Outgrowing the Curse of Multilinguality: Scaling Strategies for Efficient Language Adaptation

Kevin Glocker* **Kätriin Kukk*** **Romina Oji**
Marcel Bollmann **Marco Kuhlmann** **Jenny Kunz**
Department of Computer and Information Science
Linköping University
firstname.lastname@liu.se

Abstract

Training high-performing language models which include medium- and lower-resource languages remains a challenge. Massively multilingual models still underperform language-specific adaptations, especially at smaller scales, due to the *curse of multilinguality*. In this work, we investigate *scaling* as an efficient strategy for adapting pretrained models to new target languages. Through comprehensive ablations matched for data and training FLOPs, we test whether upscaling from English base models enables more effective and resource-efficient adaptation than standard continued pretraining. We find that upscaling is an efficient strategy for improving target-language performance by relieving capacity constraints, especially for higher-resourced languages. An initial upscaling step in English is beneficial for low-resource languages. However, in higher-resource scenarios it is more compute-efficient to directly scale up on the target language, albeit slightly compromising data efficiency for most languages. For multilingual continued pre-training, we identify directly scaling up using the target data as the optimal setup for both data efficiency and reduced English forgetting.

1 Introduction

Massively multilingual language models (Mesnard et al., 2024; Ji et al., 2025) are widely used in real-world applications, yet their performance remains uneven across languages. Especially at smaller scales, they still face the *curse of multilinguality* (Conneau et al., 2020). The need to share limited capacity across many languages often results in low performance, particularly for low-resource languages. Indeed, multilingual models can be outperformed by much smaller monolingual counterparts (Chang et al., 2026). A common remedy for this performance gap is continued pretraining on target-language data, often using parameter-efficient finetuning methods (Pfeiffer et al., 2020; Razumovskaia et al., 2025). However, evidence suggests that for smaller models, full-parameter finetuning yields better results than parameter-efficient alternatives (Yong et al., 2023).

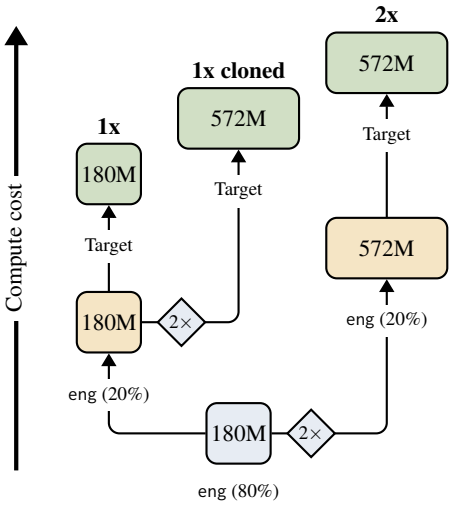


Figure 1: Illustration of our scaling setups. Base models are in yellow, target-language models in green. Arrows represent continued pretraining, with "2×" indicating where we use hypercloning to upscale models.

In this work, we therefore explore *scaling* for adapting pretrained models to new languages. Prior research shows that *upcycling* smaller models can be substantially more efficient than training large models from scratch while achieving comparable or better downstream performance. This is variously referred to as model growth (Du et al., 2024), model expansion (Samragh et al., 2024), or scaling (Wang et al., 2025). Although these methods have proven effective in monolingual contexts, their potential for language adaptation remains underexplored. In this work, we extend this line of inquiry to the multilingual setting. Specifically, we employ HyperCloning (Samragh et al., 2024), which increases the dimensionality of the model's hid-

*Equal contribution.

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 4329–4344
November 6-10, 2026 ©2026 Association for Computational Linguistics

den layers. Crucially, HyperCloning preserves the smaller model's output distribution, thereby giving the larger model a "warm start" by retaining its original capabilities before continued training. Through an extensive set of experiments, we investigate whether scaling can improve language adaptation, and study the trade-offs between compute and data requirements across scaling setups.

Our results show that upscaling is an effective strategy for language adaptation, producing models that are more data-efficient and perform better than non-scaled models. We also show that for lesser-resourced target languages, it is beneficial to first continue training in English after upscaling. In contrast, upscaling directly in the target language is more compute-efficient for higher-resourced languages. While this comes at a cost of data efficiency in the single-target-language setup, multilingually upscaled models are equally data-efficient even without an initial scaling step in English.

2 Background and Related Work

We describe the technical approach underlying our study, *upscaling* (§2.1), and related research on language adaptation (§2.2).

2.1 Upscaling Techniques

Model upscaling aims to reuse the learned behavior of small neural networks when training larger ones.

Chen et al. (2016) proposed Net2Net, a framework for function-preserving widening and deepening of CNNs. This idea was later adapted to Transformers by bert2BERT (Chen et al., 2022), which reuses weights from the current and upper layer of the source model. Gong et al. (2019) applied a stacking approach to transfer knowledge from shallow to deeper BERT models, achieving similar accuracy with fewer training steps. Du et al. (2024) showed that depth-wise stacking (simply duplicating layers) offers the best speedup, while increasing width is less effective. Tokenformer (Wang et al., 2025) replaces a Transformer's linear layers with attention between input tokens (queries) and parameter tokens (keys/values); scaling up then involves adding parameter tokens.

