

How Many Languages Make Good Multilingual Instruction Tuning? A Case Study on BLOOM

Shaoxiong Ji*[†]

Technical University of Darmstadt
University of Helsinki
shaoxiong.ji@tu-darmstadt.de

Pinzhen Chen*

University of Edinburgh
pinzhen.chen@ed.ac.uk

Abstract

Instruction tuning a large language model with multiple languages can prepare it for multilingual downstream tasks. Nonetheless, it is yet to be determined whether having a handful of languages is sufficient, or whether the benefits increase with the inclusion of more. By fine-tuning large multilingual models on 1 to 52 languages, we present a case study on BLOOM to understand three pertinent factors affecting performance: the number of languages, language exposure, and similarity between training and test languages. Overall we found that 1) expanding language coverage in multilingual instruction tuning proves to be beneficial; 2) accuracy often significantly boots if the test language appears in the instruction mixture; 3) languages’ genetic features correlate with cross-lingual transfer more than merely the number of language but different languages benefit to various degrees.

1 Introduction

Many large language models (LLMs) have been designed to handle many languages through multilingual pre-training, like mGPT (Shliazhko et al., 2022) and BLOOM (Scao et al., 2022), while some other models only (officially) support a few, e.g. the Llama series (Touvron et al., 2023; Grattafiori et al., 2024). At the alignment stage, as a more affordable route, researchers used multilingual instruction tuning (mIT) to enhance the multilingualism of LLM (Muennighoff et al., 2023). Recently, Chen et al. (2024) compared monolingual and multilingual instruction tuning under a resource-fair scenario with multiple LLMs. Kew et al. (2024) experimented with English-centric LLMs such as Llama 2 (Touvron et al., 2023) and Falcon (Almazrouei et al., 2023) and found that mIT using as few as three languages enables cross-lingual transfer. Similarly,

Shaham et al. (2024) studied the same topic featuring “just a pinch of multilinguality” of 2–4 languages. Moreover, Chirkova and Nikoulina (2024) showed that instruction tuning in only English with a carefully set learning rate enables responses in four other test languages. We note the variations in the choice of languages, base models, and testbeds used in these studies. More importantly, while it has been demonstrated that a handful of languages elicit (zero-shot) multilingual responses, it does not imply the optimal downstream task results, not to mention to cater to each language.

To fill the gap, we perform instruction tuning on the multilingual BLOOM model (Scao et al., 2022) on a parallel instruction dataset named Bactrain-X in 52 languages (Li et al., 2023). We progressively add a language for each mIT run, resulting in 52 models in total,¹ which are then evaluated on three multilingual benchmarks. Patterns on BLOOM reveal that *contrary to prior research, having more languages beyond a handful can further improve performance, although with diminishing returns and some outlier cases*. Our findings are summarized as follows:

1. Cross-lingual transfer improves with more languages in mIT, but the optimal number of languages depends on the task and test language, with varying behaviours across benchmarks and languages.
2. Including a specific language in the instruction tuning data generally enhances its performance, though outliers exist, and the benefits from massive mIT are limited if a language is not part of the tuning data, regardless of its presence during pre-training.

¹huggingface.co/collections/MaLA-LM → Lucky52. Fun fact: Lucky 52 was a famous variety show in China in the early 2000s. The naming denotes the 52 models fine-tuned on 1 to 52 languages.

*Equal contribution.

[†]Work done while at the University of Helsinki.

- Correlations between language similarity and performance vary, with genetic features being more predictive than the number of languages. Some languages, like Thai and Swahili, show strong inter-language effects, while others, like English and Chinese, have weaker correlations.

Our study emphasizes the importance of a closer look at the tasks, benchmarks, languages, and evaluation metrics. We advocate for more consistent future studies focused on mIT. Further variables include but are not limited to base LLMs, pre-training recipes and data. Comprehensive and consistent investigations are crucial for advancing our understanding of mIT and its implications.

2 Scaling Instruction Languages

2.1 Increasing the Number of Languages

Our setup is supervised fine-tuning, where an instruction and a task input are fed to an LLM to yield a response. We progressively include an extra language in each training run to study the precise effect the number of languages brings in. As the number of languages expands, instruction data size also grows—to mitigate this variable, we opt for parallel instruction data in which English instructions are translated into other languages. This controls that all models are trained with a comparable amount of instruction information. Moreover, the increase in data size also increases the number of optimization steps when utilizing stochastic gradient descent to update the model parameters on the same device. We express the number of updates as $U = \lceil \frac{N \times L \times E}{B \times W} \rceil$, where N is the instruction data size, L is the number of languages, E is the number of epochs, B is the batch size, and W is the number of GPUs. We increase W proportionally to L to maintain a manageable range of updates.

