LLMs Beyond English: Scaling the Multilingual Capability of LLMs with Cross-Lingual Feedback

Wen Lai[1,2,3]*, **Mohsen Mesgar**[4], **Alexander Fraser**[1,3]

[1]School of Computation, Information and Technology, TUM, Germany
[2]Center for Information and Language Processing, LMU Munich, Germany
[3]Munich Center for Machine Learning, Germany
[4]Bosch Center for Artificial Intelligence, Renningen, Germany
lavine@cis.lmu.de mohsen.mesgar@bosch.com alexander.fraser@tum.de

Abstract

To democratize large language models (LLMs) to most natural languages, it is imperative to make these models capable of *understanding* and *generating* texts in many languages, in particular low-resource ones. While recent multilingual LLMs demonstrate remarkable performance in such capabilities, these LLMs still support a limited number of human languages due to the lack of training data for low-resource languages. Moreover, these LLMs are not yet aligned with human preference for downstream tasks, which is crucial for the success of LLMs in English. In this paper, we introduce xLLaMA-100 and xBLOOM-100 (collectively **xLLMs-100**), which scale the multilingual capabilities of LLaMA and BLOOM to 100 languages. To do so, we construct two datasets: a multilingual instruction dataset including 100 languages, which represents the largest language coverage to date, and a cross-lingual human feedback dataset encompassing 30 languages. We perform multilingual instruction tuning on the constructed instruction data and further align the LLMs with human feedback using the DPO algorithm on our cross-lingual human feedback dataset. We evaluate the multilingual understanding and generating capabilities of xLLMs-100 on five multilingual benchmarks. Experimental results show that xLLMs-100 consistently outperforms its peers across the benchmarks by considerable margins, defining a new state-of-the-art multilingual LLM that supports 100 languages[1].

1 Introduction

Despite the impressive improvements have been made recently, the majority of large language models (LLMs), such as LLaMA-2 (Touvron et al., 2023), LLaMA-3 (AI@Meta, 2024) and BLOOM (BigScience et al., 2022), are predominantly trained on English texts and support only a limited number of non-English languages. For instance, while LLaMA-2 and BLOOM support 25 and 46 languages, respectively, their performance varies significantly across different languages. However, there are currently more than 7,000 languages spoken in the world[2] and only a few of them are used for training LLMs. Scaling LLMs' multilingual capabilities is challenging due to the scarcity of multilingual instruction data available for fine-tuning, particularly for low-resource languages.

The primary advantage of LLMs lies in their ability to learn task execution, in particular mapping an input text to an output text, through a textual instruction. A task instruction, input, and output text are assumed to be in the same language. However, these elements can be in different languages for many downstream tasks, which is the most intuitive manifestation of the multilingual capabilities of LLMs. We consider two types of multilingual capability in LLMs. First, when the instructions for LLMs are expressed in different languages, LLMs should understand these instructions and generate a correct output. We refer to this feature of LLMs as the *understanding capability*. Second, LLMs should be able to generate the correct response in the target language and perform consistently well on (almost) all languages when a fixed language (e.g., English) is used as the instruction language. We name this capability of LLMs the *generating capability*.

We categorize the previous work on scaling the multilingual capability of LLMs in two groups. The first group includes approaches that continue training the LLMs using as much of the training corpora as possible. The training corpora are either multilingual parallel data for machine trans-

* Work done during internship at Bosch AI.

[1]The code, datasets, and models are publicly available at https://github.com/boschresearch/ACL24-MLLM.

[2]https://www.ethnologue.com

Findings of the Association for Computational Linguistics: ACL 2024, pages 8186–8213

lation tasks (Yang et al., 2023; Zhu et al., 2023), or multilingual instruction data for instruction tuning (Üstün et al., 2024; Groeneveld et al., 2024; Luo et al., 2023; Li et al., 2023; Lai et al., 2023a). The second group includes approaches that align non-English instructions with English instructions through cross-lingual prompting in the inference stage (Huang et al., 2023; Etxaniz et al., 2023). While achieving impressive performance, both groups suffer from major problems. First, they only support a small number of languages, and most of the world's languages are still being left behind. Second, they use large corpora for supervised fine tuning (SFT), neglecting the exploration of alignment to human preferences.

To address the aforementioned issues, we aim to scale the two multilingual capabilities of LLMs at the same time. To improve the understanding capability of LLMs, we construct a multilingual instruction dataset with 100 languages by translating instructions from Alpaca (Taori et al., 2023) via ChatGPT[3] and Google Translate API[4]. We finetune LLMs on our constructed multilingual dataset using parameter efficient fine tuning (PEFT; Hu et al., 2021). The languages of instruction and output are always the same language in the existing human feedback dataset (Taori et al., 2023), which limits the generating capability of LLMs. To enhance the generating capability, we construct a cross-lingual human feedback data (i.e., instruction and output are different languages) covering 30 languages[5]. We then further align the LLMs with human feedback using the DPO algorithm (Rafailov et al., 2023). Finally, we obtain an LLM that supports 100 languages which has the capability to understand the instructions of 100 languages and supports output in 100 languages.

We conduct a comprehensive evaluation to verify the effectiveness of xLLMs-100 on five diverse multilingual benchmarks, covering understanding (PAWS-X; Yang et al., 2019), reasoning (XCOPA; Ponti et al., 2020), generation (XL-Sum; Hasan et al., 2021 and FLORES-101; Goyal et al., 2022) and expert-written (Self-Instruct; Wang et al., 2023) tasks in the zero-shot setting. Each benchmark includes multilingual evaluation data covering both high-resource and low-resource languages. The experimental results clearly demonstrate that xLLMs-100 significantly enhances both the understanding and generating capabilities of LLMs simultaneously across all benchmarks. Furthermore, our extensive analysis experiments reveal that xLLMs-100 not only mitigate the off-target problem (i.e., LLMs generate the text into an incorrect language (Zhang et al., 2020)) but also enhance language democratization (i.e., democratization degree of tasks between languages (Huang et al., 2023)) compared with strong LLMs.

In summary, we make the following contributions: **(i)** We construct two datasets, one of which contains a multilingual instructions in 100 languages, and the other one contains cross-lingual human preferences in 30 languages. **(ii)** We evaluate the multilingual capabilities of LLMs in two dimensions: understanding and generating capability. Unlike previous studies that assess these capabilities in isolation, we urge the community to consider both capabilities when evaluating the multilingual performance of LLMs. **(iii)** We scale the multilingual capabilities of LLMs to perform well across 100 languages.

2 Related Work

Multilingual Capabilities of LLMs. Recent studies have applied LLMs to various NLP tasks in a multilingual setting (Üstün et al., 2024; Groeneveld et al., 2024; Lai et al., 2023a; Weissweiler et al., 2023). In general, two kinds of corpora have been used to improve the multilingual capabilities of LLMs: multilingual parallel corpora (BigTranslate; Yang et al., 2023) and multilingual instruction datasets (Bactrian-X; Li et al., 2023 and xP3; Muennighoff et al., 2023). Yang et al. (2023) continued training LLaMA (Touvron et al., 2023) using a multilingual parallel corpus that covers 102 languages and achieved good results on a multilingual machine translation task. However, the results from our initial experiments indicate that the performance of this model on non-machine translation tasks is not as good as its performance on the evaluated machine translation task (see Section 6.2 for more details). Li et al. (2023) finetunes LLaMA-2 using a multilingual instruction dataset including 52 languages, and achieved good performance on several multilingual NLP tasks. In contrast, we construct a multilingual instruction dataset including 100 languages. To the best of our knowledge, our dataset is the multilingual instruction dataset with the largest language coverage to date.

[3] https://chat.openai.com

[4] https://translate.google.com

[5] We include 30 languages for that ChatGPT provides high quality feedback.

Aligning LLMs with Human Feedback. A prominent method for LLM training is reinforcement learning from human feedback (RLHF; Ouyang et al., 2022), which learns from human feedback instead of relying on a pre-defined reward function. Despite its popularity, there are significant flaws. Collecting human preferences is time consuming. The instability of the RLHF method during training also poses a significant challenge in learning the optimal reward function from human preference data (Schulman et al., 2017). Recently, the DPO algorithm (Rafailov et al., 2023) has demonstrated that this challenge can be addressed by fine-tuning LLMs to align with human preferences using a supervised learning regime, without the requirement of explicit reward modeling or reinforcement learning. DPO has only been applied to monolingual human feedback data (i.e., instructions, inputs, and outputs are in the same language). While such fine-tuned LLMs are beneficial for monolingual tasks, they have not been shown to generalize to tasks that require text understanding and generation in various languages. In this paper, we construct a large-scale cross-lingual human preference dataset covering 30 languages and show promising results in multiple NLP tasks.

3 Method

To scale the multilingual capabilities of LLMs, we construct two datasets: a multilingual instruction dataset (Section 3.1) and a cross-lingual feedback dataset (Section 3.2). Then, we evaluate the quality of the translated instructions and generated responses on our constructed datasets (Section 3.3). Finally, we introduce the training process of our multilingual instruction fine-tuning (Section 3.4). Data statistics and analyses can be found in the Appendix A.

3.1 Multilingual Instruction Dataset

Alpaca (Taori et al., 2023) contains 52K instruction and demonstration pairs generated by OpenAI's `text-davinci-003` engine using the self-instruct technique (Wang et al., 2023). Some of the existing multilingual instruction datasets are primarily translated from the Alpaca datasets. For instance, Lai et al. (2023a) and Li et al. (2023) expanded the Alpaca dataset to include 26 and 52 languages, respectively. To further scale the multilingual capabilities of LLMs, we expand the Alpaca dataset to 100 languages. Our construction process contains two steps: instruction translation and hybrid response generation.

Instruction Translation. We use Google Translate API to translate English instructions and inputs in the Alpaca dataset into 100 languages covered in the FLORES-101 dataset (Goyal et al., 2022). We chose the Google Translate API because it outperforms state-of-the-art translators such as NLLB, DeepL[6], and GPT-4 (Achiam et al., 2023) in translation performance across multiple languages (Yang et al., 2023; Robinson et al., 2023). For languages that are not supported by Google Translate, we employ the NLLB model (Costa-jussà et al., 2022) for translation, as it is currently considered the state-of-the-art for multilingual translation, particularly for low-resource languages. Similar to Bactrian-X (Li et al., 2023), we do not translate instructions that contain program-related text.

Hybrid Response Generation. Intuitively, there are two alternatives to obtain responses in various languages: the translation-based method and the generation-based method. The translation-based approach involves directly translating Alpaca's English responses into one of the 100 target languages using either the Google Translate API or the state-of-the-art multilingual machine translation model. Generation-based methods, like Bactrian-X (Li et al., 2023), take instructions that have been translated into the desired target language and feed them into LLMs (e.g., ChatGPT), resulting in a response expressed in the target language. Both of these methods can produce responses in the target language, however, they also have notable limitations. Without the context, the translation-based approach usually translates responses in a different style from the native speaker and has the potential problem of `translationese` (Riley et al., 2020). More importantly, the understanding and generating capabilities of LLMs varies significantly across languages, and it is extremely difficult for the generation-based method to generate a high quality response in low-resource languages. To solve the above problems, we generate responses in a hybrid mode. We motivate our approach with the translation performance (i.e., translation task from English to other languages) of ChatGPT in the FLORES benchmark demonstrated in Lu et al. (2023). We assume that languages exhibiting poor translation performance (typically with BLEU scores below 10) by Chat-

[6] `www.deepl.com`

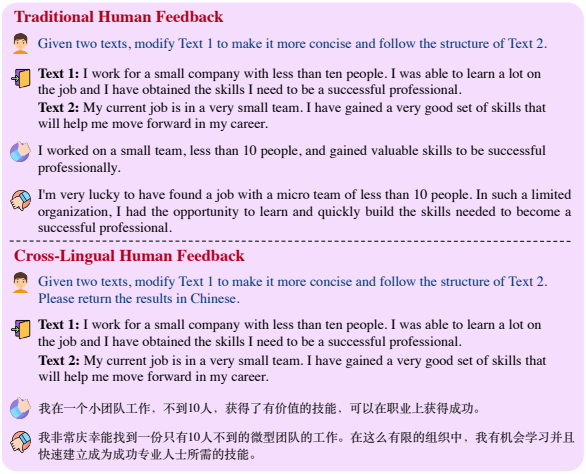

Figure 1: Cross-lingual human feedback dataset. Given instructions and inputs written in English, both the accepted and rejected outputs are written in Chinese.

GPT also possess limited generating capabilities. Consequently, for languages with poor translation quality, we directly translate the English responses in Alpaca into the specific target language using the Google Translate API or NLLB model, while languages with good translation quality have their responses generated by ChatGPT. ChatGPT's responses are preferable because they seem to have a more consistent style. Table 11 in the Appendix A details the translator (Google Translate API or the NLLB model) used for each language's translation instructions, as well as the method (ChatGPT, Google Translate API, or the NLLB model) employed to generate responses.

3.2 Cross-Lingual Feedback Dataset

Aligning LLMs with human preferences is crucial in enhancing the truthfulness of their generated responses. Most datasets with human feedback are monolingual, e.g., English (Peng et al., 2023) and Chinese (Sun et al., 2023). Recently, Lai et al. (2023a) extend the human feedback dataset to 26 languages using two rounds of dialogue (translation and ranking) via ChatGPT. However, one of the biggest problems with human feedback data nowadays is that their instructions, inputs and outputs are in the same language, which limits the generative capability of LLMs. To enhance the generating capability, we construct a cross-lingual human feedback dataset covering 30 languages. This dataset provides both the instruction and output in different languages. Such a design is advantageous as it simulates a wide range of generating scenarios, thereby enhancing the generating capability of LLMs. For instance, if we have human preference data available in 30 languages, we can simulate up to $30 \times 29 = 870$ generation scenarios. The construction process has two steps: instruction design and response generation. We show an example generated cross-lingual human feedback in Figure 1.

Instruction Design. Given a source language ℓ_s, we first translate the instruction written in English into the instruction written in ℓ_s using Google Translate API. We denote the instruction written in ℓ_s as I^{ℓ_s}. Then, we randomly select one of the 30 languages, excluding ℓ_s, as the target language ℓ_t and design an instruction written in ℓ_s to return an output written in ℓ_t. We denote the instruction written in ℓ_t as I^{ℓ_t}. Finally, we merge I^{ℓ_s} and I^{ℓ_t} to construct a new instruction as $I^{\ell_t}_{\ell_s}$.