In this work, we use *HyperCloning* (Samragh et al., 2024), a symmetric method for layer expansion that scales up a weight matrix W_S as

$$W_L = \begin{bmatrix} \frac{W_S}{2} + \eta_1 & \frac{W_S}{2} - \eta_1 \\ \frac{W_S}{2} + \eta_2 & \frac{W_S}{2} - \eta_2 \end{bmatrix} \quad (1)$$

where η_1, η_2 are noise to break the symmetry in the continued training after upscaling. Normalization layers and positional embeddings are expanded in the same way, while attention layers are scaled by increasing the number of heads.

2.2 Language Adaptation

We propose upscaling as a means of *language adaptation* through continued pretraining of a base model. Concurrent work (Lee et al., 2026) tests this idea on a single language, whereas we look at a diverse set of twelve target languages.

Continued pretraining is an established strategy for improving target-language performance (Etxaniz et al., 2024; Samuel et al., 2025). It is frequently implemented using parameter-efficient finetuning techniques (Pfeiffer et al., 2020; Razumovskaia et al., 2025; Cui et al., 2024). While effective for larger models, full-parameter finetuning can yield superior performance for smaller models (Yong et al., 2023).

A challenge in continued pretraining is the catastrophic forgetting of previously learned languages (Gogoulou et al., 2024). Elhady et al. (2025) demonstrate that including English during continued pretraining is crucial not only for mitigating forgetting but also for preserving and enhancing capabilities in the target language.

3 Experimental Setup

We describe the training data (§3.1), followed by our upscaling setup (§3.2), our method for compute-matched comparison of models (§3.3), and the multilingual models (§3.4). Finally, we summarize the evaluation datasets and procedures (§3.5)[1].

3.1 Data

Languages We select a diverse set of languages representing different language families, scripts, morphological characteristics and resource levels. In total, we consider twelve target languages, presented in Table 1. The three Germanic languages (Swedish, Icelandic, and Faroese) are Indo-European, with Swedish typologically closest to English. Icelandic and Faroese are morphologically richer and, especially the latter, lesser-resourced. Persian, Southern Pashto and Central Kurdish are also Indo-European but from the Indo-

[1]Models and datasets are available in this collection: https://huggingface.co/collections/liu-nlp/grow-up-and-merge. The code is available here: https://github.com/liu-nlp/multilingual-scaling.

ISO3	Dataset	# Docs	# Tokens	# Seen
	Code	7.68M	3.34B	
eng	English	190M	187B	
ekk	Estonian	10.2M	16.4B	73B
fin	Finnish	36.7M	54.7B	85.5B
sme	Northern Sámi	65.7K	74.9M	449M
fao	Faroese	291K	230M	1.4B
isl	Icelandic	3.01M	4.3B	25.8B
swe	Swedish	59.5M	64.2B	64.2B
ckb	Central Kurdish	555K	1.04B	6.2B
fas	Persian	58.8M	60.5B	60.5B
pbt	Southern Pashto	640K	850M	5.1B
sna	Shona	84.4K	136M	816M
swh	Swahili	1.21M	1.24B	7.4B
zul	Zulu	127K	200M	1.2B

Table 1: Overview of the training datasets. *ISO3* refers to the three-letter language code as per ISO-639-3. *# Tokens* refers to the size of the dataset whereas *# Seen* refers to the number of training tokens.

Iranian branch. Furthermore, they use Arabic-based scripts with minimal token overlap with the Latin-script languages, posing known challenges for multilingual models (Muller et al., 2021; Liu et al., 2024). The Uralic languages (Finnish, Estonian, and Northern Sámi) are agglutinative with case systems, with Northern Sámi being the least-resourced language in our experiments. The Niger-Congo languages (Shona, Swahili, and Zulu) are known for complex noun class systems.

Corpora We use three different data sources for training our models: deduplicated FineWeb-Edu (Lozhkov et al., 2024) and Python-Edu (Allal et al., 2024) for base model pretraining, and the training splits from FineWeb-2 (Penedo et al., 2025) for continued pretraining on target languages. We will refer to these as *English*, *Code* and *Multilingual data*. Table 1 shows the number of documents for each data source, as well as the number of tokens when applying the Llama 3.3 tokenizer and the number of tokens used for training.

Setup We randomly split the English data into an 80% and a 20% subset, where the former is used to train a "seed" model, and the latter is set aside for experiments comparing continued pretraining and upscaling. For target language adaptation, we combine the target-language data with replay data, which has been shown to mitigate catastrophic forgetting (Scialom et al., 2022). For replay, we use random subsets of the English and code data that were used for pretraining the seed model. The amount of replay data is proportional to the amount of training data in the respective target language. For Swedish, we use 1% of the seed model's English data and 5% of the code data; for other languages, we scale this down linearly based on the number of documents in that language (cf. Table 1) compared to Swedish. We also considered alternatives (e.g., leaving out code data), but found this setup to perform best in preliminary experiments. We exclude replay data for the three lowest-resource languages (Northern Sámi, Shona and Zulu) with less than 0.5% of the number of Swedish documents.

