when trained on recursively generated data. *Nature*, 631(8022):755–759.

Vaibhav Singh, Amrith Krishna, Karthika NJ, and Ganesh Ramakrishnan. 2024. A three-pronged approach to cross-lingual adaptation with multilingual llms. *arXiv preprint arXiv:2406.17377*.

Guijin Son, Dongkeun Yoon, Juyoung Suk, Javier Aula-Blasco, Mano Aslan, Vu Trong Kim, Shayekh Bin Islam, Jaume Prats-Cristià, Lucía Tormo-Bañuelos, and Seungone Kim. 2024. Mm-eval: A multilingual meta-evaluation benchmark for llm-as-a-judge and reward models. *arXiv preprint arXiv:2410.17578*.

Eshaan Tanwar, Subhabrata Dutta, Manish Borthakur, and Tanmoy Chakraborty. 2023. Multilingual LLMs are better cross-lingual in-context learners with alignment. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6292–6307, Toronto, Canada. Association for Computational Linguistics.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Nick Craswell, and Jimmy Lin. 2024. Umbrela: Umbrela is the (open-source reproduction of the) bing relevance assessor. *arXiv preprint arXiv:2406.06519*.

Teng Wang, Zhenqi He, Wing-Yin Yu, Xiaojin Fu, and Xiongwei Han. 2025. Large language models are good multi-lingual learners : When LLMs meet cross-lingual prompts. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 4442–4456, Abu Dhabi, UAE.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023a. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Yue Wang, Xinrui Wang, Juntao Li, Jinxiong Chang, Qishen Zhang, Zhongyi Liu, Guannan Zhang, and Min Zhang. 2023b. Harnessing the power of david against goliath: Exploring instruction data generation without using closed-source models. *arXiv preprint arXiv:2308.12711*.

Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. 2024. Self-preference bias in llm-as-a-judge. *arXiv preprint arXiv:2410.21819*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

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)*, pages 7025–7046, Bangkok, Thailand. Association for Computational Linguistics.

Lianmin Zheng and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

# A Data

## A.1 The XL-AlpacaEval Benchmark

Here we provide some additional details on the creation of the XL-AlpacaEval benchmark, which has 797 cross-lingual prompts in total, and currently supports 11 languages - the 8 languages used for the primary experiments in this work (Chinese, German, Hindi, Hungarian, Irish, Lithuanian, Maltese and Portuguese) and 3 additional languages (French, Finnish and Turkish) which we use for zero-shot evaluation in future sections. It is trivial to extend it to other languages – one simply has to run a script to append cross-lingual generation instructions (Section A.1.2) to our filtered AlpacaEval test set (Section A.1.1) and such extensions are being planned as a part of future work.

### A.1.1 Manual Verification

Before creating our cross-lingual benchmark, we conduct a rigorous stage of manual verification to ensure that the prompts are suitable for answering cross-lingually. In Table 4, we show the prompts we removed from AlpacaEval that were too culturally specific (for instance, prompt 183) or tailored towards eliciting an English response (prompts 350 and 714). In the latter, we felt mandating a non-English response might make evaluating a "correct" response challenging. In other cases where the prompt simply requested a response in English, we replaced with a generic templated variable `{language}` for downstream substitution with the name of the desired target language. This leaves us with a total of 797 prompts. It is important to note that as far as possible, we tried to keep complex multi-step, multilingual prompts in our evaluation set, and only removed cases that were clearly invalid – in keeping with the goal of this work to build robust cross-lingual models.

### A.1.2 Generation prompts

Next, we randomly sample prompts from a list of cross-lingual generation instructions (given in Table 5), and append it to each prompt in the filtered test set from the previous stage. To add further diversity to the instructions in the benchmark, we remove the word "language" from the prompts given in Table 5 – thus converting "Answer in German language" to "Answer in German". This leads to the creation of the final XL-AlpacaEval benchmark.

## A.2 License

The `XL-AlpacaEval` dataset, which is derived from the AlpacaEval dataset, is released under a CC-by-NC 4.0 license following its predecessor. This dataset is thus primarily intended for use in research contexts. In contrast, the `XL-Instruct` dataset, which is provided as a training dataset, is derived from the CulturaX corpus – which in turn sources from the mC4 (Xue et al., 2021) and OSCAR (Ortiz Suárez et al., 2020). mC4 is released under an ODC-BY license, and OSCAR is released under CC0 no rights reserved. Hence, `XL-Instruct` can be used in both commercial and research contexts, as long as the corresponding licenses are respected.