Response Generation. Inspired by Lai et al. (2023a), we rank the responses from ChatGPT according to their quality for the instruction and input text. Given the new instruction $I^{\ell_t}_{\ell_s}$, we use ChatGPT to generate the responses in target language ℓ_t and rank the responses. The ranking process scores different responses based on three factors: correctness, coherence and naturalness. We take the response with the highest score as the accepted response, denoted as R_a and the response with the lowest score as the rejected response, denoted as R_r. Note that, both the R_a and R_r are written in ℓ_t.

	BLEU	COMET
[0,10)	2	0
[10,20)	7	0
[20,30)	15	0
[30,40)	18	0
[40,50)	26	3
[50,60)	16	8
(60,70]	9	19
(70,80]	5	29
(80,90]	2	32
(90,100]	0	9

Table 1: The number of languages for BLEU and COMET scores fall within each interval, obtained by back-translating from 100 languages into English.

3.3 Instruction and Response Quality

To evaluate the quality of the instructions in our constructed dataset, we randomly choose 50 in-

High			Low		
	BLEU	CP		BLEU	CP
Arabic	73.16	0.82	Armenian	47.16	0.64
Chinese	80.27	0.91	Gujarati	39.68	0.55
French	77.71	0.85	Kannada	41.72	0.57
German	75.50	0.84	Malayalam	45.24	0.62
Hindi	73.26	0.81	Marathi	41.37	0.56
Avg.	75.98	0.85	**Avg.**	43.03	0.59

Table 2: BLEU and content preservation (CP) of the response quality for 5 high-resource laguages and 5 low-resource languages.

structions per language and translate them into English using the Google Translate API. Then, we evaluate the BLEU (Post, 2018) and COMET (Rei et al., 2020) of the back-translated instructions against the original English instructions in Alpaca. Table 1 shows the number of languages within each interval segment for BLEU and COMET scores. We find that most languages have BLEU scores between 20 and 60 and COMET scores between 60 and 90, indicating that the quality of the constructed instructions is high.

To evaluate the quality of the generated responses, we randomly select 5 high-resource (Arabic, Chinese, French, German and Hindi) and 5 low-resource languages (Armenian, Gujarati, Kannada, Malayalam and Marathi), and evaluate 100 responses in each language. We use two metrics for this evaluation. First, similar to our evaluation on the instruction quality, we back-translate the responses into English and calculate the BLEU score. Second, we assess content preservation (CP), which is a crucial metric in text style transfer (Jin et al., 2022). It measures the degree of meaning preservation between two texts by calculating the cosine similarity between the vectors of the original and generated texts. We use this metrics by mapping the language-specific responses and the original English responses in Alpaca into the same vector space using a multilingual sentence embedding model (LaBSE; Feng et al., 2022), then we compute their cosine distance. As shown in Table 2, the BLEU scores for responses in high-resource languages ranges from 73.16 to 80.27 and effectively preserved meaning (i.e., CP ranges from 0.81 to 0.91). In addition, the CP score is more than 0.8 for high-resource languages and about 0.6 for low-resource ones, which indicate enough quality for the purpose of this research.

3.4 Multilingual Instruction Tuning

To enhance both the understanding and generating capabilities of LLMs, we employ a two-step training process: supervised fine-tuning and alignment with human preferences.

Supervised Fine-tuning (SFT). Starting with an LLM, e.g., LLaMA (Touvron et al., 2023) or BLOOM (BigScience et al., 2022), we perform supervised fine-tuning on the LLM using our constructed multilingual instruction dataset. Since fine-tuning on full parameters across all layers of an LLM is computationally expensive, we apply a parameter-efficient fine-tuning technique, specifically LoRA (Hu et al., 2021). LoRA incorporates trainable rank decomposition matrices into the LLM layers and only updates the newly introduced parameters while freezing all parameters of the original LLMs. LoRA has been shown to achieve comparable performance to full parameter fine-tuning methods on LLMs (Chen et al., 2023).

Aligning LLMs with human feedback. Common practice for aligning LLMs with human feedback is to use reinforcement learning (RLHF; Ouyang et al., 2022) to employ the Proximal Policy Optimization (PPO; (Schulman et al., 2017)) algorithm to maximizes the reward of the model. However, RLHF suffers from instability of reinforcement technical and requires significant computational resources (Casper et al., 2023). To save computational resources, we further fine-tune the trained SFT model from the last step with the DPO algorithm (Rafailov et al., 2023) with our constructed cross-linguistic human feedback dataset. Note that we also use LoRA in the whole DPO training process.

4 Experiments

4.1 Datasets and Tasks

We evaluate xLLMs-100 on five typical benchmarks including generation, reasoning, understanding and expert-written tasks that measure the multilingual capabilities of LLMs, including both high-resource and low-resource languages.

Understanding Task. We evaluate our LLM for the paraphrase identification task on PAWS-X (Yang et al., 2019) benchmark on 7 languages. The benchmark provides two sentences and asks the model to determine whether they paraphrase each other or not.

Generation Task. We evaluate the multilingual capability of our LLM on the FLORES-101 (Goyal et al., 2022) and XL-Sum (Hasan et al., 2021) benchmarks. FLORES-101 is a machine translation benchmark, containing parallel sentences between 101 languages. FLORES-101 lets us evaluate our LLM for tasks in which input and output texts are in different languages. XL-Sum is a summarization benchmark covering 44 languages, where an LLM should summarize a long text into a short text. XL-Sum lets us evaluate LLMs for tasks where input and output texts are in the same language.

Reasoning Task. We evaluate the commonsense reasoning task using the XCOPA (Ponti et al., 2020) benchmark, which includes texts written in 11 languages. A data sample in XCOPA consists of one premise and two choices, and requires the model to select which choice is the effect or cause of the given premise.

Expert-written Task. We use the Google Translate API to translate the Self-Instruct (Wang et al., 2023) benchmark from English to five high-resource languages (Arabic, Czech, German, Chinese, Hindi) and five low-resource languages (Armenian, Kyrgyz, Yoruba, Tamil, Mongolian). We call this dataset Self-Instruct*. This benchmark contains the instruction, input and output in each instance, and requires that the LLM predicts the correct answer in the correct target language.

4.2 Baselines

We compare xLLMs-100 with the following baselines:

Off-the-shelf LLMs. We evaluate LLaMA-2 (Touvron et al., 2023) and BLOOM (BigScience et al., 2022) as vanilla LLM baselines without additional finetuning.

Publicly available multilingual instruction-tuned models: Bactrian-X is the instruction-tuned model proposed by Li et al. (2023). These models were instruction-tuned on 52 languages. They released models based on LLaMA and BLOOM. We refer to them as BX_{LLaMA} and BX_{BLOOM}, respectively.

Supervised Fine-Tuning (SFT): We performed instruction tuning (Taori et al., 2023) by utilizing our constructed multilingual instruction dataset. We denote these models as $\text{SFT}_{\text{LLaMA}}$ and $\text{SFT}_{\text{BLOOM}}$.

4.3 Implementation

We use the 7B model of LLaMA (chat version; `Llama-2-7b-chat-hf`) and BLOOM (basic version; `bloom-7b1`) in all experiments as the base model. We train our model using PyTorch with the HuggingFace transformers[7] and PEFT[8] implementation. Hyperparameters used for training our model can be found in Appendix B. We evaluate our model in the zero-shot setting. Recent studies have shown that slight modifications in input prompts can lead to varied results (Loya et al., 2023). To ensure the reproducibility, we present the prompts used for each task in the experiment in Appendix C. To mitigate over-fitting, we set the number of epochs to 1 for both our multilingual fine-tuning and DPO training processes.

4.4 Experimental Settings

Similar to Huang et al. (2023), to save inference time, we randomly select 200 test samples for each language in FLORES-101 and 250 test samples for each language in XL-Sum. For FLORES-101, we have two settings: translation tasks from English to other languages and translation tasks from the other languages to English, denoted as FLORES(f) and FLORES(t), respectively. In addition to XL-Sum and PAWS-X benchmark, we categorize the languages in each dataset into low-resource and high-resource languages based on the language classification in Costa-jussà et al. (2022). We classify the languages in XL-Sum dataset into low, mid, and high categories according to Hasan et al. (2021). The PAWS-X dataset does not include low-resource languages.

4.5 Evaluation Metric

For FLORES-101, we report case-sensitive detokenized BLEU with SacreBLEU[9](Post, 2018). For the XCOPA and PAWS-X benchmarks, we utilize the accuracy score for evaluation. For the XL-Sum and Self-Instruct* benchmark, we report the multilingual ROUGE-1 score implemented by Lin (2004).

5 Results

Our goal is to simultaneously evaluate the understanding capability and the generating capability

[7] https://github.com/huggingface/transformers
[8] https://github.com/huggingface/peft
[9] https://github.com/mjpost/sacrebleu

Understanding Capabilities												
	PAWS-X	**XCOPA**		**Self-Instruct***		**XL-Sum**			**FLORES(f)**		**FLORES(t)**	
		low	high	low	high	low	mid	high	low	high	low	high
LLaMA	38.10	47.44	47.22	7.09	12.57	4.07	5.44	2.84	3.07	4.95	2.96	6.61
BX_{LLaMA}	37.28	49.53	49.00	6.31	11.88	2.17	5.52	7.89	2.69	2.38	3.15	5.31
SFT_{LLaMA}	42.32	50.19	49.86	7.32	12.72	4.70	7.34	7.55	3.13	3.93	3.16	6.92
xLLMs-100	**46.95**	**51.53**	**51.96**	**12.94**	**15.35**	**8.83**	**13.90**	**17.29**	**3.27**	**8.09**	**4.04**	**14.18**
BLOOM	36.47	44.27	49.14	7.56	8.67	9.03	14.06	16.80	2.54	2.04	2.05	2.56
BX_{BLOOM}	36.42	46.28	50.35	4.81	8.11	4.89	8.47	11.71	2.14	1.74	2.41	1.57
SFT_{BLOOM}	36.67	49.42	52.31	6.31	11.88	5.62	10.12	14.33	**3.12**	3.79	2.62	2.52
xLLMs-100	**39.83**	**52.50**	**55.59**	**7.94**	**13.35**	**12.87**	**15.23**	**18.38**	3.02	**4.71**	**3.94**	**6.54**
Generating Capabilities												
	PAWS-X	**XCOPA**		**Self-Instruct***		**XL-Sum**			**FLORES(f)**		**FLORES(t)**	
		low	high	low	high	low	mid	high	low	high	low	high
LLaMA	50.22	49.33	51.52	5.38	8.81	6.26	5.80	8.08	1.35	3.90	2.11	4.95
BX_{LLaMA}	48.41	48.00	49.85	7.01	9.80	1.11	2.74	1.70	1.56	5.33	1.37	1.61
SFT_{LLaMA}	50.36	48.93	50.05	7.10	12.15	4.51	6.06	9.21	2.42	4.56	2.71	7.29
xLLMs-100	**61.94**	**49.71**	**54.68**	**9.16**	**14.71**	**9.99**	**13.57**	**16.61**	**2.89**	**9.07**	**5.64**	**16.98**
BLOOM	47.39	49.85	49.47	4.07	7.01	6.08	7.77	8.91	0.78	1.20	0.99	1.49
BX_{BLOOM}	47.26	47.72	49.98	5.88	8.21	1.98	3.59	4.58	0.47	0.82	1.95	2.33
SFT_{BLOOM}	48.50	49.13	49.28	7.78	11.51	3.89	8.87	10.89	2.59	3.12	2.05	2.56
xLLMs-100	**50.53**	**52.36**	**52.26**	**10.17**	**13.62**	**8.77**	**11.74**	**12.36**	**3.97**	**5.79**	**4.22**	**7.68**

Table 3: **Understanding and generating capability of LLMs.** We evaluate on five benchmarks covers both on high-resource and low-resource languages. We utilize accuracy score to evaluate on PAWS-X and XCOPA. We evaluate Self-Instruct* and XL-Sum using ROUGE-1 score and FLORES using BLEU.

of the LLMs. To evaluate the understanding capability of LLMs, we use Google Translate API to translate the instructions for each task into different languages. To evaluate the generating capability of an LLM, we use English as the instruction language during the inference phase. We choose English because LLMs demonstrate effective comprehension of English instructions, thus avoiding any potential impact on the generating capability of LLMs due to misunderstanding of instructions. Table 3 presents the average score for all languages in each benchmark. For more comprehensive results of each benchmark per language, please refer to Appendix D.

The off-the-shelf LLMs demonstrate limited multilingual capabilities, in particular in generation tasks, where performance is exceedingly poor. After fine-tuning the LLMs using the multilingual instruction dataset, there is a marginal (but not significant) improvement in the multilingual capability of the LLMs across most benchmarks, especially when describing instructions in non-English languages. For example, the understanding capability of the BX-based models (BX_{LLaMA} and BX_{BLOOM}) drops significantly on the Self-Instruct*, XL-Sum, and FLORES benchmarks. This phenomenon is due to the fact that the BX-based models are fine-tuned in 52 languages and do not completely support all the languages in the benchmarks.

Low-Resource vs High-Resource. We observe that the multilingual capabilities of the examined LLMs are significantly superior in high-resource languages to those in low-resource languages. It is a common problem for multilingual models that the scarcity of training corpora in low-resource languages makes it challenging to train a robust decoder, leading to the generation of incorrect outputs (Lai et al., 2023b).

Understanding Capability vs Generating Capability. Our findings indicate that instructions written in English outperform instructions written in non-English languages, aligning with our initial expectations. This phenomenon can be attributed to the accumulation of biases in LLMs' understanding of instructions across different languages. Such biases can introduce errors during the generation of results, resulting in incorrect outputs or outputs in the wrong language (i.e., off-target problem; Zhang

	Low		High	
	mono	cross	mono	cross
PAWS-X	-	-	58.43	**61.94**
XCOPA	47.26	**49.71**	52.15	**54.68**
Self-Instruct*	3.25	**9.16**	12.14	**14.71**
XL-Sum	3.38	**9.99**	12.52	**16.61**
FLORES(f)	0.85	**2.89**	4.57	**9.07**
FLORES(t)	1.55	**5.64**	8.45	**16.98**

Table 4: An ablation study of xLLMs-100 using monolingual and cross-lingual human feedback data on low- and high-resource languages.