3.2 Upscaling Experiments

Architecture For all experiments, we adapt the SmolLM2 architecture (Allal et al., 2025). Following their setup, our smallest models have 180M parameters due to using the heavily multilingual tokenizer of Llama 3.3 with a vocabulary size of 128K. For upscaling, we use HyperCloning (Samragh et al., 2024), as presented in Section 2.1. Since the input and output embeddings in our models are tied, we scale the output embedding matrix at runtime to match the output magnitude of the original model, following the reference implementation.[2] Our training framework is based on nanotron. [3]

Training Details We use a linear warm-up of the learning rate during the first 0.2% of the training steps (with a minimum of 10 steps where applicable) and a linear decay over the last 20% of steps. We train all models with a global batch size of 5,120 samples. For the **1× cloned** models, we add noise and follow Samragh et al. (2024) by applying a signal-to-noise ratio of 10 dB. For all models in our experiments, the total combined training cost was 17,258 hours on NVIDIA A100 40GB GPUs.

Base models Following the findings of Aryabumi et al. (2025), who show that initializing language model training from a code-pretrained checkpoint enhances reasoning capabilities, we first train a 180M-parameter "seed" model on *Code* for two epochs, followed by training on a mix of *Code* and the 80% *English* split for one epoch. From this initialisation, we derive two base models using the 20% *English* split: (a) a 180M-parameter model, obtained by continued pretraining of the seed model as-is, and (b) a 572M-parameter model, obtained by upscaling the seed model via Hyper-

[2] https://github.com/apple/ml-hypercloning
[3] https://github.com/huggingface/nanotron

Cloning with a scaling factor of 2 before continued pretraining. This design ensures that both our base models have seen the same total amount of English data, allowing us to attribute any performance differences solely to scaling effects rather than training data variation.

Target-language models We use three different setups for target language adaptation, illustrated in Figure 1: (1) We continue pretraining the 180M-parameter base model on the target-language data (**1×**). (2) We start from the same base model but scale it to 572M parameters before continuing pretraining on the target language (**1× cloned**). (3) We continue pretraining the 572M-parameter base model on the target-language data (**2×**). Thus, the total amount of training data per language remains unchanged between the three setups. We ensure consistent data ordering between setups.

3.3 Compute-Matched Comparison

In addition to final checkpoint scores, we analyze trajectories across checkpoints. To better understand training dynamics across model scales and scaling strategies, we calculate the total compute cost of each checkpoint, measured in hardware-independent model FLOPs similar to previous work (Hoffmann et al., 2022; Kaplan et al., 2020), including the pretraining cost of their corresponding base model. We record scores at intermediate checkpoints in 250-step intervals, or approximately 2.6 billion tokens for most models. For models with fewer than 250 steps per epoch, we instead save a checkpoint at the end of each epoch. Heineman et al. (2026) show that applying the exponential moving average (EMA) to training curves improves the decision accuracy of identifying the best of two models based on pairs of intermediate checkpoints during training. Following their result, we apply EMA over $N = 20$ checkpoint windows to overcome high checkpoint-to-checkpoint noise, particularly in linguistic acceptability accuracies, and facilitate interpretation of training behavior.

3.4 Multilingual Models

To investigate scaling dynamics during multilingual pre-training, we continuously pretrain three multilingual models. For that, we combine the data for each target language (including replay data) and follow the setups for **1×**, **1× cloned** and **2×**. We refer to these multilingual baselines as **1× (multi)**, **1× cloned (multi)**, and **2× (multi)**.

To determine the total amount of training data for each language, we use a modified version of UniMax sampling (Chung et al., 2023), a strategy aiming for uniform coverage of larger languages while mitigating overfitting on smaller languages by setting a maximum number of epochs. We set the UniMax character budget to 850.225 billion and the maximum number of epochs to 6, following a scaling law for data-constrained models indicating that returns decrease quickly after more than 4 epochs (Muennighoff et al., 2023). We continue pretraining our models for 1 epoch on Swedish and Persian, approximately 1.8 epochs on Finnish, 4.5 epochs on Estonian, and 6 epochs on all other languages. Following the UniMax algorithm exactly with our character budget would have led to slightly more than 1 epoch for Swedish and Persian; to simplify comparisons, we fix the number of epochs for both languages to exactly 1 and redistribute the remaining character budget equally to Estonian and Finnish instead.

We include six multilingual pretrained models from previous work: Gemma 3 270M, Qwen 3.5 0.8B, Gemma 3 1B, Gemma 3 4B and Qwen 3.5 4B. Both Gemma 3 (Kamath et al., 2025) and Qwen 3.5 (Team, 2026) are highly multilingual, supporting more than 140 and 200 languages and dialects, respectively. The composition of languages is not public for the Gemma models, but Qwen 3.5 supports all of our target languages except for Northern Sámi. All models included in our comparisons are base models without instruction tuning.