# B Experiments

## B.1 XL-AlpacaEval Results

**Full Results** In Table 6, we show the complete language-wise results for each base and instruct model we tuned on varying sizes of XL-Instruct data. Models like EuroLLM and Qwen continue improving until 8K-32K instructions, with gains diminishing in the last 24K instructions. This is likely because we sort the instructions in order of translation quality, and sample them accordingly, reducing the gains. It is possible that improving the translation quality further could result in larger gains. For preference-optimized (PO'ed) instruction-tuned models, performance saturates at 32 instructions, and 2K instructions with non-PO'ed models like EuroLLM 9B Instruct. The largest gains across all models are consistently for the languages included during pretraining – for instance, Qwen 7B improves on Chinese win rates from 12.62 to 34.29 and in Portuguese from 9.82 to 27.13, suggesting the criticality of this stage in building multilingual LLMs.

**Zero-Shot Results** We show in Table 7 evaluations on zero-shot performance after fine-tuning with `XL-Instruct`. We choose French, Finnish, and Turkish, 3 languages the EuroLLM model is pre-trained on, and observe huge gains in win rates, largely outperforming even the EuroLLM 9B Instruct model. This shows that even if done only for a few languages, `XL-Instruct` can still result in significant transfer that improves performance in others. We hypothesize that this is likely because the model is able to learn formatting, response structure, etc. from this process, which

| Prompt ID | Prompt Text |
|---|---|
| 183 | Write a story about Anakin Skywalker encountering a Jedi who speaks and acts like a 1920s British aristocrat. |
| 200 | Write "Test" |
| 350 | I'm an English speaker trying to learn Japanese Kanji using mnemonics. Mnemonics for Kanji are created from the primitives that make them up. The Kanji for Tax has the primitives wheat and devil, so an example would be, "Taxes are like the devil taking away your hard earned wheat". Can you create a mnemonic for the Kanji meaning Wish that has the primitives clock and heart? |
| 458 | Give me a list of 5 words where the letters of the words are in alphabetical order. One example: "doors". "d" comes before "o", "o" comes before "r", and "r" comes before "s". |
| 476 | Rewrite the given text and correct grammar, spelling, and punctuation errors. If you'd told me year ago that today I would finish a marathon, I would of laughed. Your support had a huge affect on me! |
| 495 | During writing, we added an asterisk for the word that did not come to mind. You will need to provide several examples to demonstrate all the words that can be used in the sentence instead of the asterisk. |
| 635 | Correct the transcription of an excerpt containing errors. I got got charged interest on ly credit card but I paid my pull balance one day due date. I not missed a pavement year yet. Man you reverse the interest charge? |
| 662 | You should capitalize the sentence according to the guide. Guide: Every other letter alternates between lower case and upper case. Sentence: A giant spider blocks your path. |
| 663 | Create alliterations by finding synonyms for words in the given sentence. David wears a hat everyday. |
| 714 | Rewrite the text and correct the spelling errors. It solves problems comon and uniqe to every team. |

Table 4: Culturally specific prompts removed from the AlpacaEval dataset.

| Prompts | |
|---|---|
| Answer in {} language | Output an answer in {} language |
| Generate your answer in {} language | Respond in {} language |
| Produce an answer in {} language | Please write in {} language |

Table 5: Cross-Lingual Generation Instructions

also supports the boosts in English generation one observes in Table 3.

**Fine-grained Evaluation** Finally, we also tabulate the macro-averaged GPT-4o scorings on Precision, Informativeness, Naturalness, and Objectivity metrics (evaluation prompts and rubrics available at `https://tinyurl.com/xl-gen-eval`). As noted previously, the EuroLLM 9B model performs the worst, but the `XL-Instruct` model performs comparably or slightly better than EuroLLM 9B Instruct, indicating the efficacy of our data generation method.

### B.2 Comparison with X-Instruction

We also compare the efficacy of our method with the most similar work to ours and the only cross-lingual open-ended generation dataset we are aware of: X-Instruction (Li et al., 2024a). In this work, the authors prompt a teacher LLM for reverse instructions directly in the low-resource language. Given this initially results in instructions of much poorer quality due to poor teacher capabilites, they also conduct iterative scoring and refinement to improve quality, achieving impressive results. They also release their dataset publicly at `https://huggingface.co/datasets/James-WYang/X-Instruction`.

We use the Hindi, Finnish, and Turkish splits of this dataset since these are supported by EuroLLM and are also available in X-Instruction. We also generate XL-Instruct data in these languages, by redoing the XL-Instruct pipeline (Section 4) from Stage 3 (Response Translation) for these languages. We LoRA fine-tune EuroLLM 9B on various X-Instruction and XL-Instruct datasets. For the former, we use both the entire 1M sized dataset available for these languages (in total), and a 40K instructions subset which is more comparable to our XL-Instruct baselines. For XL-Instruct, we train two baselines – one trained on 'naive' translations