	Low		High	
	para	instruct	para	instruct
PAWS-X	-	-	40.17	**50.36**
XCOPA	37.14	**48.93**	42.13	**50.05**
Self-Instruct*	2.63	**7.10**	5.48	**12.15**
XL-Sum	1.10	**4.51**	5.12	**9.21**
FLORES(f)	**5.06**	2.42	**13.27**	4.56
FLORES(t)	**12.36**	2.71	**18.27**	7.29

Table 5: Multilingual Tuning on multilingual parallel corpora and multilingual instruction dataset in high-resource and low-resource languages.

	FLORES(f)		FLORES(t)	
	Low	High	Low	High
LLaMA	23.26	16.76	14.15	10.16
BX_{LLaMA}	14.13	8.32	12.17	8.24
SFT_{LLaMA}	10.26	6.34	8.72	6.23
xLLMs-100	**8.82**	**3.47**	**6.95**	**1.46**

Table 6: OTR scores (lower is better) of examined multilingual LLMs on the FLORES benchmark.

et al., 2020).

xLLMs-100. Compared with the baseline models, our models demonstrate consistent improvements for both multilingual capabilities across all examined tasks. In comparison to the SFT-based models, xLLMs-100 increases the step of alignment with human preferences, resulting in additional improvements. This observation highlights the significance of aligning with human preferences after supervised fine-tuning. Furthermore, our model significantly outperforms other models on generative task datasets such as XL-Sum and FLORES, showing the effectiveness of our dataset and human feedback finetuning.

6 Analysis

In Section 5 we show the effectiveness of our approach compared with previous work. In this section, we study the performance of our multilingual model in detail.

6.1 Different Human Feedback Datasets

To investigate the importance of cross-lingual properties in aligning LLMs with human preferences, we conduct an ablation study on the same five benchmarks as shown in Table 3. In particular, we employ the DPO algorithm (Rafailov et al., 2023) to finetune our model, xLLMs-100, on two distinct datasets. The first dataset is our constructed cross-lingual human feedback dataset, where instructions and outputs are in different languages. The second dataset is a traditional monolingual human feedback dataset (Lai et al., 2023a), where both instructions and outputs are in the same language. Table 4 show the results categorized by low- and high-resource languages. We observe that (1) Aligning xLLMs-100 using our cross-lingual human feedback dataset yields superior results compared with using monolingual human feedback. This improvement is evident for datasets with generation tasks such as XL-SUM and FLORES, showing that our novel cross-lingual human feedback dataset effectively simulates the multilingual generation task, reducing the possibilities of generating incorrect outputs. (2) Finetuning xLLMs-100 on our cross-lingual human feedback dataset is more effective for low-resource languages than high-resource ones. This is due to the fact that high-resource languages already exhibit strong understanding and generation capabilities in the vanilla LLMs (as shown in Table 3), which mitigates the impact of further finetuning xLLMs-100 on the cross-lingual preference data. (3) It is worth noting that despite aligning xLLMs-100 with cross-lingual human preferences, its performance in low-resource languages is still not as good as that in high-resource languages. Although our constructed cross-lingual feedback dataset enhances multilingual performance, the inclusion of additional languages (our dataset currently includes 30 languages) might be necessary to support all low-resource languages in the evaluation benchmarks.

	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100
PAWS-X	60.56	58.77	60.63	**66.43**
XCOPA	93.33	98.52	**99.31**	89.63
Self-Instruct*	57.85	68.68	62.63	**73.92**
XL-Sum	47.09	8.90	50.35	**67.21**
FLORES(f)	34.33	34.00	25.84	**34.68**
FLORES(t)	49.84	**58.28**	35.53	48.28

Table 7: **Language Democratization:** Mitigating the gap between the average performance and the best performances of each task in different languages.

6.2 Different Datasets for Multilingual Tuning

In Section 1, we introduced two types of datasets to enhance the multilingual capabilities of LLMs: multilingual parallel corpus and multilingual instruction dataset. We study how these two types of data impact the multilingual capabilities of LLMs. To do so, we conduct comprehensive comparison experiments on these two types of dataset. For the multilingual parallel corpus, we use the NLLB dataset (Costa-jussà et al., 2022) collected by allenai[10]. We use the same set of 100 languages as xLLMs-100 for the sake of a fair comparison.

The experimental results presented in Table 5 clearly show that utilizing the multilingual parallel corpus leads to a significant improvement in the machine translation task. However, there is a notable decrease in performance observed in non machine translation tasks. This phenomenon is referred to as catastrophic forgetting (McCloskey and Cohen, 1989), where the model after fine-tuning achieves better performance on the new task at the expense of the model's performance on other tasks. On the other hand, the model fine-tuned with multilingual instruction data demonstrates improvement across all tasks, indicating that attention should be given to the models' performance on a broad range of tasks during fine-tuning, rather than focusing solely on performance gains in a single task. It is obvious that the multilingual instruction dataset is a better choice compare to the multilingual parallel dataset when finetuning LLMs.

6.3 Off-Target Analysis

Off-target (Zhang et al., 2020) refers to the generation of output in an incorrect language, which is a common issue in multilingual models. Following the approach of Lai et al. (2023c), we calculate the Off-Target Ratio (OTR), which represents the proportion of output sentences generated by a multilingual model that are in the wrong language. This metric helps to assess the accuracy of language generation in multilingual models. Table 6 shows the results on the XL-Sum and FLORES benchmark. We observe that the off-target problem is more prominent in low-resource languages. Although our model has made some progress in addressing this issue, there is still significant room for improvement. Overcoming the off-target problem in low-resource languages continues to be a challenge that necessitates further research and development efforts.

[10] https://huggingface.co/datasets/allenai/nllb

6.4 Language Democratization

Language democratization, as proposed by Huang et al. (2023), is a metric used to evaluate the level of task democratization across different languages of a multilingual model. This metric is obtained by calculating the average percentage of different languages relative to the best performing language among all languages. It provides insights into the fairness and equality of performance across different languages in a multilingual model. According to Table 7, we observe that xLLMs-100 demonstrates a higher degree of linguistic democratization its peers across four out of six examined benchmarks. This implies that our model exhibits a smaller performance gap between different languages, indicating a more equal and fair distribution of performance across languages. This is a positive outcome, suggesting that our model is successful in reducing disparities and achieving more balanced performance across various languages.

7 Conclusions

To enhance the multilingual capability of LLMs in two dimensions (understanding and generating), we present xLLMs-100, two LLMs that support 100 languages. We train xLLMs-100 on a novel multilingual instruction dataset containing 100 languages. To improve the generating capability, we construct a cross-lingual human feedback dataset to further align the LLMs with human feedback and to enable the LLMs to generate output in multiple languages. Experiments on five benchmarks demonstrate the effectiveness of our datasets and also the multilingual capability of our models both on high-resource languages and low-resource languages.

8 Limitations

This work has the following limitations: (i) To make our computations environment friendly, our experiments have so far been limited to 7B size on LLaMA and BLOOM. However, our dataset has the potential to be deployed for larger LLMs (e.g., 13B and 70B models). We hope that the community will contribute to realizing this potential using our dataset. (ii) Our constructed human feedback dataset currently covers 30 languages. Given that ChatGPT also faces difficulties in obtaining high-quality feedback across a majority of languages, determining how to extend the cross-lingual human feedback dataset would be promising future work. (iii) As shown in Appendix E and F, a noticeable discrepancy persists between xLLMs-100, small language models (SLMs) and the current state-of-the-art LLMs, such as ChatGPT. This disparity is primarily due to models like ChatGPT leveraging larger quantities of data and model sizes. Therefore, bridging this gap will be a critical objective in future research. (iv) We do not conduct an analysis on toxicity (Deshpande et al., 2023), domain (Lai et al., 2022a,b), bias and fairness (Gallegos et al., 2023) aspects of xLLMs-100, which should be discussed more in future work.

Acknowledgement

This publication was partially supported by LMUexcellent, funded by the Federal Ministry of Education and Research (BMBF) and the Free State of Bavaria under the Excellence Strategy of the Federal Government and the Länder; and by the German Research Foundation (DFG; grant FR 2829/4-1).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 model card.

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*, pages 4623–4637, Online. Association for Computational Linguistics.

Workshop BigScience, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. 2023. Open problems and fundamental limitations of reinforcement learning from human feedback. *arXiv preprint arXiv:2307.15217*.

Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Barry Haddow, and Kenneth Heafield. 2023. Monolingual or multilingual instruction tuning: Which makes a better alpaca. *arXiv preprint arXiv:2309.08958*.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Ameet Deshpande, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, and Karthik Narasimhan. 2023. Toxicity in chatgpt: Analyzing persona-assigned language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1236–1270, Singapore. Association for Computational Linguistics.

Julen Etxaniz, Gorka Azkune, Aitor Soroa, Oier Lopez de Lacalle, and Mikel Artetxe. 2023. Do multilingual language models think better in english? *arXiv preprint arXiv:2308.01223*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. 2023. Bias and fairness in large language models: A survey. *arXiv preprint arXiv:2309.00770*.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. 2021. XL-sum: Large-scale multilingual abstractive summarization for 44 languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online. Association for Computational Linguistics.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2021. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

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.

Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. 2022. Deep learning for text style transfer: A survey. *Computational Linguistics*, 48(1):155–205.

Viet Lai, Chien Nguyen, Nghia Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan Rossi, and Thien Nguyen. 2023a. 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.

Wen Lai, Alexandra Chronopoulou, and Alexander Fraser. 2022a. m^4 adapter: Multilingual multi-domain adaptation for machine translation with a meta-adapter. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4282–4296, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Wen Lai, Alexandra Chronopoulou, and Alexander Fraser. 2023b. Mitigating data imbalance and representation degeneration in multilingual machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14279–14294, Singapore. Association for Computational Linguistics.

Wen Lai, Viktor Hangya, and Alexander Fraser. 2023c. Extending multilingual machine translation through imitation learning. *arXiv preprint arXiv:2311.08538*.

Wen Lai, Jindřich Libovický, and Alexander Fraser. 2022b. Improving both domain robustness and domain adaptability in machine translation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5191–5204, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

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

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Manikanta Loya, Divya Sinha, and Richard Futrell. 2023. Exploring the sensitivity of LLMs' decision-making capabilities: Insights from prompt variations and hyperparameters. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3711–3716, Singapore. Association for Computational Linguistics.

Hongyuan Lu, Haoyang Huang, Dongdong Zhang, Haoran Yang, Wai Lam, and Furu Wei. 2023. Chain-of-dictionary prompting elicits translation in large language models. *arXiv preprint arXiv:2305.06575*.

Yin Luo, Qingchao Kong, Nan Xu, Jia Cao, Bao Hao, Baoyu Qu, Bo Chen, Chao Zhu, Chenyang Zhao, Donglei Zhang, et al. 2023. Yayi 2: Multilingual open-source large language models. *arXiv preprint arXiv:2312.14862*.

Michael McCloskey and Neal J Cohen. 1989. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier.

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.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Edoardo Maria Ponti, Goran Glavaš, Olga Majewska, Qianchu Liu, Ivan Vulić, and Anna Korhonen. 2020. XCOPA: A multilingual dataset for causal commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2362–2376, Online. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290*.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Parker Riley, Isaac Caswell, Markus Freitag, and David Grangier. 2020. Translationese as a language in "multilingual" NMT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7737–7746, Online. Association for Computational Linguistics.

Nathaniel Robinson, Perez Ogayo, David R. Mortensen, and Graham Neubig. 2023. ChatGPT MT: Competitive for high- (but not low-) resource languages. In *Proceedings of the Eighth Conference on Machine Translation*, pages 392–418, Singapore. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Hao Sun, Zhexin Zhang, Jiawen Deng, Jiale Cheng, and Minlie Huang. 2023. Safety assessment of chinese large language models. *arXiv preprint arXiv:2304.10436*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

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

Ahmet Üstün, Viraat Aryabumi, Zheng-Xin Yong, Wei-Yin Ko, Daniel D'souza, Gbemileke Onilude, Neel Bhandari, Shivalika Singh, Hui-Lee Ooi, Amr Kayid, et al. 2024. Aya model: An instruction finetuned open-access multilingual language model. *arXiv preprint arXiv:2402.07827*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. 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.

Leonie Weissweiler, Valentin Hofmann, Anjali Kantharuban, Anna Cai, Ritam Dutt, Amey Hengle, Anubha Kabra, Atharva Kulkarni, Abhishek Vijayakumar, Haofei Yu, Hinrich Schuetze, Kemal Oflazer, and David Mortensen. 2023. Counting the bugs in ChatGPT's wugs: A multilingual investigation into the morphological capabilities of a large language model. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6508–6524, Singapore. Association for Computational Linguistics.

Wen Yang, Chong Li, Jiajun Zhang, and Chengqing Zong. 2023. Bigtrans: Augmenting large language models with multilingual translation capability over 100 languages. *arXiv preprint arXiv:2305.18098*.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.

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

A Datasets

We construct two datasets for multilingual tuning: multilingual instruction dataset in 100 languages and cross-lingual human feedback dataset in 30 languages. We tokenize the inputs, instructions, and outputs of these two datasets separately, and present the statistics of the results in Table 12. We observe that both LLaMA have a large average number of tokens, indicating that the original LLMs do not adequately support all languages in our constructed datasets. The sentences are segmented into smaller units, which increases the difficulty of capturing semantic information. Addressing the tokenizer issue in LLMs is an important step towards expanding LLMs to new languages, and we would like to address this in future work. In addition, we show the languages covered in our constructed cross-lingual dataset in Table 8. Finally, the translator used for each language when translating the instructions and responses are shown in Table 11.