3.5 Evaluation

Information Parity (IP) Information Parity (Tsvetkov and Kipnis, 2024) is an intrinsic measure of multilingual capability that compares how efficiently a model compresses a target language relative to English, via a ratio of negative log-likelihoods. Values near parity (1) indicate similar encoding efficiency. As IP correlates strongly with downstream performance (Tsvetkov and Kipnis, 2024), it serves as a simple proxy for cross-lingual generalization. In the original definition of IP for multilingual models, the log-likelihood of the English translation of a given text is taken from the same model as the target language log-likelihoods. However, for adaptation to a single target language, this would lead to IPs being artificially high when the English capabilities of the base model are partially lost. Therefore, we use the log-likelihoods of our largest English base model (**2×**) as a refer-

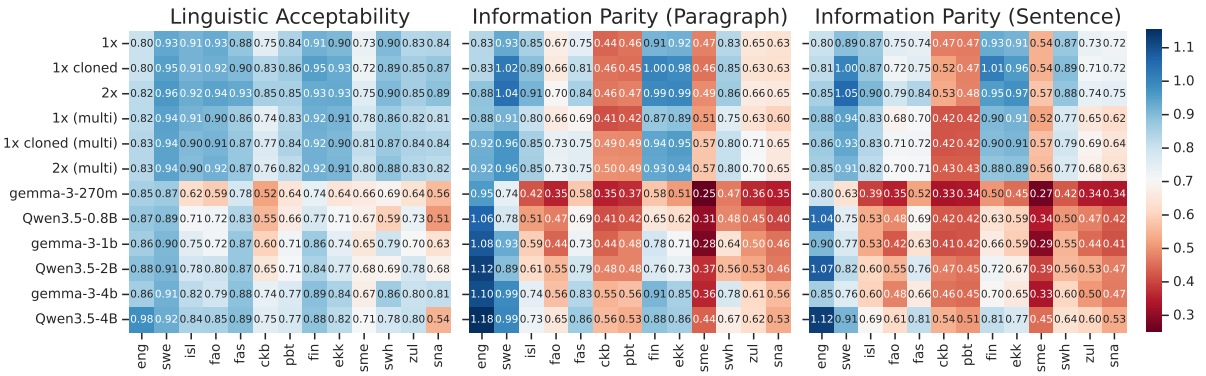


Figure 2: Results for all models and multilingual baselines at different scales. Linguistic acceptability scores are averages over the tasks introduced in Section 3.5. English scores of single-target models are averaged across models.

ence to ensure consistent scores across languages, scales and training setups. This allows us to also report IPs on English as an additional measure of catastrophic forgetting. We compute IP using the BOUQuET dataset (Andrews et al., 2025; Alastruey et al., 2026), which provides parallel sentence-aligned paragraphs across 275 languages.

Linguistic Acceptability (LA) Linguistic Acceptability probes quantify whether a model captures fine-grained grammatical knowledge. This is crucial for assessing whether adaptation yields robust morphosyntactic competence. We evaluate LA using both existing manually annotated and custom automatically generated datasets. All evaluations follow a minimal-pair setup (Linzen et al., 2016), in which each test item consists of one grammatically correct and one incorrect variant, and the model is expected to assign a higher probability to the correct form, and use accuracy over all pairs as the metric. We use two pre-existing benchmarks: English BLiMP (Warstadt et al., 2020) and the multilingual MultiBLiMP (Jumelet et al., 2026). Inspired by MultiBLiMP, we construct a more diverse (in phenomena covered) set of language-specific BLiMP-style datasets for morphology through systematic perturbations based on UniMorph annotations. We sample sentences from high-quality Wikipedia articles (e.g., articles tagged as *excellent* or *article of the month*) and create minimal pairs by replacing a single word with an alternative form that differs in exactly one UniMorph feature (e.g., person, gender, or case). Table 2 in Appendix A provides detailed statistics for each dataset.

Statistical significance testing We use paired bootstrap resampling to evaluate the statistical significance of the observed differences between the three different setups for target-language adaptation. For each of the three pairwise model comparisons, we collect 10,000 resamples per language and calculate the two-sided p-value as

$$p = 2 \min \left(P(\Delta \leq 0), P(\Delta \geq 0) \right) .$$

Finally, we apply the Benjamini–Hochberg correction (Benjamini and Hochberg, 1995) within each pairwise comparison, across languages. We report detailed results in Appendix D.

4 Results

We start by comparing our single-target and multilingual models across scaling setups on the same amount of training data (§4.1). Thereafter, we analyze training dynamics in the compute-matched setting, first for single-target models (§4.2) and then for multilingual models (§4.3). Finally, we probe the internal representations of the multilingual models for language and meaning discrimination (§4.4). Furthermore, to complement and further support the intrinsic evaluation measures reported in this section, we report downstream evaluation results on the SIB-200 topic classification dataset (Adelani et al., 2023) in Appendix B.

4.1 Data-Matched Comparisons

Figure 2 presents the results for training our single-target and multilingual models across different strategies on the same amount of data, alongside strong multilingual baselines from prior work. Our pairwise significance tests (Appendix D) show that all absolute IP differences greater than 0.005 between our data-matched models are significant (adjusted $p < 0.05$) under bootstrap resampling.