2.2 Multilingual Instruction Data

We use the Bactrian-X dataset (Li et al., 2023) comprising 3.4 million instruction-response pairs with an equal share in each of its 52 languages. We fine-tune an LLM from 1 to 52 languages resulting in 52 models. The languages are added in a specific order: en (English), zh (Chinese), and the rest in alphabetical order. Refer to Appendix A for an exhaustive list of languages and their 2-digit codes.

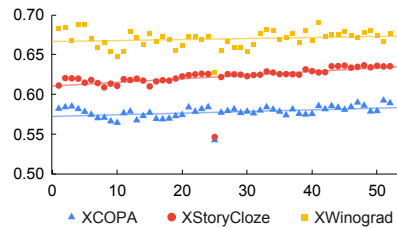


Figure 1: Test performance across all languages; x-axis: number of languages in mIT; y-axis: average accuracy.

2.3 Base Language Model Tuning

Multilingual language models can inherit distribution biases present in the training data, which may affect their capability across languages after instruction tuning. We hence base our experiments on BLOOM (Scao et al., 2022) which has been developed with careful consideration in multiple natural and coding languages. Its massive multilingual tokenization support makes it well-suited for studying a large number of languages in a fair manner. In our specific application, we opt for the BLOOM-7B1 variant with 7.1B parameters.

Training Details We use the transformers framework (Wolf et al., 2019) with DeepSpeed integration (Rasley et al., 2020) for fine-tuning. We set the learning rate to $3e-5$ and the batch size to 4 per device. Gradient accumulation, with a step size of 4, enables the aggregation of gradients over multiple steps. The number of epochs is fixed at 3. The maximum model length is set to 768, the same as in Bactrian-X (Li et al., 2023). Models are trained on a cluster with 4 AMD MI250X GPUs (in total 8 GPU dies) in each node. We adopt distributed training on multiple nodes ranging from 2 to 10 with the increase in the number of languages, making the global batch size range from 256 to 1280.

2.4 Benchmarks and Evaluation

We test on three multilingual benchmarks. XCOPA (Ponti et al., 2020) is a multilingual dataset for causal commonsense reasoning in 11 languages. XStoryCloze (Lin et al., 2021) is a multilingual dataset for commonsense reasoning in the story in 10 non-English languages. XWinograd (Tikhonov and Ryabinin, 2021) is a multilingual compilation of Winograd Schemas (Levesque et al., 2012) available in 6 languages. We run zero-shot prompting via lm-evaluation-harness (Gao et al., 2023). Different models, trained with progressively added languages, are evaluated on these benchmarks using accuracy (0-1) as the metric.

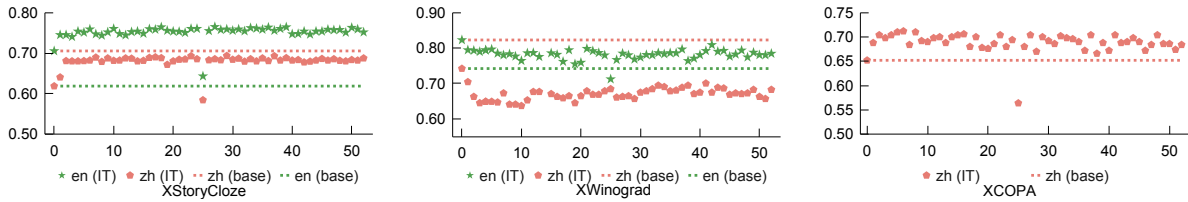


Figure 2: Accuracy for English and Chinese on XStoryCloze, XWinograd, and XCOPA.

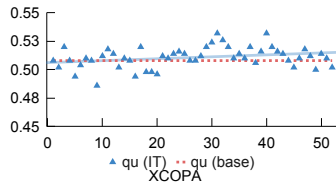


Figure 3: Accuracy for Quechuan, unseen both by the base model and during IT.