#Langs	Type	#Langs	Type
Arabic	High	Russian	High
Basque	High	Slovak	High
Bengali	High	Spanish	High
Chinese	High	Swedish	High
Croatian	High	Ukrainian	High
Danish	High	Vietnamese	High
Dutch	High	Armenian	Low
French	High	Gujarati	Low
German	High	Kannada	Low
Hindi	High	Malayalam	Low
Hungarian	High	Marathi	Low
Indonesian	High	Nepali	Low
Italian	High	Serbian	Low
Portuguese	High	Tamil	Low
Romanian	High	Telugu	Low

Table 8: Languages covered in our constructed cross-lingual human feedback dataset.

B Model Configuration

The training process of xLLMs-100 contains two steps. In supervised fine-tuning, we use LORA to fine-tune LLMs on our constructed multilingual instruction dataset. In aligning LLMs with human feedback, we use DPO with LORA to fine-tune the LLMs. We train xLLMs-100 on one machine with 8 A100 80GB GPUs. The hyper-parameters for xLLMs-100 are presented in Table 9.

SFT		LORA		DPO	
batch size	4	r	8	batch size	8
epoch	1	alpha	16	epoch	1
learning rate	1e-4	dropout	0.05	learning rate	5e-4
max length	1024			max length	1024

Table 9: Hyper-parameters for multilingual tuning.

C Prompt Setting

To evaluate the generating capability of LLMs, we use English as the prompt language as shown in Table 13. To evaluate the understanding capability of LLMs, we use Google Translate API to translate the English prompt to the other languages.

D Complete Results

We present the results of all languages for each benchmark: PAWS-X (Table 16), XCOPA (Table 18), Self-Instruct* (Table 17), XL-Sum (Table 19, Table 20) and FLORES (Table 21, 22, 23, 24, 25, 26, 27 and Table 28).

	Language of inputs and outputs						
LLM	**En**	**Zh**	**Vi**	**Tr**	**Ar**	**El**	**Hi**
Understanding capability (Instruction identical to inputs)							
ChatGPT	**56.0**	20.5	26.8	18.3	24.1	17.7	0.6
LLaMA	**76.6**	27.2	36.6	27.8	11.8	22.3	14.3
BLOOM	**83.9**	83.0	79.9	27.4	79.2	22.8	82.7
Generation capability (Instructions in English)							
ChatGPT	**56.0**	37.1	36.1	34.5	32.0	29.7	17.5
LLaMA	**76.6**	66.3	42.9	38.1	24.2	40.7	30.8
BLOOM	**83.9**	81.8	79.2	27.6	77.2	49.2	80.8

Table 10: A primary evaluation for the multilingual capability (understanding and generation) of LLMs (ChatGPT, LLaMA and BLOOM) on the XQuAD dataset (Artetxe et al., 2020) in terms of exact match (EM). To evaluate the *understanding capability* we use an identical language to represent instructions and inputs to LLMs. To evaluate the *generating capability* instructions are always in English, regardless of the language of input text.

E Multilingual Capability of LLMs

Table 10 illustrates the significant variation in performance achieved by LLMs across different examined languages, with the highest scores observed when tasks are described and presented in English. Furthermore, LLMs exhibit varying degrees of understanding capabilities across different languages, with some cases where they fail to understand the

language. For instance, when using *Hindi* as the instruction language, ChatGPT lacks an understanding of the instructions, leading to subpar performance.

F Comparison with Fine-tuned SLM

While it is not feasible to evaluate a single small language model (SLM) across all the benchmarks we employed, we omit these results from Table 3. However, we do provide some experimental findings for reference. As shown in Table 14, our model does not perform as well as certain SLM fine-tuned on specific benchmarks (e.g., XL-Sum), we successfully narrow the performance gap between the original large language models (e.g., LLaMA) and closed-source large models (e.g., ChatGPT). This achievement is particularly meaningful as we refrain from using any task-specific data for fine-tuning. Interestingly, when SLM are not fine-tuned for a specific task, Table 15 demonstrates that our model significantly outperform the SLM in the Self-Instruct* benchmark.

#Lang.	Ins_trans	Response	#Lang.	Ins_trans	Response
afrikaans	Google Translate	ChatGPT	norwegian	Google Translate	ChatGPT
albanian	Google Translate	ChatGPT	persian	Google Translate	ChatGPT
amharic	Google Translate	Google Translate	polish	Google Translate	ChatGPT
arabic	Google Translate	ChatGPT	portuguese	Google Translate	ChatGPT
armenian	Google Translate	Google Translate	romanian	Google Translate	ChatGPT
azerbaijani	Google Translate	Google Translate	russian	Google Translate	ChatGPT
belarusian	Google Translate	Google Translate	serbian	Google Translate	Google Translate
bengali	Google Translate	Google Translate	sindhi	Google Translate	Google Translate
bosnian	Google Translate	ChatGPT	sinhala	Google Translate	Google Translate
bulgarian	Google Translate	ChatGPT	slovak	Google Translate	ChatGPT
catalan	Google Translate	ChatGPT	slovenian	Google Translate	ChatGPT
cebuano	Google Translate	ChatGPT	somali	Google Translate	Google Translate
chinese	Google Translate	ChatGPT	spanish	Google Translate	ChatGPT
croatian	Google Translate	ChatGPT	sundanese	Google Translate	Google Translate
czech	Google Translate	ChatGPT	swahili	Google Translate	ChatGPT
danish	Google Translate	ChatGPT	swedish	Google Translate	ChatGPT
dutch	Google Translate	ChatGPT	tamil	Google Translate	Google Translate
english	Google Translate	ChatGPT	thai	Google Translate	Google Translate
estonian	Google Translate	ChatGPT	turkish	Google Translate	Google Translate
finnish	Google Translate	ChatGPT	ukrainian	Google Translate	ChatGPT
french	Google Translate	ChatGPT	urdu	Google Translate	ChatGPT
galician	Google Translate	ChatGPT	uzbek	Google Translate	Google Translate
georgian	Google Translate	Google Translate	vietnamese	Google Translate	ChatGPT
german	Google Translate	ChatGPT	welsh	Google Translate	ChatGPT
gujarati	Google Translate	Google Translate	xhosa	Google Translate	Google Translate
hausa	Google Translate	Google Translate	yiddish	Google Translate	Google Translate
hebrew	Google Translate	ChatGPT	yoruba	Google Translate	Google Translate
hindi	Google Translate	ChatGPT	zulu	Google Translate	Google Translate
hungarian	Google Translate	ChatGPT	asturian	NLLB	ChatGPT
icelandic	Google Translate	ChatGPT	bashkir	NLLB	NLLB
igbo	Google Translate	Google Translate	breton	NLLB	NLLB
indonesian	Google Translate	ChatGPT	burmese	NLLB	NLLB
irish	Google Translate	ChatGPT	frisian	NLLB	NLLB
italian	Google Translate	ChatGPT	fulah	NLLB	NLLB
japanese	Google Translate	ChatGPT	gaelic	NLLB	NLLB
javanese	Google Translate	ChatGPT	ganda	NLLB	NLLB
kannada	Google Translate	Google Translate	greek	NLLB	ChatGPT
kazakh	Google Translate	Google Translate	haitian	NLLB	ChatGPT
korean	Google Translate	ChatGPT	iloko	NLLB	NLLB
lao	Google Translate	Google Translate	khmer	NLLB	NLLB
latvian	Google Translate	ChatGPT	lingala	NLLB	NLLB
lithuanian	Google Translate	ChatGPT	northern sotho	NLLB	NLLB
luxembourgish	Google Translate	ChatGPT	occitan	NLLB	ChatGPT
macedonian	Google Translate	ChatGPT	oriya	NLLB	NLLB
malagasy	Google Translate	Google Translate	panjabi	NLLB	NLLB
malay	Google Translate	ChatGPT	pashto	NLLB	NLLB
malayalam	Google Translate	Google Translate	swati	NLLB	NLLB
marathi	Google Translate	Google Translate	tagalog	NLLB	ChatGPT
mongolian	Google Translate	Google Translate	tswana	NLLB	NLLB
nepali	Google Translate	Google Translate	wolof	NLLB	NLLB

Table 11: The translator used for each language when translating the instruction described in Section 3.1.

Multilingual Instruction Datasets					
Tokenizer	Vocab size	Instruction tokens	Input tokens	Response (acc) tokens	Response (rej) tokens
m2m_100	128,104	19.25	37.85	180.35	-
LLaMA	32,000	38.49	65.16	370.44	-
BLOOM	251,680	25.20	44.62	239.76	-
Cross-Lingual Human Feedback Datasets					
m2m_100	128,104	20.31	20.54	100.96	101.2
LLaMA	32,000	52.32	49.27	259.08	260.61
BLOOM	251,680	20.08	20.17	98.38	99.24

Table 12: Statistics of average tokens in each instructions, inputs and outputs on our constructed datasets.

Benchmark	Prompt
PAWS-X	The following are two {lang_name} sentences. Sentence 1: {sentence1} Sentence 2: {sentence2} Does sentence 1 paraphrase sentence 2? yes or no?
XCOPA	Here is a premise: {premise}. What is the {question}? Help me to pick the more plausible option -A: {choice1}, -B: {choice2}
Self-Instruct	Instruction: {instruction}
XL-Sum	The following is an article. Article: {article} Please summarize the provided article.
FLORES	Translate the following {source_lang} sentence from {source_lang} to {target_lang}. {souce_lang} Sentence: {source_sentence} Please only return the translated {target_lang} text

Table 13: The prompt of each benchmark used in our experiments.

	Low			Medium			High		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
mT5-XXL	24.89	9.75	21.08	31.20	12.91	25.50	33.14	14.44	27.57
chatGPT	15.85	4.78	11.24	18.68	5.68	12.99	19.15	5.80	13.65
LLaMA	6.26	2.47	6.52	5.80	2.18	7.27	8.08	2.59	7.36
xLLMs-100	9.99	3.75	9.27	13.57	5.03	11.28	16.61	5.27	11.33

Table 14: Compare with mT5-XXL finetuned model in XL-Sum benchmark.

	Low			High		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
T5-LM	1.23	0.35	1.86	3.17	0.83	2.67
chatGPT	21.93	8.24	15.23	31.60	13.81	25.15
LLaMA	5.38	2.16	4.37	8.81	2.15	6.24
xLLMs-100	9.16	5.25	9.33	14.71	7.28	12.75

Table 15: Compare with T5-LM in Self-Instruct benchmark.

Generating Capability								
	de	en	es	fr	ja	ko	zh	Avg.
LLaMA	45.65	62.91	50.06	46.55	46.40	48.00	51.98	50.22
BX_{LLaMA}	44.50	63.43	46.43	45.15	45.05	46.40	47.88	48.41
SFT_{LLaMA}	47.80	65.44	48.32	48.20	47.15	47.75	47.85	50.36
xLLMs-100	**55.25**	**70.68**	**70.68**	**54.85**	**55.70**	**55.15**	**71.25**	**61.94**
BLOOM	45.00	62.35	45.45	45.15	44.10	44.80	44.85	47.39
BX_{BLOOM}	44.50	62.49	45.43	45.25	43.95	44.40	44.81	47.26
SFT_{BLOOM}	46.50	63.44	46.18	46.10	45.45	45.85	46.00	48.50
xLLMs-100	**47.80**	**65.21**	**49.49**	**48.10**	**47.10**	**47.70**	**48.29**	**50.53**
Understanding Capability								
	de	en	es	fr	ja	ko	zh	Avg.
LLaMA	32.61	62.91	42.43	33.43	33.53	30.93	30.86	38.10
BX_{LLaMA}	34.23	63.43	33.70	31.68	**35.73**	30.93	**31.24**	37.28
SFT_{LLaMA}	**34.51**	65.44	69.80	31.20	33.47	30.94	30.89	42.32
xLLMs-100	31.27	**70.68**	**70.68**	**63.49**	30.72	**30.94**	30.89	**46.95**
BLOOM	35.00	62.35	**31.84**	31.77	32.59	30.86	30.89	36.47
BX_{BLOOM}	34.83	62.49	31.20	31.28	32.79	31.20	**31.12**	36.42
SFT_{BLOOM}	**35.13**	63.44	31.29	32.54	32.41	30.99	30.92	36.67
xLLMs-100	34.08	**65.21**	31.20	**49.69**	**33.84**	**33.86**	30.92	**39.83**

Table 16: Accuracy scores on the PAWS-X benchmark (Yang et al., 2019).

Generating Capability												
	ar	cs	de	zh	hi	hy	ky	yo	ta	mn	Avg_L	Avg_H
LLaMA	7.09	7.95	8.70	15.23	5.10	5.39	4.85	6.95	5.44	4.28	5.38	8.81
BX_{LLaMA}	9.11	7.57	9.51	14.27	8.53	5.52	6.94	10.43	5.79	6.37	7.01	9.80
SFT_{LLaMA}	10.06	8.54	9.77	19.40	12.98	7.77	5.27	9.73	7.62	5.11	7.10	12.15
xLLMs-100	12.44	12.78	15.49	19.90	12.94	3.79	9.28	11.48	10.82	10.42	**9.16**	**14.71**
BLOOM	6.52	4.90	6.12	11.13	6.41	2.37	3.44	5.99	5.20	3.34	4.07	7.01
BX_{BLOOM}	5.74	7.08	9.40	12.26	6.55	3.32	6.58	8.15	5.34	6.02	5.88	8.21
SFT_{BLOOM}	6.30	11.63	13.35	18.24	8.04	7.24	7.18	9.10	7.89	7.47	7.78	11.51
xLLMs-100	11.12	10.69	13.36	19.22	13.71	10.51	9.24	12.72	9.48	8.88	**10.17**	**13.62**
Understanding Capability												
	ar	cs	de	zh	hi	hy	ky	yo	ta	mn	Avg_L	Avg_H
LLaMA	6.71	12.64	13.52	20.19	9.79	5.59	4.50	14.82	5.08	5.45	7.09	12.57
BX_{LLaMA}	8.89	7.88	11.99	21.09	9.54	5.31	4.36	11.64	4.88	5.37	6.31	11.88
SFT_{LLaMA}	10.67	9.33	13.25	17.87	12.47	19.10	7.47	12.29	8.76	7.90	7.32	12.72
xLLMs-100	13.69	12.90	15.31	21.31	13.56	23.19	8.94	13.35	9.31	9.93	**12.94**	**15.35**
BLOOM	9.41	6.97	12.29	2.80	11.87	7.85	5.05	11.33	8.03	5.56	7.56	8.67
BX_{BLOOM}	7.22	6.50	7.66	4.43	14.72	0.00	1.06	7.70	9.86	5.43	4.81	8.11
SFT_{BLOOM}	8.89	7.88	11.99	21.10	9.54	5.31	4.36	11.63	4.88	5.37	6.31	11.88
xLLMs-100	12.06	8.35	13.07	22.45	10.83	6.48	5.16	13.31	8.33	6.43	**7.94**	**13.35**

Table 17: ROUGE-1 scores on the Self-Instruct* benchmark (Wang et al., 2023) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages.