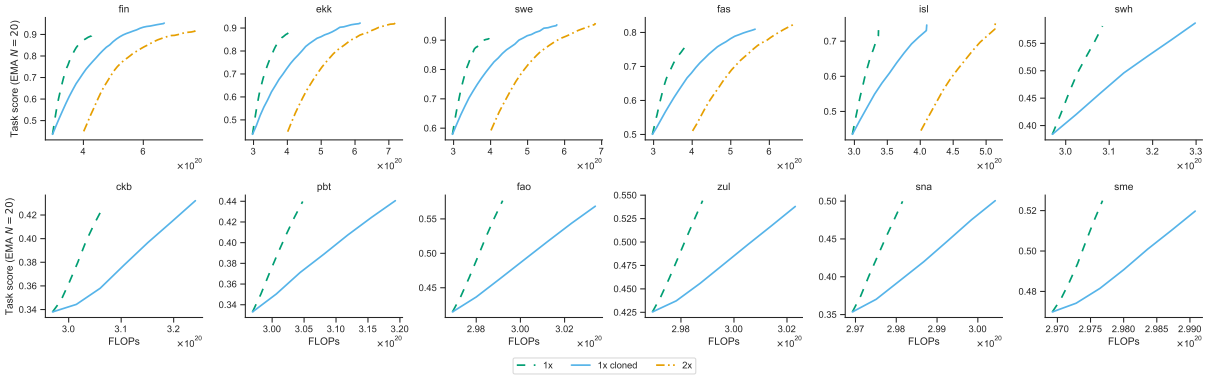


Figure 3: Training curves for models continually pre-trained on a single target language by total training FLOPs, incl. base models. Scores are averaged over all sub-tasks and smoothed with EMA over a window of 20 checkpoints.

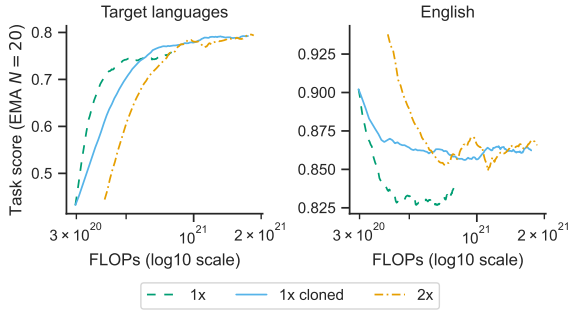


Figure 4: Training curves for models continually pre-trained on our multilingual data mixture by total training FLOPs, incl. base models. Scores are averaged over sub-tasks and smoothed (EMA; window of 20 checkpoints).

Data efficiency All of our monolingual and multilingual models achieve comparable or higher scores on LA and IP compared to the baselines on most target languages, including the largest among them, Qwen3.5 4B and Gemma 3 4B, with over 22 times more parameters than our **1×** models. The exceptions are the IP of the three Indo-Iranian languages and Swedish, as well as Persian LA, for which our **1×** models score higher than baselines at or below one billion parameters. On English, all baselines outperform our continuously pre-trained models. LA scores show only small differences across model scales for our models, but are substantially higher than the scores of the massively multilingual baseline models. This suggests that target-language adaptation is beneficial compared to using a general multilingual model, but that the model capacity necessary to capture syntactic and morphological preferences is relatively small for a specialized model. For the longer contexts in paragraph-level IP, however, both the **2×** models and the multilingual and high-resource single-target **1× cloned** upscaled models clearly outperform the **1×** models, showing that the larger models can handle long contexts more effectively.

Multilingual and single-target Compared to single-target models of the same scale, our *multilingual* models generally underperform on LA and sentence-level IP across most languages. However, on paragraph-level IP the multilingual **2×** and **1× cloned** models often outperform for the lower-resource languages in our study, indicating some positive multilingual transfer in this higher-context setting. Multilingual training is most consistently beneficial for our lowest-resource language, Northern Sámi, and for English across all metrics.

Comparing our scaling strategies in the single-target setting, **2×** is overall more data-efficient than **1× cloned**, particularly in IP and for higher-resourced languages, with the exception of Finnish. **1× cloned** models are still more data-efficient or comparable to **1×** models for higher-resourced languages, but reach lower scores for some of our lowest-resource languages. Overall, **2×** most consistently outperforms **1×** across the languages we consider.

Our multilingual **1× cloned** and **2×** models perform mostly identically on average, with differences of no more than 0.03 for the individual languages across all metrics, and less than 0.01 for paragraph-level IP. The multilingual **1×** model scores close to identically on sentence-level IP and LA tasks. However, we see a clear benefit of scale in the longer context paragraph-level IP scores, in which multilingual **2×** and **1× cloned** substantially outperform **1×** across languages.

Average English scores are nearly identical between **1×** and **1× cloned** but higher in **2×** single-target models. In contrast, **1× cloned** and **2×** En-

glish scores are almost identical and higher than **1×** scores in multilingual models. The exceptions are sentence-level IP scores, which are higher for **1×** models. We further analyze the dynamics of English in multilingual models in §4.3.

4.2 Compute-Matched Single Target Language Scaling Dynamics

When analyzing evaluation scores during training, matched for FLOPs across single-target models (Figure 3), we find that **1×** setups are generally more compute-efficient than **1× cloned** as they can be trained on more data at substantially lower cost. However, given a sufficient amount of training data, **1× cloned** becomes more data-efficient. While **2×** models initially converge faster when matched for training data, **1× cloned** models follow similar learning trajectories. Combined with the lower initial cost of their base models, this results in increased compute efficiency for higher-resource languages. However, as discussed in §4.1, **2×** is more data-efficient for low-resource languages and some higher-resource languages such as Persian.