3 Results and Discussions

3.1 Number of Languages

Overall pattern We first study the effect of the number of languages on multilingual performance—how much multilingualism do we want for instruction tuning an LLM, i.e., BLOOM-7B1 in our case study? Figure 1 illustrates the average accuracy on the three benchmarks with different numbers of languages in the instruction data. For XCOPA and XStoryCloze, there is a positive correlation between the number of instruction languages and performance; but for XWinograd, we observe fluctuating results with a weaker trend. A notable drop appears in the scatter plot across all benchmarks—the point where Korean (kr) is added to the IT data. We inspected the training curve of the model trained with Korean added to instruction data and found that the training and validation loss decreases as training goes on and the model converges as expected.

English versus Chinese We move on to two specific languages, i.e. English and Chinese, as displayed in Figure 2 together with base model prompting performance. We notice a similar drop in accuracy when Korean is added, but there is no obvious benefit from cross-lingual transfer when more languages are added. For English and Chinese XStoryCloze, the highest accuracy is attained much later when the 27th (Latvian, lv) or 29th (Malaysian, ml) language is added, respectively. Yet, interestingly, while instruction tuning surpasses the base model for English, it makes it worse for Chinese. XWinograd exhibits a similar trend that instruction tuning benefits English but drastically harms Chinese. In addition, the best IT perfor-

mance for both languages is observed when there is only one language (English) in the instruction data. Specifically, the result for Chinese XCOPA peaks early when the 6th language (Bangla, bn) is added, but later models with more languages no longer improve.

Summary Instruction tuning with a few languages is useful for cross-lingual transfer, but having more languages can further improve the average results when many languages are of concern. However, distinct behaviours can be witnessed for different benchmarks and individual languages, so the optimal number of languages in mIT depends on the task and test language.

3.2 Language Exposure

A test language can fall into one of the cases depending on being seen or unseen during the pre-training and instruction tuning phases: (1) unseen by the base and during IT: qu (XCOPA). (2) seen by the base but unseen during IT: ht (XCOPA) and eu (XStoryCloze). (3) unseen by the base but seen during IT: et, it, th, tr (XCOPA); my, ru (XStoryCloze); ja, ru (XWinograd). (4) seen by the base and during IT, e.g.: id, sw, ta, vi, zh (XCOPA); ar, en, es, hi, id, sw, te, zh (XStoryCloze); en, fr, pt, zh (Winograd). We are interested in understanding model performance in the first three categories where the number and closeness of mIT languages may benefit or harm unseen languages.

Unseen by base, unseen during IT Only one language is not covered by either pre-training or instruction tuning: Quechuan (qu) in XCOPA. We plot its performance across all data mixtures in Figure 3. As the number of IT languages grows, accuracy fluctuates around the base model performance, showing a weak trend. This implies that if a language has no presence at all, there is very little transfer mIT can do.

Unseen by base, seen during IT We then investigate an important use of multilingual instruction tuning—to adapt the base LLM to unseen lan-

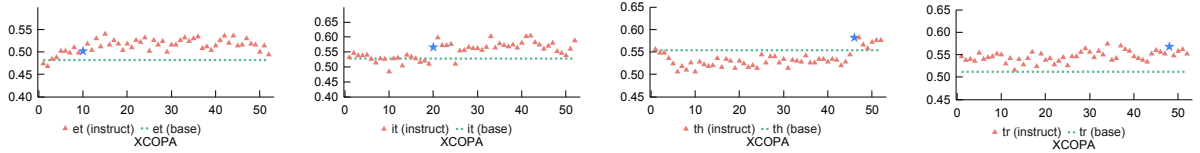


Figure 4: Accuracy on XCOPA for various languages, unseen by the base model but seen during IT. ★ indicates the point the test language starts to be included in the mIT data. In most cases, performance can benefit (et, it, th) from the test language appearing in mIT despite outliers (tr).

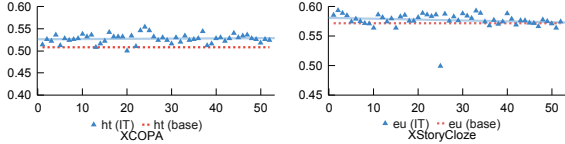


Figure 5: Accuracy for Haitian on XCOPA and Basque on XStoryCloze, seen by base unseen during IT.

languages during pre-training. Figure 4 show the accuracy of various languages that have not been (intentionally) learned during pre-training but appeared in instruction tuning (at some point), with additional languages exhibiting similar trends in Appendix B Figure 6. We find that in the majority of scenarios, including a language in IT can immediately aid the performance of that language as anticipated. However, we also notice two interesting cases: 1) for Turkish (tr) tested in XCOPA (Figure 4), the accuracy is similar before and after introducing the language to the IT data; 2) for Russian (ru) tested in XWinograd (Figure 6b), the performance is below base model prompting even after the language appears in mIT. Besides, cross-lingual transfer is observed in mIT, for instance: the performance of Estonia (et) and Italian (it) tested in XCOPA can further grow after more languages are added; the performance of Turkish (tr) tested in XCOPA is already favourable without the language itself.