Generating Capability													
	et	ht	id	it	qu	sw	ta	th	tr	vi	zh	Avg_L	Avg_H
LLaMA	50.00	49.80	51.00	51.80	48.40	50.40	49.80	52.40	50.60	50.80	55.20	49.33	51.52
BX_{LLaMA}	50.60	47.60	49.40	49.40	48.60	50.20	47.80	49.20	49.00	50.60	50.40	48.00	49.85
SFT_{LLaMA}	50.40	50.00	49.60	50.00	47.00	50.00	49.80	50.00	50.20	50.20	50.00	48.93	50.05
xLLMs-100	50.00	51.00	54.80	60.60	48.20	52.60	49.93	53.20	54.00	51.20	61.00	49.71	**54.68**
BLOOM	48.00	49.16	47.84	48.27	50.22	50.63	50.16	50.83	50.72	49.37	50.11	49.85	49.47
BX_{BLOOM}	48.14	43.57	50.60	49.32	49.80	50.20	49.80	50.60	50.40	48.88	51.67	47.72	49.98
SFT_{BLOOM}	49.00	50.00	48.80	49.20	46.80	47.80	50.60	49.20	47.40	51.00	51.80	49.13	49.28
xLLMs-100	51.22	51.59	52.35	51.42	52.83	51.28	52.66	51.50	51.73	53.26	55.34	52.36	**52.26**
Understanding Capability													
	et	ht	id	it	qu	sw	ta	th	tr	vi	zh	Avg_L	Avg_H
LLaMA	47.83	48.24	51.80	50.23	50.60	48.52	43.47	42.57	45.14	46.37	45.26	47.44	47.22
BX_{LLaMA}	48.59	49.17	52.26	50.84	51.06	49.27	48.36	46.07	47.49	48.93	48.58	49.53	49.00
SFT_{LLaMA}	49.26	50.28	52.85	51.44	51.59	50.17	48.70	48.26	48.37	49.34	49.15	50.19	49.86
xLLMs-100	52.80	49.60	53.20	53.17	52.48	51.00	52.51	50.75	50.37	51.75	52.66	51.53	**51.96**
BLOOM	46.15	43.27	50.14	46.25	48.28	47.34	41.25	48.74	48.37	53.16	52.96	44.27	49.14
BX_{BLOOM}	47.26	45.14	52.68	47.85	49.94	48.50	43.77	49.16	48.93	54.25	54.16	46.28	50.35
SFT_{BLOOM}	50.14	49.27	54.35	50.22	52.63	50.48	46.37	51.06	50.45	56.47	55.29	49.42	52.31
xLLMs-100	52.70	51.20	58.96	52.97	54.55	53.29	51.75	53.26	54.25	60.37	58.93	52.50	**55.59**

Table 18: Accuracy scores on the XCOPA benchmark (Ponti et al., 2020).

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
marathi	High	0.64	0.01	4.00	5.77	0.90	1.97	3.12	6.48
english	High	16.43	19.06	18.29	20.77	12.66	15.75	12.19	16.91
pashto	High	17.17	0.67	17.21	19.02	22.03	8.26	7.96	21.20
thai	Medium	15.55	8.35	15.69	18.26	9.47	5.02	8.89	10.37
ukrainian	High	1.24	0.50	1.25	10.93	4.87	2.05	6.77	8.13
serbian_cyrillic	Medium	1.73	0.47	0.56	9.65	3.76	1.40	12.23	10.33
spanish	High	6.37	3.86	4.40	21.43	14.76	10.49	23.99	10.70
japanese	Medium	3.45	0.72	14.03	11.54	8.82	0.36	18.32	11.35
french	Medium	8.21	4.37	6.11	24.71	16.07	4.49	20.63	19.59
vietnamese	High	13.01	1.60	10.70	12.08	8.23	7.30	7.94	18.08
punjabi	Medium	3.36	0.07	2.80	4.57	12.18	0.43	13.23	8.60
chinese_simplified	High	7.99	4.75	11.89	18.66	11.67	6.69	9.97	10.86
kyrgyz	Low	1.21	0.19	1.66	7.78	4.75	0.71	3.62	5.19
chinese_traditional	High	8.76	4.15	10.36	16.62	13.09	6.16	13.99	9.81
kirundi	Low	6.92	3.20	5.12	16.40	11.89	2.50	4.38	16.10
turkish	High	6.37	1.81	9.85	16.61	3.96	7.82	9.27	14.26
oromo	Medium	3.23	1.02	3.10	11.19	3.24	1.30	1.70	7.40
hausa	Medium	9.95	3.96	9.50	19.66	8.49	5.56	7.75	15.32
somali	Low	8.65	2.19	6.26	18.41	9.09	3.00	2.94	14.56
telugu	High	3.13	1.26	4.38	10.17	7.98	0.83	3.21	5.45
arabic	High	6.98	0.19	10.51	18.06	3.31	3.23	6.19	11.57
serbian_latin	Medium	1.60	1.18	1.56	11.56	4.76	5.39	7.41	9.77
uzbek	Low	0.50	0.14	1.23	5.47	5.77	3.56	2.45	5.62
hindi	High	8.07	0.11	17.84	20.33	2.15	1.61	16.76	13.20
indonesian	High	8.10	3.26	5.45	21.67	8.62	6.23	10.87	15.43
azerbaijani	Medium	3.30	0.84	4.69	9.69	3.75	5.50	3.77	7.65
tamil	High	3.22	0.06	3.48	3.89	3.07	2.08	1.52	4.69
portuguese	High	8.67	3.49	5.50	23.56	11.36	6.32	20.02	12.69
igbo	Low	14.72	2.55	6.99	17.18	6.71	4.81	4.22	15.34
burmese	Low	16.14	0.63	4.98	5.75	6.17	0.68	7.32	14.06
gujarati	Medium	1.18	0.24	0.83	6.26	3.99	0.26	6.76	6.29
bengali	Medium	1.78	0.05	5.73	4.79	5.80	1.06	4.28	4.71
pidgin	Medium	10.28	13.39	13.45	16.64	8.91	10.85	9.45	18.31
amharic	Low	3.34	0.03	2.03	2.03	3.83	0.58	7.12	1.85
yoruba	Medium	10.36	2.38	5.60	16.70	8.29	4.36	4.73	13.09
welsh	Medium	5.49	1.26	3.24	18.42	6.99	2.51	5.56	16.39
urdu	High	15.35	0.35	17.98	19.20	6.57	1.12	14.34	12.27
persian	High	9.78	0.18	14.81	19.06	16.74	0.32	8.56	18.54
swahili	Medium	7.54	2.79	3.96	19.90	11.97	5.32	8.32	16.91
tigrinya	Low	4.54	0.08	1.73	7.15	8.53	0.83	2.37	4.17
scottish_gaelic	Low	6.25	3.54	8.43	21.92	4.98	4.74	5.27	15.55
sinhala	Low	3.88	0.06	2.58	3.40	0.38	0.52	2.45	3.73
nepali	Low	5.75	0.03	7.56	6.31	7.82	1.35	3.15	5.29
korean	Low	3.18	0.68	5.60	8.05	3.03	0.47	1.45	3.75
russian	High	5.15	0.92	1.74	14.47	4.16	2.71	12.83	10.93
Avg_L		6.26	1.11	4.51	**9.99**	6.08	1.98	3.89	**8.77**
Avg_M		5.80	2.74	6.06	**13.57**	7.77	3.59	8.87	**11.74**
Avg_H		8.08	1.70	9.21	**16.61**	8.91	4.58	10.89	**12.36**

Table 19: **Generating Capability:** ROUGE-1 scores on the XL-Sum benchmark (Hasan et al., 2021) and average scores on low-resource (Avg_L), medium-resource (Avg_M) and high-resource (Avg_H) languages.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
marathi	High	0.70	1.55	4.65	4.99	10.45	7.84	8.01	12.43
english	High	20.06	19.07	18.28	18.74	13.37	15.78	12.19	18.65
pashto	High	5.81	12.74	9.92	21.49	18.72	0.02	7.86	20.76
thai	Medium	7.92	11.48	13.48	18.65	13.47	2.90	7.69	11.37
ukrainian	High	0.69	0.88	2.23	12.10	9.06	4.21	7.61	10.61
serbian_cyrillic	Medium	0.66	4.96	0.84	5.68	9.81	4.07	8.63	11.75
spanish	High	7.77	17.11	4.99	22.43	19.42	20.14	26.30	21.47
japanese	Medium	1.33	4.61	10.74	30.76	26.76	3.80	23.18	28.70
french	Medium	5.74	17.06	7.18	23.78	21.35	21.47	19.97	23.36
vietnamese	High	2.57	2.93	5.30	14.97	15.65	17.14	12.18	17.70
punjabi	Medium	6.56	0.05	2.61	3.17	16.17	1.01	14.79	18.16
chinese_simplified	High	3.59	8.61	9.80	24.96	19.25	15.52	21.44	21.30
kyrgyz	Low	1.15	0.38	1.18	6.28	4.38	1.39	5.16	6.36
chinese_traditional	High	3.69	6.72	9.14	21.82	19.98	14.94	22.22	21.94
kirundi	Low	6.24	0.19	5.43	11.85	18.61	13.56	6.28	17.58
turkish	High	2.64	9.85	7.03	14.25	14.75	6.30	8.30	16.73
oromo	Medium	3.79	1.87	4.61	10.43	5.44	0.44	0.27	7.49
hausa	Medium	9.23	5.90	7.79	19.31	11.04	7.63	6.96	13.11
somali	Low	5.19	1.44	6.88	14.91	20.62	5.39	1.76	11.49
telugu	High	0.84	0.46	7.19	13.25	8.95	5.03	3.84	10.95
arabic	High	0.72	6.68	9.69	16.25	17.07	10.92	14.52	19.04
serbian_latin	Medium	1.54	5.28	2.83	6.43	11.55	6.83	10.46	13.56
uzbek	Low	0.42	0.02	0.97	3.47	2.29	2.04	1.14	4.29
hindi	High	0.10	11.19	12.50	18.81	20.69	20.94	21.65	22.69
indonesian	High	3.98	5.88	5.63	20.12	16.30	15.72	17.26	18.31
azerbaijani	Medium	2.02	2.83	4.08	7.53	9.40	5.86	5.45	10.62
tamil	High	4.43	0.46	3.75	2.70	7.38	10.84	2.84	9.37
portuguese	High	5.04	21.83	7.56	20.50	20.60	20.28	19.74	22.61
igbo	Low	8.00	6.34	10.15	18.98	13.95	15.91	10.14	15.95
burmese	Low	9.26	4.43	5.54	6.92	8.16	0.65	5.27	8.07
gujarati	Medium	0.57	0.18	9.67	13.76	8.39	3.88	6.28	10.69
bengali	Medium	0.56	0.77	4.54	5.14	10.45	14.60	9.39	12.50
pidgin	Medium	14.27	11.84	15.06	13.80	17.48	14.14	11.58	19.53
amharic	Low	2.97	0.42	1.59	1.93	3.60	0.52	5.24	13.85
yoruba	Medium	9.36	2.51	6.26	16.24	19.94	12.04	5.61	13.91
welsh	Medium	5.41	5.25	11.80	16.20	14.77	13.91	8.76	16.84
urdu	High	2.56	7.27	14.76	17.36	21.77	15.81	20.08	23.75
persian	High	0.21	7.04	9.22	18.86	23.22	4.88	9.53	20.32
swahili	Medium	12.59	8.27	8.61	17.57	14.87	14.53	12.78	16.84
tigrinya	Low	4.70	2.10	1.23	4.74	4.58	2.47	11.27	16.61
scottish_gaelic	Low	5.69	4.99	12.73	18.69	14.07	6.64	6.82	16.08
sinhala	Low	4.23	1.04	1.52	2.40	2.28	1.26	6.37	14.28
nepali	Low	0.31	1.73	5.45	6.55	9.92	8.69	5.55	11.91
korean	Low	0.68	2.96	3.76	9.21	5.94	0.19	2.49	17.99
russian	High	0.84	6.59	2.01	16.80	16.00	4.72	13.96	16.50
Avg_L		4.07	2.17	4.70	**8.83**	9.03	4.89	5.62	**12.87**
Avg_M		5.44	5.52	7.34	**13.90**	14.06	8.47	10.12	**15.23**
Avg_H		2.84	7.89	7.55	**17.29**	16.80	11.71	14.33	**18.38**