4.3 Compute-Matched Multilingual Scaling Dynamics

We compare scaling strategies for training multilingual models relative to total training FLOPs separately on an average of target language scores and English scores (Figure 4). As in our single-target experiments, **1×** initially reaches higher scores at a lower compute budget but is overtaken by **1× cloned** approximately halfway through training, and then by **2×** by the end of training. As established in Section 4.1, **1× cloned** and **2×** show comparable levels of data efficiency. By skipping the initial English upscaling step, **1× cloned** reaches these comparable average scores $\approx 1.04 \times 10^{20}$ FLOPs earlier. On English, we find that all models show signs of catastrophic forgetting. **1×** reaches the lowest scores on average. The average score of **1× cloned** decreases more slowly. The **2×** model starts at a higher average score due to the increased capacity of its English base model and initially loses English capabilities more slowly, but reaches a similar level as **1× cloned** by the end of training.

4.4 Language and Meaning Discrimination

To gain further insight into the differences in training dynamics between standard continued pretraining and hypercloning initializations, as well as into the representations across layers, we analyze

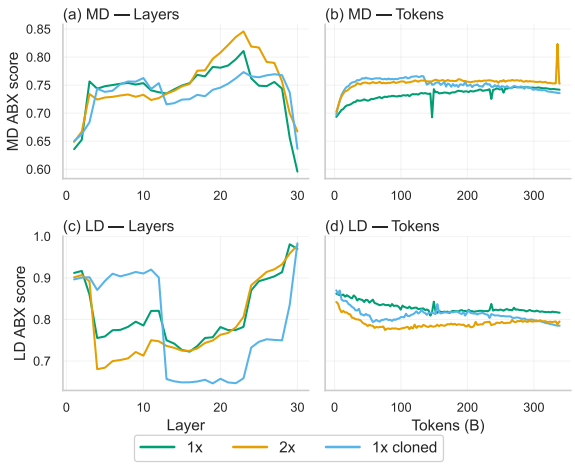


Figure 5: ABX scores for our multilingual models. All scores are averaged over language pairs; checkpoint scores are also averaged over layers.

the multilingual alignment of model representations across intermediate checkpoints and layers. We apply the ABX task proposed by de Seyssel et al. (2025) using parallel sentences from the BOUQuET dataset (Andrews et al., 2025; Alastruey et al., 2026) to investigate how language and meaning differentiation evolve during training in our models. ABX probes use triplets of sentences (X, A, B) and compute the proportion of triplets for which the representation of X is closer to A than to B under cosine distance. Like de Seyssel et al. (2025), we employ two tasks: In *Meaning Discrimination (MD)*, for each language pair (ℓ_1, ℓ_2), triplets are constructed such that X is a sentence in ℓ_1, A is the translation of X into ℓ_2, and B is a different sentence in ℓ_2. A high MD score indicates that representations of translation equivalents are more similar to each other than to unrelated sentences in the same language, i.e. that meaning is encoded cross-lingually. For *Language Discrimination (LD)*, triplets are constructed such that X is a sentence in ℓ_1, A is a different sentence in the same language, and B is the translation of A into ℓ_2. A high LD score indicates that same-language representations cluster together more.

We use ABX probes to study two questions: (1) how the representations change during training, and (2) how the representations are organized across layers in the final model. We compare the three multilingual models (**1×**, **1× cloned**, and **2×**) across both checkpoints and layers[4]. Figure 5 reveals that the three models show clear differences in how

[4]Appendix C contains additional results on monolingual models and ablations across languages and groupings.

their representations are learned and organized.

Across layers **MD** scores increase quickly in the lower layers, peak in the middle layers, and decrease again toward the final layers. This matches earlier work showing that middle layers encode more language-neutral semantic information (Tenney et al., 2019; Jawahar et al., 2019). The **2×** and **1× cloned** models behave similarly in the early layers, but begin to differ around layers 12–13. The **2×** model continues improving and reaches the highest MD scores overall. In contrast, the **1× cloned** model drops strongly in the middle layers and recovers only partly in later layers. The **1×** model has lower MD scores throughout. **LD** scores show the opposite pattern. They are highest in the first and last layers, where more token-level and language-specific information is processed. The **1×** and **2×** models reduce LD quickly in the early layers. The **1× cloned** model behaves differently: it keeps higher LD for longer, then drops sharply around layers 12–13 (the same region where its MD scores decline) and stays below the other models until the final layers. So, although **2×** and **1× cloned** reach very similar performance in evaluations, it appears that the representational space of the **2×** model is more semantically aligned across languages. In contrast to the other models, the **1× cloned** model has a *more* language-distinct representation in the first 12 layers, but a *less* language-distinct representation in layers 13–28. An ablation in Appendix C (Figure 7) shows that this happens independently of script or language family differences.