Seen by base, unseen during IT Finally, Figure 5 displays the two languages in this category. In both cases, IT is better than base model prompting, indicating that mIT can transfer to unseen IT languages that have been pre-trained. Nonetheless, increasing the number of languages during mIT does not significantly bring benefits.

Summary By examining result patterns in exhaustive seen-unseen cases, we can infer that having a particular language in instruction tuning data is often beneficial for its performance, although some outliers can be observed. Regardless of whether a language is learned during pre-training,

if it does not appear in the IT data composite, the benefit from massive mIT is usually limited.

3.3 Language Similarity

We conduct a post-hoc analysis on how language closeness affects cross-lingual transfer. Instead of studying the relation between the number of fine-tuning languages and test set performance, we define an aggregated similarity measure between all languages present in a fine-tuning corpus and a test language L_{test} :

$$\text{similarity}_{\text{train, test}} = \sum_{L \in \text{corpus}} \text{sim}(L, L_{\text{test}})$$

where $\text{sim}(\cdot, \cdot)$ is a similarity metric between two languages. We measure “aggregated similarity” instead of “average similarity” because we argue that, given their giant sizes, LLMs have the capacity to model all language data in the training set simultaneously.

We adopt different similarity measures based on syntactic, geographic, phonological, genetic, inventory, and featural distances scored by `lang2vec` (Littell et al., 2017; Malaviya et al., 2017).² In addition, we gathered from another source a language closeness score derived from sound (consonants) overlap, which is deemed to reflect genetic similarity (Beaufils and Tomin, 2020).³ In total, we test out seven measures, where the similarity score is always normalized to between 0 and 1 to the lowest and highest similarity. The choice of language features is similar to a contemporaneous study on language transferability and similarity (Philippy et al., 2024). As a baseline comparison, we provide Pearson correlation coefficients between the number of languages and performance: XStoryClose in Table 1, XWingrad in Table 2, and XCOPA in Table 3. Also, since empirically the addition of Korean leads to an outlying performance, we compute coefficients without the particular checkpoint too.

For XCOPA and XStoryCloze, `lang2vec` genetic features stand out, usually resulting in a stronger correlation than simply the number of lan-

²github.com/antonisa/lang2vec

³linguistics.net/language_evolution.html

	ar	en	es	hi	id	my	ru	sw	te	zh
num. lang.	-0.07	0.15	0.46	0.51	0.53	0.75	0.81	0.56	-0.47	0.11
num. lang. w/o ko	0.08	0.41	0.73	0.66	0.63	0.75	0.86	0.56	-0.53	0.31
sound correspondence	-0.06	0.15	0.48	0.52	0.57	0.82	0.83	0.67	-0.43	0.12
lang2vec featural	-0.05	0.15	0.47	0.51	0.53	0.77	0.83	0.58	-0.46	0.13
lang2vec genetic	0.17	0.16	0.50	0.54	0.66	0.96	0.87	0.96	-0.26	0.37
lang2vec geographic	0.17	0.15	0.47	0.51	0.54	0.76	0.81	0.96	-0.48	0.37
lang2vec inventory	-0.06	0.15	0.46	0.51	0.54	0.76	0.83	0.55	-0.46	0.13
lang2vec phonological	-0.05	0.15	0.47	0.51	0.53	0.76	0.83	0.57	-0.45	0.13
lang2vec syntactic	-0.05	0.15	0.47	0.51	0.53	0.78	0.82	0.57	-0.45	0.13

Table 1: Pearson correlation between XStoryCloze performance and mIT data similarity

	en	fr	ja	pt	ru	zh
num. lang.	-0.02	0.01	0.62	-0.32	-0.07	0.49
num. lang. w/o ko	-0.03	0.00	0.66	-0.35	-0.07	0.50
sound correspondence	-0.01	-0.01	0.66	-0.33	-0.06	0.45
lang2vec featural	-0.01	0.00	0.62	-0.31	-0.06	0.47
lang2vec genetic	-0.02	-0.08	0.72	-0.35	-0.05	-0.31
lang2vec geographic	-0.02	-0.01	0.62	-0.33	-0.07	-0.31
lang2vec inventory	-0.01	0.00	0.62	-0.31	-0.06	0.48
lang2vec phonological	-0.01	0.01	0.63	-0.32	-0.06	0.48
lang2vec syntactic	-0.02	0.00	0.62	-0.32	-0.06	0.47