Table 20: **Understanding Capability:** ROUGE-1 scores on the XL-Sum benchmark (Hasan et al., 2021) and average scores on low-resource (Avg_L), medium-resource (Avg_M) and high-resource (Avg_H) languages.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
amh_Ethi	Low	0.33	0.11	0.52	0.26	0.18	0.13	0.62	2.82
arb_Arab	High	2.14	2.65	2.25	3.85	0.28	0.40	2.25	5.29
asm_Beng	Low	0.30	0.97	0.59	0.63	0.35	0.10	1.30	3.59
ast_Latn	Low	2.70	3.13	5.56	6.76	1.79	1.26	5.57	5.69
azj_Latn	Low	1.36	1.26	2.37	2.41	0.38	0.39	2.30	3.71
bel_Cyrl	Low	1.60	1.71	1.95	2.81	0.61	0.31	1.92	3.94
ben_Beng	High	1.03	0.94	1.00	1.06	0.37	0.11	1.34	5.09
bos_Latn	High	3.43	5.58	3.72	8.63	0.86	0.83	2.79	4.46
bul_Cyrl	High	4.55	7.87	3.52	11.65	0.89	0.60	2.36	4.63
cat_Latn	High	7.76	8.03	6.50	19.24	5.98	3.27	5.01	14.83
ceb_Latn	Low	3.04	3.95	4.51	6.79	1.30	0.46	4.37	4.79
ces_Latn	High	4.52	7.10	5.38	10.79	0.96	0.77	3.01	4.13
ckb_Arab	Low	0.73	1.15	1.82	1.41	0.29	0.06	1.91	3.27
cym_Latn	Low	1.94	1.69	2.95	3.48	0.88	0.54	3.30	3.99
dan_Latn	High	7.63	9.01	7.10	19.85	1.19	0.81	3.51	5.27
deu_Latn	High	6.15	7.56	9.58	17.39	1.57	0.79	4.36	5.70
ell_Grek	High	1.86	2.19	1.78	2.97	0.75	0.78	1.74	3.95
est_Latn	High	1.85	2.37	3.00	3.41	0.65	0.26	2.61	3.92
fin_Latn	High	3.57	4.91	2.57	7.61	0.67	0.65	2.36	3.87
fra_Latn	High	11.37	15.67	17.63	26.14	4.20	1.92	7.79	14.02
fuv_Latn	Low	1.45	1.36	3.17	2.09	0.95	0.81	3.33	3.52
gle_Latn	Low	1.96	1.74	3.20	3.33	1.05	0.77	2.97	3.92
glg_Latn	Low	3.55	5.17	5.38	9.34	2.54	2.01	7.56	7.78
guj_Gujr	Low	0.59	0.62	1.56	0.40	0.56	0.33	2.04	4.58
hau_Latn	Low	1.27	1.08	3.01	2.45	0.89	0.20	2.73	3.53
heb_Hebr	High	1.48	1.91	2.10	3.09	0.35	0.28	1.69	3.63
hin_Deva	High	2.50	2.65	1.53	4.09	0.66	0.45	1.92	7.59
hrv_Latn	High	4.20	6.11	3.32	9.93	1.05	0.83	2.94	4.21
hun_Latn	High	3.95	6.14	3.75	8.61	0.88	1.04	2.54	4.05
hye_Armn	Low	0.97	1.45	1.12	1.22	0.35	0.47	1.63	3.97
ibo_Latn	Low	0.90	1.12	2.62	2.16	0.98	0.73	2.92	3.60
ind_Latn	High	7.20	11.31	3.84	17.97	1.97	0.85	4.98	11.31
isl_Latn	High	1.90	1.90	2.93	4.01	0.91	0.96	2.83	4.21
ita_Latn	High	5.75	8.00	9.88	15.04	1.85	0.94	3.51	6.33
jav_Latn	Low	1.81	2.04	3.04	4.30	0.85	0.49	3.11	4.29
jpn_Jpan	High	0.02	0.22	0.04	0.05	0.03	0.00	0.04	2.51
kam_Latn	Low	1.15	0.89	2.89	2.46	0.83	0.54	2.91	3.87
kan_Knda	Low	0.43	0.33	1.50	0.56	0.65	0.05	2.26	4.13
kat_Geor	Low	1.01	0.86	1.86	2.44	0.36	0.33	2.03	3.69
kaz_Cyrl	High	0.85	1.22	1.76	2.52	0.61	0.26	2.46	3.21
khm_Khmr	Low	0.60	0.00	0.72	0.03	0.51	0.18	1.26	3.15
kir_Cyrl	Low	1.16	1.04	1.91	2.31	0.53	0.19	2.26	3.53
kor_Hang	High	2.44	3.63	2.08	4.11	0.42	0.37	2.13	3.50
lao_Laoo	Low	0.41	0.08	2.21	4.44	0.78	0.58	2.81	4.29
lij_Latn	Low	1.63	1.23	3.24	3.21	1.42	0.57	3.07	3.95
lim_Latn	Low	1.91	2.39	3.66	4.50	1.28	0.45	3.49	3.84
lin_Latn	Low	1.52	1.84	2.98	2.70	1.08	0.72	3.36	4.01
lit_Latn	High	1.82	2.21	2.45	3.48	0.67	0.54	2.71	3.85
ltz_Latn	Low	1.70	2.29	2.99	3.27	0.59	0.63	2.76	4.15
lug_Latn	Low	1.34	0.79	3.42	3.21	1.00	0.50	3.03	4.11
Avg_L		1.35	1.56	2.42	**2.89**	0.78	0.47	2.59	**3.97**
Avg_H		3.90	5.33	4.56	**9.07**	1.20	0.82	3.12	**5.79**

Table 21: **Generating Capability (Part I):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from English to into other languages.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
luo_Latn	Low	1.27	0.61	2.73	2.43	0.95	0.45	2.12	3.93
lvs_Latn	High	1.56	2.00	2.62	3.33	0.55	0.42	2.55	3.96
mal_Mlym	Low	0.68	0.71	1.94	1.48	0.64	0.53	2.41	3.58
mar_Deva	Low	1.20	1.47	1.63	2.06	0.59	0.33	2.22	4.09
mkd_Cyrl	High	2.16	3.09	2.54	5.43	0.70	0.13	2.02	4.11
mlt_Latn	High	1.85	1.81	3.34	3.82	0.98	0.57	3.40	3.65
khk_Cyrl	Low	0.47	0.02	2.16	1.61	0.46	0.25	1.99	3.63
mri_Latn	Low	1.84	2.54	1.95	4.19	1.02	0.68	2.50	4.04
mya_Mymr	Low	0.04	0.00	0.41	0.10	0.21	0.09	0.60	2.80
nld_Latn	High	5.51	8.47	11.45	15.05	1.70	1.10	4.79	5.01
nob_Latn	Low	6.36	9.80	4.82	16.48	0.95	0.90	3.19	5.05
npi_Deva	Low	1.48	1.38	1.38	1.35	0.41	0.19	1.53	4.33
nso_Latn	Low	1.49	1.42	3.92	2.49	1.22	0.96	3.93	4.16
nya_Latn	Low	1.43	0.80	3.09	3.51	1.01	0.31	3.46	3.82
oci_Latn	Low	2.44	2.94	3.44	5.54	1.63	1.16	3.95	4.36
gaz_Latn	Low	0.66	0.43	2.34	2.37	0.60	0.38	2.16	3.53
ory_Orya	Low	0.19	0.35	0.91	0.45	0.49	0.20	1.52	4.37
pan_Guru	Low	0.59	0.84	1.36	0.54	0.47	0.13	1.49	3.05
pes_Arab	High	2.06	2.20	2.08	3.19	0.50	0.44	1.94	3.43
pol_Latn	High	3.85	4.78	4.46	7.36	0.72	0.85	2.79	4.24
por_Latn	High	8.95	14.07	8.17	23.75	5.38	2.29	5.96	13.73
pbt_Arab	Low	0.69	1.61	1.66	1.78	0.52	0.20	1.66	3.47
ron_Latn	High	6.57	9.57	12.30	14.97	1.09	0.94	3.67	4.61
rus_Cyrl	High	4.80	8.05	9.55	13.63	0.70	0.66	2.74	5.62
slk_Latn	High	2.92	3.45	2.95	5.29	0.87	0.79	2.77	4.31
sna_Latn	Low	1.17	1.38	3.35	2.94	0.83	0.61	1.98	3.82
snd_Arab	Low	0.44	1.58	1.53	1.27	0.53	0.18	2.08	3.13
som_Latn	Low	1.44	1.44	2.69	2.59	0.88	0.71	3.11	3.92
spa_Latn	High	7.81	8.80	12.14	16.00	3.46	2.72	5.66	9.92
srp_Cyrl	Low	3.82	7.04	2.40	7.95	0.57	0.49	2.38	4.16
swe_Latn	High	7.86	9.20	13.39	19.23	0.80	0.78	3.46	4.81
swh_Latn	High	1.35	1.37	2.97	3.36	0.78	0.65	3.16	5.15
tam_Taml	Low	0.45	0.98	1.45	1.82	0.69	0.20	2.55	4.70
tel_Telu	Low	0.20	0.22	1.64	0.83	0.74	0.54	2.75	3.33
tgk_Cyrl	Low	0.68	0.22	2.17	1.54	0.34	0.20	2.14	3.59
tgl_Latn	High	4.31	5.28	6.21	8.15	1.19	1.21	4.09	4.73
tha_Thai	High	0.93	0.80	0.54	1.92	0.29	0.06	0.93	3.18
tur_Latn	High	2.06	2.67	2.85	3.47	0.74	0.61	2.74	3.92
ukr_Cyrl	High	3.88	5.91	3.01	12.10	0.56	1.05	2.25	4.25
umb_Latn	Low	1.22	1.04	2.51	2.42	0.67	0.57	2.51	3.22
urd_Arab	Low	1.69	1.64	1.11	2.26	0.25	0.18	0.80	5.62
uzn_Latn	High	0.95	1.33	2.31	2.66	0.53	0.50	2.20	3.59
vie_Latn	High	7.49	11.59	2.55	11.83	1.93	1.09	3.66	14.71
wol_Latn	Low	1.46	1.18	2.17	2.01	0.64	0.14	2.67	3.71
xho_Latn	High	1.12	1.64	2.91	2.80	0.73	0.59	2.77	3.95
yor_Latn	Low	0.76	0.71	2.51	2.43	0.89	0.68	2.92	3.49
zho_Hans	High	8.22	11.32	0.99	15.38	1.83	1.17	6.14	12.24
zho_Hant	High	4.16	7.41	0.73	13.36	0.82	0.64	2.44	7.41
zsm_Latn	High	4.23	5.11	3.82	11.95	1.19	1.02	4.87	8.52
zul_Latn	High	0.91	1.31	2.53	2.55	0.77	0.59	2.81	3.50
Avg_L		1.35	1.56	2.42	**2.89**	0.78	0.47	2.59	**3.97**
Avg_H		3.90	5.33	4.56	**9.07**	1.20	0.82	3.12	**5.79**

Table 22: **Generating Capability (Part II):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from English to into other languages.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
amh_Ethi	Low	1.00	0.57	1.29	1.83	0.20	0.34	1.84	0.66
arb_Arab	High	3.19	1.08	3.16	10.80	2.89	3.37	3.18	14.78
asm_Beng	Low	1.41	0.34	1.17	2.29	0.62	2.89	2.96	5.32
ast_Latn	Low	5.38	2.17	11.02	21.11	1.46	3.64	4.98	15.45
azj_Latn	Low	1.71	1.86	1.88	3.79	1.14	1.76	1.19	2.32
bel_Cyrl	Low	2.39	1.53	1.87	6.32	1.18	1.72	1.51	1.71
ben_Beng	High	1.65	0.44	1.23	3.13	0.89	2.28	3.56	12.00
bos_Latn	High	7.33	1.78	8.68	25.84	1.16	2.06	1.78	5.19
bul_Cyrl	High	7.13	1.36	14.65	25.71	0.98	1.64	2.14	5.16
cat_Latn	High	8.93	1.63	18.66	31.92	2.33	2.52	3.75	20.24
ceb_Latn	Low	2.55	2.57	2.99	8.96	1.72	2.03	2.28	4.14
ces_Latn	High	7.28	2.41	8.50	26.21	1.28	2.01	2.21	5.02
ckb_Arab	Low	1.26	1.16	1.22	2.27	0.46	1.27	1.09	1.60
cym_Latn	Low	1.60	2.00	1.37	5.90	1.36	1.91	1.72	1.89
dan_Latn	High	9.61	1.78	14.79	31.44	1.09	2.85	3.80	6.07
deu_Latn	High	9.12	1.59	15.03	31.77	1.73	3.47	5.19	12.29
ell_Grek	High	2.54	1.11	2.64	11.13	1.43	1.61	1.55	2.35
est_Latn	High	1.89	1.89	3.10	6.07	1.13	2.01	1.34	2.52
fin_Latn	High	4.71	1.81	4.20	21.15	1.29	1.82	1.23	2.77
fra_Latn	High	9.94	2.32	19.79	34.70	4.34	3.54	3.63	22.52
fuv_Latn	Low	1.45	1.60	1.96	2.69	1.43	2.39	1.41	2.45
gle_Latn	Low	1.79	1.71	1.88	5.73	1.17	1.40	1.62	2.36
glg_Latn	Low	7.81	1.73	13.62	24.74	0.98	2.36	4.61	10.76
guj_Gujr	Low	0.93	0.24	1.08	1.92	0.61	2.05	2.39	11.12
hau_Latn	Low	1.48	1.75	1.84	2.98	0.76	1.86	1.40	2.33
heb_Hebr	High	2.06	1.09	3.77	7.70	1.34	1.78	1.08	3.89
hin_Deva	High	1.95	0.92	1.32	7.16	1.91	1.93	4.21	14.65
hrv_Latn	High	6.64	1.57	9.55	24.46	1.13	2.02	2.09	4.56
hun_Latn	High	5.49	1.80	7.37	20.07	1.24	1.80	1.50	2.60
hye_Armn	Low	1.21	0.77	1.13	2.20	0.06	1.13	1.15	1.38
ibo_Latn	Low	1.82	1.77	2.72	2.42	0.85	1.57	1.65	2.81
ind_Latn	High	6.88	2.07	4.08	25.75	2.43	2.64	4.86	19.78
isl_Latn	High	2.05	1.97	2.37	7.00	1.30	2.16	1.65	2.43
ita_Latn	High	8.32	1.55	14.56	24.42	2.47	2.00	5.27	11.75
jav_Latn	Low	2.10	1.75	1.88	4.11	1.04	1.80	2.36	3.50
jpn_Jpan	High	4.34	0.71	3.19	13.33	0.77	1.25	2.31	5.66
kam_Latn	Low	1.43	1.37	2.25	2.64	1.28	2.45	1.49	1.99
kan_Knda	Low	1.10	0.44	1.14	1.88	0.47	2.71	2.86	6.65
kat_Geor	Low	1.59	1.07	2.16	3.25	0.34	1.22	1.71	1.89
kaz_Cyrl	High	1.48	1.67	2.30	3.43	1.40	2.00	1.30	2.32
khm_Khmr	Low	1.13	0.09	0.89	2.38	0.37	0.46	1.58	2.07
kir_Cyrl	Low	1.40	1.53	1.87	3.18	1.11	1.81	1.29	1.69
kor_Hang	High	3.77	1.97	2.70	13.67	1.12	1.88	1.66	3.86
lao_Laoo	Low	0.89	0.05	1.45	2.75	0.23	0.31	0.87	1.84
lij_Latn	Low	2.76	1.80	4.54	9.48	1.17	2.50	3.41	4.96
lim_Latn	Low	3.85	2.77	4.56	13.07	1.17	2.99	2.81	4.00
lin_Latn	Low	1.67	1.46	1.42	3.21	0.87	1.47	1.40	2.28
lit_Latn	High	1.80	2.01	3.31	5.93	1.01	1.73	1.48	2.81
ltz_Latn	Low	3.01	1.79	3.07	8.00	1.17	2.15	2.26	2.11
lug_Latn	Low	1.57	1.98	1.67	2.97	1.30	2.35	1.37	2.46
Avg_L		2.11	1.37	2.71	**5.64**	0.99	1.95	2.05	**4.22**
Avg_H		4.95	1.61	7.29	**16.98**	1.49	2.33	2.56	**7.68**