| Model | Avg | zho | deu | hin | hun | gle | lit | mlt | por |
|---|---|---|---|---|---|---|---|---|---|
| **EuroLLM 9B** | 7.36 | 8.97 | 9.96 | 4.49 | 4.13 | 6.09 | 9.94 | 4.66 | 10.61 |
| +2K instructions | 18.63 | 18.77 | 23.65 | 13.22 | 13.70 | 16.03 | 25.48 | 14.75 | 23.47 |
| +8K instructions | **21.54** | 20.98 | **26.76** | **16.26** | **17.27** | 20.99 | **28.52** | **15.72** | **25.81** |
| +32K instructions | 20.54 | **21.24** | 24.07 | 15.26 | 17.11 | **21.08** | 28.09 | 15.64 | 21.79 |
| **EuroLLM 9B Instruct** | 12.70 | 14.82 | 16.49 | 8.23 | 8.66 | 9.37 | 16.57 | 8.51 | 18.94 |
| +32 instructions | 20.84 | 23.52 | 22.96 | 13.10 | 17.37 | 17.10 | 25.61 | 21.30 | 25.79 |
| +256 instructions | 17.83 | 21.13 | 21.73 | 12.90 | 14.05 | 13.25 | 21.13 | 15.32 | 23.11 |
| +2K instructions | **21.18** | **23.62** | **24.39** | 14.49 | **16.63** | 20.17 | **27.87** | **18.02** | **24.22** |
| +8K instructions | 19.75 | 23.10 | 22.65 | **14.50** | 14.97 | **20.55** | 26.92 | 15.34 | 19.96 |
| **Qwen 2.5 7B** | 5.80 | 12.62 | 6.36 | 3.40 | 2.73 | 4.50 | 4.33 | 2.62 | 9.82 |
| +2K instructions | 13.85 | 33.64 | 18.37 | 6.67 | **6.50** | 5.00 | 10.73 | **3.63** | 26.22 |
| +8K instructions | 13.91 | 34.22 | **19.80** | 6.61 | 6.28 | 3.92 | 10.22 | 3.07 | **27.13** |
| +32K instructions | **13.94** | **34.29** | 18.88 | **6.88** | 5.72 | **5.44** | **10.77** | 3.36 | 26.18 |
| **Qwen 2.5 7B Instruct** | 16.73 | 44.63 | 16.35 | 9.59 | 6.82 | 7.17 | 14.68 | 3.69 | 30.88 |
| +32 instructions | **22.85** | **50.16** | **31.66** | **12.36** | **12.52** | **8.66** | **19.40** | **4.91** | **43.10** |
| +256 instructions | 17.00 | 38.04 | 22.45 | 9.46 | 7.85 | 5.39 | 15.86 | 4.02 | 32.92 |
| +2K instructions | 14.97 | 36.17 | 18.95 | 8.44 | 7.14 | 5.06 | 12.02 | 3.02 | 28.92 |
| +8K instructions | 15.57 | 42.74 | 18.85 | 8.32 | 6.54 | 4.41 | 11.99 | 3.49 | 28.19 |
| **Aya Expanse 8B** | 35.67 | 57.22 | 60.27 | 56.99 | 8.62 | 10.43 | 19.54 | 9.51 | 62.75 |
| +32 instructions | **38.61** | **64.08** | **65.07** | **59.76** | **10.71** | **11.72** | **21.57** | **10.70** | **65.28** |
| +256 instructions | 30.39 | 55.65 | 52.93 | 44.50 | 6.51 | 6.45 | 17.90 | 6.07 | 53.10 |
| +2K instructions | 25.30 | 41.84 | 46.43 | 37.03 | 6.77 | 4.55 | 15.94 | 3.75 | 46.12 |
| +8K instructions | 23.32 | 43.16 | 42.23 | 28.19 | 5.44 | 6.00 | 15.94 | 4.91 | 40.72 |

Table 6: Full language-wise Win Rates against GPT-4o-Mini on XL-AlpacaEval, after LoRA fine-tuning on varying sizes of XL-Instruct data on different LLMs. GPT-4o is the judge. The best scores per model are highlighted in bold.

(ie. using only EuroLLM 9B Instruct) and another using the 'best-of-3' method (refer to Section 4.3 for a detailed explanation).

We see that both XL-Instruct baselines significantly outperform X-Instruction, with our best model achieving a 70.68% improvement over the latter – showcasing the relative superiority of our pipeline. This also suggests it might be more effective to prompt a teacher model in English due to inherently superior capabilities, and we hypothesize it might allow for greater quality and diversity in responses, as well as allow for more complex operations – like refinement following specifically defined, custom criteria.

### B.3 Ablations

Lastly, we conduct an ablation to verify the importance of the translation selection strategy. Given the cross-lingual part of the dataset mainly comes from Machine Translations, and translations can be quite noisy, we experiment with 2 MT techniques, 'naive' and 'best-of-3' responses. We also include a 'random' sampling strategy, where random responses are chosen for subsampling, regardless of MT quality. We fine-tune the EuroLLM 9B and EuroLLM 9B Instruct models using 8K and 32 instructions respectively, which are respectively the optimal SFT data sizes for each model (check Figure 2).