Across checkpoints The checkpoint analysis shows how these differences develop during training. For **MD**, early in training, **1× cloned** model scores highest, followed by **2×** and then **1×**. Around 135–145B tokens, however, MD in the **1× cloned** model starts to *decrease* (as the only model) and eventually falls below both other models. A closer layer-wise analysis (Appendix C, Figure 8) shows that the decline mainly happens in the middle layers, while the final layer is less affected, which explains why the models do not differ much in their final *performance*, but only in their *representational space*. The **LD** scores show that all models become less language-discriminative during training, but with different dynamics. The **1×** model decreases steadily, the **2×** model first decreases and then partially recovers, and the **1× cloned** model fluctuates more strongly. By the end of training, the **1× cloned** and **2×** models reach similar LD levels, while the **1×** model retains the most language-specific representations. Overall, we observe that **1× cloned** shows the most unstable behavior, which can also be traced to the middle layers (Appendix C).

5 Discussion

The results from Section 4 demonstrate that scaling via HyperCloning improves target-language performance over regular continuous pre-training given the same data, supported by the statistical significance tests presented in Appendix D. Both our single-target and multilingual models outperform heavily multilingual state-of-the-art baselines with as much as 22 times more parameters on eight out of thirteen languages, showing that heavy multilinguality still incurs a cost, particularly for lower-resource languages.

When comparing **1× cloned** and **2×** strategies for continued pre-training for single target languages, we find that **1× cloned** is generally more compute-efficient when factoring in the additional cost of upscaling on English first. However, data efficiency varies between languages. **2×** is generally equivalent or more data-efficient than **1× cloned** except for Finnish. This is particularly the case for some of the lowest resource languages in our comparisons, such as Northern Sámi, Shona, and Zulu, where **1× cloned** models are not trained long enough to optimally utilize their newly increased parameter space. For limited compute budgets, **1×** models are generally more efficient as they can ingest more data with fewer FLOPs at the cost of lower data efficiency. This suggests **1×** could still be considered in highly compute-constrained environments, particularly when data is plentiful.

Moreover, particularly when adapting larger models, scaling may not be desirable due to the increase in inference cost when deployed. In contrast, as established in language model scaling law literature (Hoffmann et al., 2022; Kaplan et al., 2020), the smaller **1×** models hit a point of diminishing returns earlier, particularly in high-resource settings. Therefore, where data efficiency is critical, such as for low-resource languages, we show that investing additional compute into scaling in a high resource base model's language first can be more data efficient. However, the degree of improvement varies by target language.

When training multilingual models, we find that upscaling directly on the multilingual data outperforms continuously pre-training from our smaller model and performs almost identically to upscaling on English first, making it the ideal scaling setup for increasing multilingual capacity in our experiments. Furthermore, while English forgetting occurs in all models in our setup, **1× cloned** models show less forgetting than **1×** models and similar final English scores to **2×** at the end of training, despite the scaled base model of **2×** being more proficient in English initially. The greatest differences between model scales were revealed by paragraph-level information parity scores, showing the utility of longer documents in BOUQuET and similar translation datasets for more robust and discriminative evaluation of multilingual models.

Through an analysis of the internal representations of the multilingual models, we find evidence of higher alignment and less language-dependent activations across languages in the **2×** model. While initially following a similar trend during training, alignment scores in **1× cloned** start fluctuating and ultimately drop to the lower level of **1×**, indicating a lower representational stability of **1× cloned** that is not reflected in task scores. Further work is needed to investigate potential impacts on, e.g., cross-lingual generalization.

6 Conclusion

In this paper, we evaluated upscaling as a strategy for training and adapting models to new target languages. We found that upscaling the base model overcomes capacity constraints and improves target-language performance, particularly for high-resource languages and when training multilingual models. The choice of upscaling strategy depends on the use case: upscaling via English first is primarily advantageous for low-resource languages. Upscaling directly on target-language data is more compute-efficient when sufficient target-language data is available at the cost of data efficiency for some languages.

Furthermore, we show that upscaling on data from multiple target languages directly, rather than scaling on English first, is the ideal setup for multilingual models both in terms of data efficiency and reduced English forgetting. Through an analysis of the representation space, we find higher multilingual alignment when scaling on English first but lower stability when directly scaling on target languages, warranting further analysis of the mechanisms and impacts of this effect in future work. While we present a carefully controlled study of adaptation from a monolingual model, future work could investigate scaling as a means to add languages to or reduce language interference in pre-trained multilingual models.

Limitations

We only study the upscaling behavior of models on twelve target languages due to compute constraints. For the same reason, we perform experiments only for a limited number of model sizes, and we can thus only draw conclusions regarding these small sizes. Thus, future work is needed to study the effects of target-language upscaling for a broader range of languages and for larger models. Lastly, we use pretraining datasets published by other researchers due to time and budget constraints. While it was possible to choose a corpus of English that presumably does not contain unethical material due to its educational content, the choice of available large-scale corpora for our target languages is more limited and might contain inappropriate content.

Acknowledgments

This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, by TrustLLM funded by Horizon Europe GA 101135671, and by the Graduate School in Computer Science (CUGS). The computations were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre and by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

References

David Ifeoluwa Adelani, Hannah Liu, Xiaoyu Shen, Nikita Vassilyev, Jesujoba O. Alabi, Yanke Mao, Haonan Gao, and Annie En-Shiun Lee. 2023. SIB-200: A simple, inclusive, and big evaluation dataset for topic classification in 200+ languages and dialects. *Preprint*, arXiv:2309.07445.