Table 23: **Generating Capability (Part I):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from the other languages into English.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
luo_Latn	Low	1.37	1.42	1.59	2.64	1.21	2.32	1.32	2.16
lvs_Latn	High	1.90	1.67	3.67	4.36	0.91	1.81	1.34	3.31
mal_Mlym	Low	1.40	1.04	1.20	2.10	0.62	2.68	2.94	6.26
mar_Deva	Low	1.57	1.18	1.18	3.09	0.99	2.33	2.64	10.29
mkd_Cyrl	High	4.91	1.61	10.99	18.43	0.89	1.62	2.24	3.59
mlt_Latn	High	2.20	2.55	2.95	6.69	1.56	2.86	2.05	2.60
khk_Cyrl	Low	1.28	1.61	1.77	2.27	1.04	1.71	1.31	1.84
mri_Latn	Low	1.54	1.41	1.58	3.74	0.93	1.37	1.46	2.59
mya_Mymr	Low	1.04	0.25	1.18	1.76	0.70	0.09	0.89	0.79
nld_Latn	High	6.27	1.98	12.48	23.54	1.49	2.70	3.70	6.17
nob_Latn	Low	8.83	2.14	8.71	31.52	1.28	2.51	3.41	4.41
npi_Deva	Low	1.61	1.25	1.24	3.53	0.52	1.67	2.80	10.97
nso_Latn	Low	1.75	2.50	2.15	4.51	1.28	1.97	1.80	3.21
nya_Latn	Low	1.50	1.62	1.80	3.00	1.70	2.55	1.70	2.73
oci_Latn	Low	6.98	1.89	10.30	23.03	2.39	3.12	5.93	11.08
gaz_Latn	Low	1.24	1.58	1.35	1.86	1.06	1.79	1.09	1.60
ory_Orya	Low	1.03	0.17	0.98	1.88	0.90	2.43	2.43	6.94
pan_Guru	Low	1.11	0.24	1.15	1.93	0.94	2.28	1.99	9.02
pes_Arab	High	2.41	0.87	2.34	7.21	0.84	1.65	1.53	4.86
pol_Latn	High	6.23	1.77	7.55	18.86	1.28	2.08	2.02	4.79
por_Latn	High	9.61	2.03	20.51	35.16	2.61	2.46	4.29	25.06
pbt_Arab	Low	1.37	1.13	1.72	2.64	0.66	0.92	0.58	2.20
ron_Latn	High	7.99	1.83	12.34	29.17	1.50	2.22	3.72	5.27
rus_Cyrl	High	7.71	1.52	9.83	24.94	1.39	1.70	2.43	8.95
slk_Latn	High	5.94	1.88	5.57	20.97	1.14	2.12	2.22	3.00
sna_Latn	Low	1.84	2.01	1.86	3.48	1.27	2.18	1.64	2.26
snd_Arab	Low	1.21	1.70	1.25	2.38	0.40	1.28	1.03	2.00
som_Latn	Low	1.61	1.57	1.86	3.08	1.00	2.30	1.33	2.37
spa_Latn	High	6.97	1.56	15.30	23.22	1.15	1.59	2.18	14.17
srp_Cyrl	Low	7.30	1.63	16.41	28.93	1.11	1.83	1.65	4.76
swe_Latn	High	9.82	1.91	20.49	32.88	1.15	2.78	3.05	7.62
swh_Latn	High	1.41	1.71	2.99	4.09	1.15	2.36	2.65	11.32
tam_Taml	Low	1.50	0.91	1.28	2.23	0.66	3.24	2.94	9.13
tel_Telu	Low	1.02	0.44	1.30	1.94	1.19	1.74	3.28	8.42
tgk_Cyrl	Low	1.34	1.57	1.38	2.63	1.25	1.58	1.31	1.71
tgl_Latn	High	3.18	1.89	6.26	14.34	1.46	1.78	2.18	3.97
tha_Thai	High	1.62	0.16	1.27	4.15	0.64	0.41	1.61	2.81
tur_Latn	High	2.72	2.14	2.86	11.36	1.24	1.94	1.31	2.63
ukr_Cyrl	High	7.43	1.47	13.18	26.37	1.21	1.73	1.91	6.43
umb_Latn	Low	1.31	1.78	1.50	3.00	1.10	2.12	1.34	2.35
urd_Arab	Low	1.50	0.63	1.18	3.53	1.37	3.02	3.73	6.92
uzn_Latn	High	1.50	1.55	1.68	2.76	1.13	1.75	1.27	2.15
vie_Latn	High	5.17	2.00	2.41	18.55	2.39	3.29	4.51	15.24
wol_Latn	Low	1.71	1.39	1.54	3.45	1.08	2.25	1.31	1.90
xho_Latn	High	1.78	1.99	2.76	2.87	1.06	1.97	1.41	2.27
yor_Latn	Low	1.32	1.70	2.08	2.82	1.08	1.58	1.58	3.75
zho_Hans	High	5.72	0.67	4.90	15.66	1.86	5.29	3.05	10.66
zho_Hant	High	5.57	0.64	4.36	15.21	1.88	7.02	3.10	11.18
zsm_Latn	High	5.27	2.22	4.43	21.53	2.88	4.62	4.78	15.45
zul_Latn	High	1.40	1.73	2.46	2.72	0.79	1.43	1.14	2.03
Avg_L		2.11	1.37	2.71	**5.64**	0.99	1.95	2.05	**4.22**
Avg_H		4.95	1.61	7.29	**16.98**	1.49	2.33	2.56	**7.68**

Table 24: **Generating Capability (Part II):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from the other languages into English.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
amh_Ethi	Low	0.22	0.52	0.28	0.26	0.41	0.20	0.44	0.54
arb_Arab	High	2.29	1.44	2.31	2.45	1.33	2.02	5.09	6.10
asm_Beng	Low	0.39	9.90	7.68	6.76	2.31	3.51	2.26	0.72
ast_Latn	Low	5.54	5.10	8.76	1.82	6.09	4.66	7.12	1.94
azj_Latn	Low	5.44	6.17	4.25	6.15	9.03	6.62	1.15	9.84
bel_Cyrl	Low	1.35	1.48	1.31	1.56	1.20	1.41	1.83	1.47
ben_Beng	High	0.88	0.98	0.93	0.75	1.30	0.84	2.98	3.15
bos_Latn	High	4.35	1.72	3.31	8.17	0.88	1.15	2.64	2.00
bul_Cyrl	High	7.17	2.65	5.46	12.28	1.36	1.27	2.11	2.07
cat_Latn	High	9.88	2.77	7.52	12.65	5.90	4.70	4.61	15.89
ceb_Latn	Low	3.55	3.75	2.69	3.47	2.22	1.39	3.69	2.67
ces_Latn	High	5.36	2.68	4.21	9.54	1.26	0.86	2.27	1.70
ckb_Arab	Low	8.78	6.13	5.77	6.64	9.10	3.24	5.05	7.32
cym_Latn	Low	1.92	1.58	1.28	1.39	1.93	1.13	2.20	1.45
dan_Latn	High	8.30	5.20	7.88	20.46	1.41	1.28	2.74	3.39
deu_Latn	High	5.84	4.50	5.58	16.15	2.08	1.57	4.09	4.58
ell_Grek	High	2.51	1.73	1.98	2.65	1.48	1.46	1.57	1.80
est_Latn	High	1.68	2.00	1.23	1.99	1.11	0.31	2.49	1.68
fin_Latn	High	3.88	2.27	2.93	7.00	1.06	1.01	2.06	1.45
fra_Latn	High	7.55	2.44	11.70	26.58	5.78	6.66	16.54	18.91
fuv_Latn	Low	5.85	5.96	3.68	7.74	3.92	5.65	2.02	5.68
gle_Latn	Low	1.98	1.41	1.48	1.67	1.65	0.78	2.38	1.68
glg_Latn	Low	5.28	3.16	3.54	7.41	2.42	2.39	8.09	6.21
guj_Gujr	Low	0.47	0.02	0.57	0.62	2.60	0.57	2.91	2.26
hau_Latn	Low	2.09	1.28	1.55	1.47	1.24	0.87	1.86	1.76
heb_Hebr	High	2.20	1.72	1.51	2.33	1.36	0.64	1.59	1.55
hin_Deva	High	2.47	1.56	2.74	3.70	2.11	1.81	4.04	6.07
hrv_Latn	High	3.61	2.58	4.01	9.36	1.04	0.71	2.67	1.97
hun_Latn	High	5.93	2.00	3.70	8.26	1.32	1.18	1.63	1.77
hye_Armn	Low	1.48	0.95	0.81	1.22	1.27	0.41	0.93	1.47
ibo_Latn	Low	2.03	1.42	1.29	1.23	1.40	0.56	1.87	1.02
ind_Latn	High	10.54	2.72	6.56	12.96	5.45	3.77	8.63	14.36
isl_Latn	High	1.95	1.96	1.74	2.63	1.57	1.14	2.19	1.90
ita_Latn	High	7.99	4.43	7.03	14.13	2.27	1.93	6.57	5.38
jav_Latn	Low	2.28	1.34	1.54	2.23	0.75	0.53	1.96	1.64
jpn_Jpan	High	0.00	0.03	0.03	0.21	0.02	0.05	0.01	0.26
kam_Latn	Low	3.33	1.67	4.50	8.90	9.15	3.33	6.67	7.02
kan_Knda	Low	0.04	0.12	0.49	0.77	1.38	0.49	2.48	2.99
kat_Geor	Low	1.38	1.84	1.20	2.00	1.24	0.62	1.99	1.52
kaz_Cyrl	High	1.63	2.06	1.11	1.07	1.51	1.20	2.36	1.40
khm_Khmr	Low	0.16	0.06	0.27	0.50	0.02	0.57	0.11	0.04
kir_Cyrl	Low	1.54	1.46	0.98	0.85	1.36	0.92	1.99	1.17
kor_Hang	High	3.06	2.34	2.20	2.78	0.94	0.83	1.96	1.32
lao_Laoo	Low	0.00	1.42	0.82	1.21	0.38	0.89	1.62	0.04
lij_Latn	Low	5.70	3.32	4.82	1.95	4.72	3.50	6.06	1.27
lim_Latn	Low	3.40	8.23	2.48	5.81	4.57	6.13	0.33	9.33
lin_Latn	Low	6.67	2.73	9.36	9.64	2.29	0.36	3.36	4.14
lit_Latn	High	1.95	2.33	1.39	2.55	1.30	1.04	2.32	1.62
ltz_Latn	Low	2.09	1.53	1.34	1.36	1.21	0.89	1.78	1.63
lug_Latn	Low	3.94	7.60	1.36	3.66	2.49	7.97	1.43	6.02
Avg_L		3.07	2.69	3.13	**3.27**	2.54	2.14	3.12	**3.02**
Avg_H		4.95	2.38	3.93	**8.09**	2.04	1.74	3.79	**4.71**