Table 2: Pearson correlation between XWinograd performance and mIT data similarity

	et	id	it	sw	ta	th	tr	vi	zh
num. lang.	0.44	0.44	0.63	0.54	-0.80	0.53	0.45	-0.46	-0.20
num. lang. w/o ko	0.44	0.50	0.64	0.54	-0.80	0.53	0.46	-0.50	-0.39
sound correspond.	0.51	0.48	0.64	0.64	-0.83	0.62	0.45	-0.36	-0.20
l2v featural	0.46	0.45	0.63	0.56	-0.81	0.55	0.45	-0.44	-0.19
l2v genetic	0.67	0.58	0.67	0.93	-0.84	0.82	0.47	0.02	0.01
l2v geographic	0.43	0.46	0.64	0.93	-0.80	0.55	0.45	-0.45	0.01
l2v inventory	0.46	0.45	0.64	0.52	-0.80	0.55	0.45	-0.45	-0.19
l2v phonological	0.45	0.45	0.62	0.54	-0.80	0.55	0.44	-0.45	-0.19
l2v syntactic	0.45	0.45	0.63	0.54	-0.81	0.54	0.45	-0.45	-0.19

Table 3: Pearson correlation between XCOPA performance and mIT data similarity

guages. While most languages display a positive correlation with mIT language similarity or coverage, we notice that some are negatively affected: ta, and vi in XCOPA, te in XStoryCloze, and pt in XWinograd. Finally, across different test sets, behaviours could be diverging for the same language: genetic similarity benefits ru in XStoryCloze but has no correlation in XWinograd.

Summary Many factors contribute to the train-test similarity and performance correlation in both positive and negative ways: languages, test sets, and similarity measures.

4 Conclusion

While instruction tuning of large multilingual models enables versatile language processing, it requires careful handling of language-specific nu-

ances. This paper presents an experimental analysis that controls the base model, instructions, and training recipe to study the number, closeness, and exposure of languages. Our findings, compared with prior work, show that multilingual instruction tuning depends heavily on factors like base models, data, tasks, and evaluation protocols. We emphasize the need for more systematic studies to validate the effectiveness and generalizability of this approach.

Limitations

Our work studies multilingual instruction tuning in 52 relatively high-resourced languages, which might be limited in size to arrive at comprehensive conclusions for thousands of living languages, which are often under-served. We did not conduct a human evaluation due to budget constraints. Future work could conduct a more systematic assessment with more rigorously controlled variables and heavier regularization during instruction tuning to prevent base model knowledge and language forgetting.

Acknowledgments

This work has received funding from the European Union’s Horizon Europe research and innovation programme under grant agreement No 101070350 and from UK Research and Innovation (UKRI) un-

der the UK government’s Horizon Europe funding guarantee [grant number 10052546].

We acknowledge CSC-IT Center for Science, Finland for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland) and the LUMI consortium through Finnish extreme scale call (project LumiNMT) and Czech Republic allocations issued by e-INFRA CZ, and IT4Innovations National Supercomputing Center.