For the instruct model, 'best of 3' introduces significant improvements over naive or random sampling strategies, taking the average win rate from 18.55 to 20.84. This is likely because at the tiny scale of 32 instructions, target response quality matters hugely and significantly impacts performance. For EuroLLM 9B, which is fine-tuned on 8K instructions, performance still improves for most languages with the best-of-3 technique. The only cases where it drops are for the least-resourced languages like Irish and Maltese, which makes the average score much lower. It is possible the

| Model | Avg | French | Finnish | Turkish |
|---|---|---|---|---|
| EuroLLM 9B | 7.80 | 9.69 | 9.78 | 3.94 |
| EuroLLM 9B Instruct | 14.08 | 19.39 | 14.05 | 8.81 |
| EuroLLM 9B XL-Instruct (best) | **20.62** | **25.8** | **22.72** | **13.33** |

Table 7: Fine-tuning with XL-Instruct yields zero-shot boosts in cross-lingual performance. Scores represent zero-shot win rates of various LLMs against GPT-4o-Mini, with GPT-4o as a judge. For the XL-Instruct baseline, we use the best-performing model from Figure 2.

| Model | Average | Precision | Informativeness | Naturalness | Objectivity |
|---|---|---|---|---|---|
| EuroLLM 9B | 2.43 | 2.52 | 2.69 | 2.25 | 2.27 |
| EuroLLM 9B Instruct | 3.56 | 3.68 | 3.80 | 3.54 | 3.23 |
| EuroLLM 9B XL-Instruct (best) | **3.60** | **3.63** | **3.88** | **3.64** | **3.24** |

Table 8: Performance of EuroLLM 9B models evaluated on Precision, Informativeness, Naturalness, and Objectivity, with the average of these metrics.

CometKiwi model we use for Quality Estimation is not very well-suited for such low-resource languages. As a result, we hypothesize that best-of-3 might sometimes end up choosing a worse translation than the naive method – which uses EuroLLM, a model known to have strong MT capabilities for all these languages.

### B.4 On AI assistant usage

AI assistants were used to aid the programming and writing process in this work. The authors have taken care to ensure its use was merely superficial and not pivotal. For coding, it was used to create helper functions for preprocessing, and resolve bugs. During writing, it was used to aid in constructing LaTeX tables, plot graphs, fix grammar, etc.

| Model | Avg | Finnish | Hindi | Turkish |
|---|---|---|---|---|
| EuroLLM 9B + X-Instruction (full 1M) | 9.73 | 14.35 | 7.22 | 7.63 |
| EuroLLM 9B + X-Instruction (40K) | 10.44 | 13.69 | 8.76 | 8.86 |
| EuroLLM 9B + XL-Instruct (naive, 40K) | 12.06 | 15.3 | 10.9 | 9.98 |
| EuroLLM 9B + XL-Instruct (best of 3, 40K) | **17.82** | **23.15** | **15.8** | **14.52** |

Table 9: Performances of EuroLLM 9B models fine-tuned on X-Instruction and XL-Instruct data

| Model | Avg | zho | deu | hin | hun | gle | lit | mlt | por |
|---|---|---|---|---|---|---|---|---|---|
| **EuroLLM 9B** | 7.36 | 8.97 | 9.96 | 4.49 | 4.13 | 6.09 | 9.94 | 4.66 | 10.61 |
| +8K instructions (random) | 22.69 | 22.66 | 25.71 | 15.59 | 18.45 | 22.40 | 29.52 | 20.50 | 26.72 |
| +8K instructions (naive) | 21.17 | 20.26 | 24.63 | 15.53 | 16.68 | 21.96 | 28.33 | 18.24 | 23.75 |
| +8K instructions (best of 3) | 21.54 | 20.98 | 26.76 | 16.26 | 17.27 | 20.99 | 28.52 | 15.72 | 25.81 |
| **EuroLLM 9B Instruct** | 12.70 | 14.82 | 16.49 | 8.23 | 8.66 | 9.37 | 16.57 | 8.51 | 18.94 |
| +32 instructions (random) | 18.49 | 22.21 | 22.69 | 12.70 | 15.18 | 15.35 | 21.87 | 14.12 | 23.79 |
| +32 instructions (naive) | 18.55 | 22.20 | 20.16 | 12.18 | 13.70 | 15.14 | 23.64 | 17.55 | 23.84 |
| +32 instructions (best of 3) | 20.84 | 23.52 | 22.96 | 13.10 | 17.37 | 17.10 | 25.61 | 21.30 | 25.79 |

Table 10: Ablations of the strategy for selecting response translations for the EuroLLM 9B and EuroLLM 9B Instruct models.