Table 25: **Understanding Capability (Part I):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from English into the other languages.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
luo_Latn	Low	0.83	2.48	8.55	2.32	3.56	2.09	3.15	6.90
lvs_Latn	High	1.94	1.69	1.49	2.14	1.17	0.94	2.14	1.61
mal_Mlym	Low	0.28	2.07	1.74	1.32	0.90	0.77	1.86	1.42
mar_Deva	Low	1.25	1.59	1.59	1.57	0.98	0.69	2.12	3.01
mkd_Cyrl	High	2.38	2.01	2.07	5.07	1.15	1.12	2.18	1.83
mlt_Latn	High	1.63	2.48	1.41	3.03	1.76	1.11	2.15	1.80
khk_Cyrl	Low	1.60	1.21	0.79	0.99	1.22	0.96	1.83	1.14
mri_Latn	Low	2.15	1.27	1.59	2.57	1.87	0.98	1.79	2.00
mya_Mymr	Low	5.94	2.67	9.24	0.68	8.26	9.09	4.95	3.47
nld_Latn	High	6.83	7.46	5.46	13.88	1.41	1.00	3.60	3.71
nob_Latn	Low	6.94	3.99	6.77	15.90	1.17	0.71	2.30	2.73
npi_Deva	Low	1.53	0.49	1.06	1.13	0.55	1.04	2.08	1.29
nso_Latn	Low	1.91	5.76	8.40	8.99	9.24	0.88	2.62	2.71
nya_Latn	Low	6.44	2.46	4.86	2.42	0.71	1.71	8.48	1.86
oci_Latn	Low	6.52	0.48	8.50	6.87	4.25	8.06	9.81	7.93
gaz_Latn	Low	8.05	9.87	7.30	1.65	4.16	5.53	7.56	6.30
ory_Orya	Low	0.37	0.02	0.33	0.67	0.92	0.14	1.52	1.47
pan_Guru	Low	8.86	6.16	5.57	8.52	2.92	1.25	7.94	8.46
pes_Arab	High	2.22	2.11	2.00	2.42	0.72	0.60	2.26	1.23
pol_Latn	High	3.86	2.70	4.44	7.60	1.35	0.84	1.80	1.62
por_Latn	High	14.20	3.15	9.29	20.82	10.32	5.81	6.54	18.79
pbt_Arab	Low	1.79	1.06	1.13	1.37	1.40	1.30	1.41	1.60
ron_Latn	High	10.12	5.96	6.47	13.76	1.24	1.46	3.31	2.60
rus_Cyrl	High	6.22	2.39	4.60	13.73	1.73	2.00	2.47	3.35
slk_Latn	High	2.73	1.88	2.94	5.23	1.09	0.73	1.45	1.84
sna_Latn	Low	1.93	1.56	1.53	1.25	1.79	1.21	2.13	1.82
snd_Arab	Low	1.74	1.21	1.10	1.20	1.20	1.02	1.68	1.05
som_Latn	Low	1.56	1.53	1.51	1.20	1.58	0.75	1.86	2.08
spa_Latn	High	9.86	2.37	7.59	15.00	3.22	5.04	4.54	10.77
srp_Cyrl	Low	5.36	1.70	3.89	13.96	1.29	0.84	1.56	1.86
swe_Latn	High	11.83	3.42	7.02	20.70	0.95	0.71	4.46	2.40
swh_Latn	High	1.39	1.54	1.39	1.95	1.31	0.93	3.20	3.23
tam_Taml	Low	1.76	1.71	1.11	1.38	1.31	1.02	2.64	2.66
tel_Telu	Low	0.14	0.51	0.72	0.82	1.54	0.94	2.84	2.44
tgk_Cyrl	Low	1.38	1.43	0.96	1.13	1.43	1.16	1.51	1.95
tgl_Latn	High	6.43	4.28	3.88	5.05	9.30	9.19	7.14	8.34
tha_Thai	High	0.81	0.28	0.97	1.47	0.56	0.20	0.07	0.66
tur_Latn	High	2.00	2.43	1.79	3.12	1.14	0.62	2.68	1.75
ukr_Cyrl	High	5.63	2.12	4.87	11.76	1.37	0.92	2.26	2.39
umb_Latn	Low	4.40	4.92	1.14	1.54	2.62	8.02	8.62	5.69
urd_Arab	Low	1.58	0.75	1.25	1.63	1.26	1.13	1.90	3.68
uzn_Latn	High	1.17	1.35	1.04	0.99	1.00	0.83	1.89	1.39
vie_Latn	High	10.90	1.72	7.86	11.97	3.79	1.35	9.22	15.66
wol_Latn	Low	9.46	3.83	9.78	2.14	1.34	1.87	5.58	1.15
xho_Latn	High	1.77	1.42	1.43	1.16	1.57	0.54	1.98	1.19
yor_Latn	Low	1.83	1.79	1.17	1.83	0.75	0.53	1.81	0.80
zho_Hans	High	11.90	0.56	8.36	14.29	1.94	3.39	16.12	15.27
zho_Hant	High	6.94	0.75	6.30	14.95	1.26	2.67	8.12	11.62
zsm_Latn	High	6.81	1.70	3.63	8.30	1.56	1.63	3.72	7.10
zul_Latn	High	1.83	1.78	1.22	1.03	0.99	0.66	1.87	1.12
Avg_L		3.07	2.69	3.13	**3.27**	2.54	2.14	3.12	**3.02**
Avg_H		4.95	2.38	3.93	**8.09**	2.04	1.74	3.79	**4.71**

Table 26: **Understanding Capability (Part II):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from English into the other languages.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
amh_Ethi	Low	0.15	0.14	1.22	1.60	1.84	0.12	0.24	1.33
arb_Arab	High	2.11	2.85	6.54	6.83	3.18	3.00	4.05	16.75
asm_Beng	Low	3.26	5.08	6.03	1.76	2.96	2.75	0.35	0.63
ast_Latn	Low	2.90	6.78	1.15	2.12	4.98	4.19	3.48	2.60
azj_Latn	Low	1.99	8.30	1.78	6.29	1.19	0.61	5.60	0.84
bel_Cyrl	Low	1.78	1.71	2.00	2.99	1.51	1.34	1.39	1.63
ben_Beng	High	0.16	1.52	2.43	1.59	3.56	0.76	1.53	12.10
bos_Latn	High	7.16	9.19	9.34	24.57	1.78	1.26	1.83	4.41
bul_Cyrl	High	9.33	6.57	7.64	16.32	2.14	1.04	1.55	3.57
cat_Latn	High	16.82	11.22	9.18	34.27	3.75	2.95	4.72	23.42
ceb_Latn	Low	2.50	2.63	4.46	3.84	2.28	1.66	2.36	1.90
ces_Latn	High	13.80	8.68	11.32	26.37	2.21	0.47	2.00	2.50
ckb_Arab	Low	2.32	4.63	8.83	5.61	1.09	9.36	5.34	6.21
cym_Latn	Low	1.69	1.39	2.90	1.93	1.72	1.23	2.24	1.52
dan_Latn	High	22.34	11.42	13.45	33.71	3.80	1.65	3.05	2.31
deu_Latn	High	11.38	6.73	8.54	25.71	5.19	1.61	2.78	8.61
ell_Grek	High	1.49	2.77	8.65	6.23	1.55	1.30	1.61	1.33
est_Latn	High	3.66	1.80	3.57	3.20	1.34	0.76	1.86	2.07
fin_Latn	High	3.84	6.42	8.97	20.80	1.23	0.96	1.71	2.10
fra_Latn	High	18.39	9.76	6.37	34.26	3.63	3.30	2.95	22.36
fuv_Latn	Low	2.60	7.70	3.28	9.27	1.41	4.95	4.76	9.46
gle_Latn	Low	1.92	1.43	2.45	2.16	1.62	1.03	2.31	1.85
glg_Latn	Low	11.38	8.42	9.92	19.33	4.61	1.30	11.41	3.73
guj_Gujr	Low	0.15	0.28	0.32	0.17	2.39	0.89	1.23	5.54
hau_Latn	Low	1.68	1.37	1.82	1.78	1.40	1.08	1.82	1.90
heb_Hebr	High	2.36	2.18	4.23	4.25	1.08	1.06	1.84	2.43
hin_Deva	High	1.49	1.87	1.55	3.97	4.21	0.93	2.28	11.41
hrv_Latn	High	10.02	7.71	9.49	22.01	2.09	1.21	1.77	3.78
hun_Latn	High	7.30	5.86	6.33	18.96	1.50	1.30	1.72	1.76
hye_Armn	Low	1.27	1.28	1.95	1.45	1.15	0.12	1.46	1.30
ibo_Latn	Low	1.72	1.81	1.97	1.68	1.65	0.45	1.06	1.94
ind_Latn	High	6.83	8.19	10.76	19.30	4.86	1.47	4.70	17.02
isl_Latn	High	1.90	1.90	3.93	3.80	1.65	1.33	2.13	2.03
ita_Latn	High	8.37	8.20	9.29	24.23	5.27	1.82	3.13	13.22
jav_Latn	Low	1.84	1.71	2.35	2.06	2.36	0.55	2.97	2.12
jpn_Jpan	High	0.56	3.38	7.98	11.06	2.31	0.29	1.22	2.20
kam_Latn	Low	3.01	5.38	5.69	5.09	1.49	5.48	0.03	7.99
kan_Knda	Low	0.05	0.17	0.01	1.01	2.86	0.42	2.02	4.46
kat_Geor	Low	1.27	1.34	1.97	1.50	1.71	1.29	0.81	1.36
kaz_Cyrl	High	1.79	1.52	1.95	1.92	1.30	1.74	1.82	1.79
khm_Khmr	Low	0.05	0.11	0.07	1.08	1.58	0.42	0.45	0.12
kir_Cyrl	Low	1.58	1.33	1.62	1.62	1.29	1.48	1.68	1.85
kor_Hang	High	3.41	4.14	5.57	9.18	1.66	1.51	1.72	2.59
lao_Laoo	Low	0.08	0.24	0.01	0.83	0.87	0.40	0.12	0.33
lij_Latn	Low	2.27	7.30	3.39	4.54	3.41	8.48	9.36	6.96
lim_Latn	Low	9.58	8.27	6.28	3.48	2.81	3.49	3.36	0.98
lin_Latn	Low	4.24	6.54	5.64	3.51	1.40	4.97	1.95	9.21
lit_Latn	High	2.39	1.69	2.72	2.81	1.48	1.40	1.82	2.73
ltz_Latn	Low	2.35	2.08	1.86	4.34	2.26	1.19	2.20	1.82
lug_Latn	Low	4.78	1.10	2.35	7.36	1.37	6.28	0.45	9.75
Avg_L		2.96	3.15	3.16	**4.04**	2.05	2.41	2.62	**3.94**
Avg_H		6.61	5.31	6.92	**14.18**	2.56	1.57	2.52	**6.54**

Table 27: **Understanding Capability (Part I):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from the other languages into English.

#Langs.	Type	LLaMA	BX_{LLaMA}	SFT_{LLaMA}	xLLMs-100	BLOOM	BX_{BLOOM}	SFT_{BLOOM}	xLLMs-100
luo_Latn	Low	3.14	0.78	1.96	3.40	1.32	1.94	8.60	6.47
lvs_Latn	High	1.65	1.62	2.44	2.37	1.34	1.13	1.74	1.98
mal_Mlym	Low	1.18	0.92	0.57	1.46	2.94	0.92	2.98	4.08
mar_Deva	Low	2.06	1.75	2.60	1.66	2.64	1.38	1.78	7.09
mkd_Cyrl	High	11.35	5.74	8.93	12.52	2.24	1.21	1.46	4.67
mlt_Latn	High	1.61	2.67	2.56	3.96	2.05	1.65	2.72	2.10
khk_Cyrl	Low	1.46	1.00	1.60	1.46	1.31	1.40	1.49	1.62
mri_Latn	Low	1.73	1.57	1.61	1.48	1.46	0.63	1.37	2.07
mya_Mymr	Low	9.92	9.17	1.14	5.15	0.89	8.24	4.79	3.87
nld_Latn	High	5.75	5.66	7.11	20.92	3.70	1.20	2.44	2.21
nob_Latn	Low	15.08	10.52	13.75	29.80	3.41	1.39	2.91	2.04
npi_Deva	Low	1.17	1.68	3.10	1.99	2.80	1.51	1.87	9.79
nso_Latn	Low	0.86	5.12	0.79	9.60	1.80	5.09	7.96	2.81
nya_Latn	Low	5.10	9.09	5.92	6.82	1.70	6.66	0.46	4.39
oci_Latn	Low	8.08	9.11	2.90	8.65	5.93	3.68	5.41	8.92
gaz_Latn	Low	8.80	0.88	8.54	2.63	1.09	0.85	3.00	8.07
ory_Orya	Low	0.28	0.10	0.17	1.21	2.43	0.88	1.77	4.63
pan_Guru	Low	0.53	3.45	8.82	0.06	1.99	7.65	0.19	8.78
pes_Arab	High	2.74	2.29	3.24	4.68	1.53	0.72	1.94	6.91
pol_Latn	High	13.06	6.08	5.31	18.40	2.02	0.62	1.70	3.40
por_Latn	High	18.15	11.58	6.45	34.86	4.29	1.67	5.36	27.78
pbt_Arab	Low	1.08	1.01	1.75	0.79	0.58	0.93	0.88	1.27
ron_Latn	High	10.32	10.31	11.65	27.41	3.72	1.47	2.34	6.69
rus_Cyrl	High	15.52	8.02	9.76	21.81	2.43	1.69	2.84	5.77
slk_Latn	High	2.71	6.72	12.06	14.19	2.22	1.20	1.71	1.84
sna_Latn	Low	1.84	2.19	2.09	2.01	1.64	1.36	2.50	2.31
snd_Arab	Low	1.01	1.22	1.83	0.85	1.03	1.06	0.69	1.51
som_Latn	Low	1.76	2.06	1.95	1.80	1.33	1.06	1.41	1.76
spa_Latn	High	6.23	8.95	9.52	24.27	2.18	3.06	2.56	4.41
srp_Cyrl	Low	8.20	6.17	11.39	16.59	1.65	1.38	1.62	1.72
swe_Latn	High	25.77	10.82	14.26	31.00	3.05	1.31	2.41	3.15
swh_Latn	High	1.96	1.46	2.45	2.08	2.65	0.74	4.83	6.83
tam_Taml	Low	1.16	0.69	1.84	1.04	2.94	0.37	2.46	4.82
tel_Telu	Low	0.06	0.18	0.18	1.25	3.28	1.17	1.81	4.76
tgk_Cyrl	Low	1.55	1.20	1.43	1.41	1.31	1.64	1.71	1.09
tgl_Latn	High	4.09	2.06	2.60	7.84	2.18	8.23	0.10	9.50
tha_Thai	High	0.49	1.60	4.37	2.10	1.61	0.63	0.74	1.85
tur_Latn	High	2.33	2.42	3.73	8.16	1.31	1.20	2.07	2.65
ukr_Cyrl	High	8.98	6.26	9.05	19.68	1.91	1.23	1.74	4.89
umb_Latn	Low	8.90	3.17	1.19	4.20	1.34	4.78	5.42	2.34
urd_Arab	Low	1.24	1.47	2.51	1.89	3.73	0.35	3.13	11.51
uzn_Latn	High	1.43	1.34	1.56	1.22	1.27	1.23	1.60	1.58
vie_Latn	High	1.55	5.62	8.20	10.25	4.51	2.57	6.24	5.99
wol_Latn	Low	0.80	2.45	4.91	6.69	1.31	4.74	0.98	9.66
xho_Latn	High	1.50	1.88	1.87	2.15	1.41	0.53	1.70	1.74
yor_Latn	Low	1.46	1.39	1.63	1.66	1.58	1.01	1.12	1.87
zho_Hans	High	5.66	6.79	13.28	13.33	3.05	2.57	5.16	12.45
zho_Hant	High	4.44	6.82	13.47	15.41	3.10	2.50	5.39	11.68
zsm_Latn	High	5.21	5.52	9.73	10.71	4.78	1.42	4.70	13.55
zul_Latn	High	1.50	1.66	1.78	2.00	1.14	0.70	1.03	1.45
Avg_L		2.96	3.15	3.16	**4.04**	2.05	2.41	2.62	**3.94**
Avg_H		6.61	5.31	6.92	**14.18**	2.56	1.57	2.52	**6.54**

Table 28: **Understanding Capability (Part II):** BLEU scores on the FLORES benchmark (Goyal et al., 2022) and average scores on low-resource (Avg_L) and high-resource (Avg_H) languages. We report on translating from the other languages into English.