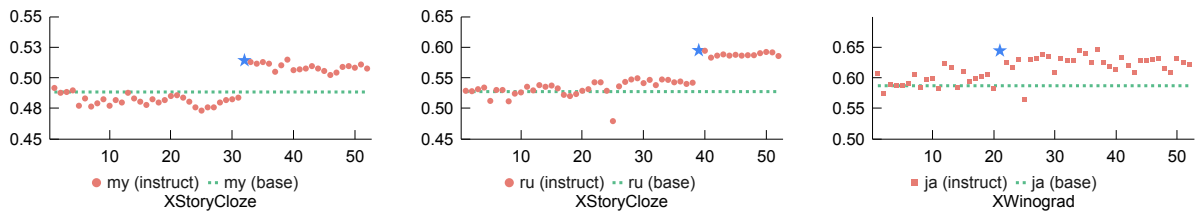
References

- Ebtesam Almazrouei et al. 2023. The Falcon series of open language models. *arXiv preprint*.
- Vincent Beaufils and Johannes Tomin. 2020. Stochastic approach to worldwide language classification: the signals and the noise towards long-range exploration. *SocArXiv*.
- Pinzhen Chen, Shaoxiong Ji, Nikolay Bogoychev, Andrey Kutuzov, Barry Haddow, and Kenneth Heafield. 2024. Monolingual or multilingual instruction tuning: Which makes a better alpaca. In *Findings of the Association for Computational Linguistics: EACL 2024*.
- Nadezhda Chirkova and Vassilina Nikoulina. 2024. Zero-shot cross-lingual transfer in instruction tuning of large language model. *arXiv preprint*.
- Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, et al. 2023. A framework for few-shot language model evaluation. Zenodo.
- Aaron Grattafiori et al. 2024. The Llama 3 herd of models. *arXiv preprint*.
- Tannon Kew, Florian Schottmann, and Rico Sennrich. 2024. Turning English-centric LLMs into polyglots: How much multilinguality is needed? In *Findings of the Association for Computational Linguistics: EMNLP 2024*.
- Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd schema challenge. In *Thirteenth international conference on the principles of knowledge representation and reasoning*.
- 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*.
- Xi Victoria Lin et al. 2021. Few-shot learning with multilingual language models. *arXiv preprint*.
- Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*.
- Chaitanya Malaviya, Graham Neubig, and Patrick Littell. 2017. Learning language representations for typology prediction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.
- Niklas Muennighoff et al. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.
- Fred Philippy, Siwen Guo, Shohreh Haddadan, Cedric Lothritz, Jacques Klein, and Tegawendé F. Bissyandé. 2024. Soft prompt tuning for cross-lingual transfer: When less is more. In *Proceedings of the 1st Workshop on Modular and Open Multilingual NLP (MOOMIN 2024)*.
- 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)*.
- Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.
- Teven Le Scao et al. 2022. BLOOM: A 176B-parameter open-access multilingual language model. *arXiv preprint*.
- Uri Shaham, Jonathan Herzig, Roei Aharoni, Idan Szpektor, Reut Tsarfaty, and Matan Eyal. 2024. Multilingual instruction tuning with just a pinch of multilinguality. *arXiv preprint*.
- Oleh Shliakhko, Alena Fenogenova, Maria Tikhonova, Vladislav Mikhailov, Anastasia Kozlova, and Tatiana Shavrina. 2022. mGPT: Few-shot learners go multilingual. *arXiv preprint*.
- Alexey Tikhonov and Max Ryabinin. 2021. It’s All in the Heads: Using Attention Heads as a Baseline for Cross-Lingual Transfer in Commonsense Reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.
- Hugo Touvron et al. 2023. LLaMA: Open and efficient foundation language models. *arXiv preprint*.
- Thomas Wolf et al. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *arXiv preprint*.

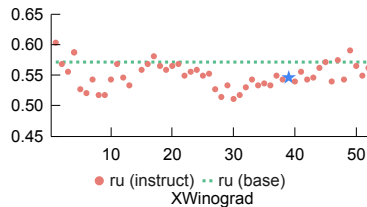
A All languages

Apart from English and Chinese, data in the other 50 languages in Bactrian-X are added in alphabetical order: af (Afrikaans), ar (Arabic), az (Azerbaijani), bn (Bengali), cs (Czech), de (German), es (Spanish), et (Estonian), fa (Farsi), fi (Finnish), fr (French), gl (Galician), gu (Gujarati), he (Hebrew), hi (Hindi), hr (Croatian), id (Indonesian), it (Italian), ja (Japanese), ka (Georgian), kk (Kazakh), km (Khmer), ko (Korean), lt (Lithuanian), lv (Latvian), mk (Macedonian), ml (Malayalam), mn (Mongolian), mr (Marathi), my (Burmese), ne (Nepali), nl (Dutch), pl (Polish), ps (Pashto), pt (Portuguese), ro (Romanian), ru (Russian), si (Sinhala), sl (Slovenian), sv (Swedish), sw (Swahili), ta (Tamil), te (Telugu), th (Thai), tl (Tagalog), tr (Turkish), uk (Ukrainian), ur (Urdu), vi (Vietnamese), and xh (Xhosa).

B Additional plots for languages unseen by base model but seen during IT



(a) Performance on XStory Malaysian and Russian as well as XWinograd Japanese benefits from the test language changing from unseen to seen in mIT.



(b) Performance on XWinograd Russian does not benefit from the test language changing from unseen to seen in mIT.

Figure 6: Accuracy for various languages unseen by the base model but seen during IT. ★ indicates the point when the test language starts to be included in the mIT data.