Belen Alastruey, Niyati Bafna, Andrea Caciolai, Kevin Heffernan, Artyom Kozhevnikov, Christophe Ropers, Eduardo Sánchez, Charles-Eric Saint-James, Ioannis Tsiamas, Chierh Cheng, Joe Chuang, Paul-Ambroise Duquenne, Mark Duppenthaler, Nate

Ekberg, Cynthia Gao, Pere Lluís Huguet Cabot, João Maria Janeiro, Jean Maillard, Gabriel Mejia Gonzalez, and 11 others. 2026. Omnilingual MT: Machine translation for 1,600 languages. *Preprint*, arXiv:2603.16309.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, and 3 others. 2025. SmolLM2: When smol goes big – data-centric training of a small language model. *Preprint*, arXiv:2502.02737.

Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. 2024. SmolLM-Corpus. `https://huggingface.co/datasets/HuggingFaceTB/smollm-corpus`. Dataset release.

Pierre Andrews, Mikel Artetxe, Mariano Coria Meglioli, Marta R. Costa-jussà, Joe Chuang, David Dale, Mark Duppenthaler, Nathanial Paul Ekberg, Cynthia Gao, Daniel Edward Licht, Jean Maillard, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Eduardo Sánchez, Ioannis Tsiamas, Arina Turkatenko, Albert Ventayol-Boada, and Shireen Yates. 2025. BOUQuET : dataset, benchmark and open initiative for universal quality evaluation in translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27515–27535, Suzhou, China. Association for Computational Linguistics.

Viraat Aryabumi, Yixuan Su, Raymond Ma, Adrien Morisot, Ivan Zhang, Acyr Locatelli, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2025. To code or not to code? exploring impact of code in pre-training. In *The Thirteenth International Conference on Learning Representations*.

Sacha Beniamine, Mari Aigro, Matthew Baerman, Jules Bouton, and Maria Copot. 2024. Eesthetic: A paralex lexicon of Estonian paradigms. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5526–5537, Torino, Italia. ELRA and ICCL.

Yoav Benjamini and Yosef Hochberg. 1995. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300.

Tyler A. Chang, Catherine Arnett, Zhuowen Tu, and Benjamin Bergen. 2026. Goldfish: Monolingual language models for 350 languages. In *Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026)*, pages 3750–3781, Palma, Mallorca, Spain. European Language Resources Association (ELRA).

Cheng Chen, Yichun Yin, Lifeng Shang, Xin Jiang, Yujia Qin, Fengyu Wang, Zhi Wang, Xiao Chen, Zhiyuan Liu, and Qun Liu. 2022. bert2BERT: Towards reusable pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2134–2148, Dublin, Ireland. Association for Computational Linguistics.

Tianqi Chen, Ian Goodfellow, and Jonathon Shlens. 2016. Net2Net: Accelerating learning via knowledge transfer. *Preprint*, arXiv:1511.05641.

Hyung Won Chung, Xavier Garcia, Adam Roberts, Yi Tay, Orhan Firat, Sharan Narang, and Noah Constant. 2023. UniMax: Fairer and more effective language sampling for large-scale multilingual pretraining. In *Proceedings of the Eleventh International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Yiming Cui, Ziqing Yang, and Xin Yao. 2024. Efficient and effective text encoding for Chinese LLaMA and Alpaca. *Preprint*, arXiv:2304.08177.

Maureen de Seyssel, Jie Chi, Skyler Seto, Maartje Ter Hoeve, Masha Fedzechkina, and Natalie Schluter. 2025. Discriminating form and meaning in multilingual models with minimal-pair ABX tasks. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23704–23725, Suzhou, China. Association for Computational Linguistics.

Wenyu Du, Tongxu Luo, Zihan Qiu, Zeyu Huang, Yikang Shen, Reynold Cheng, Yike Guo, and Jie Fu. 2024. Stacking your transformers: A closer look at model growth for efficient LLM pre-training. In *Advances in Neural Information Processing Systems 37*.

Ahmed Elhady, Eneko Agirre, and Mikel Artetxe. 2025. Emergent abilities of large language models under continued pre-training for language adaptation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32174–32186, Vienna, Austria. Association for Computational Linguistics.

Julen Etxaniz, Oscar Sainz, Naiara Perez, Itziar Aldabe, German Rigau, Eneko Agirre, Aitor Ormazabal, Mikel Artetxe, and Aitor Soroa. 2024. Latxa: An open language model and evaluation suite for Basque. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14952–14972, Bangkok, Thailand. Association for Computational Linguistics.

Evangelia Gogoulou, Timothée Lesort, Magnus Boman, and Joakim Nivre. 2024. Continual learning under language shift. In *Proceedings of Text, Speech, and Dialogue: 27th International Conference*, pages 71–84, Berlin, Heidelberg. Springer-Verlag.

Linyuan Gong, Di He, Zhuohan Li, Tao Qin, Liwei Wang, and Tieyan Liu. 2019. Efficient training of BERT by progressively stacking. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2337–2346. PMLR.

David Heineman, Valentin Hofmann, Ian Magnusson, Yuling Gu, Noah A. Smith, Hannaneh Hajishirzi, Kyle Lo, and Jesse Dodge. 2026. Signal and noise: A framework for reducing uncertainty in language model evaluation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Shaoxiong Ji, Zihao Li, Indraneil Paul, Jaakko Paavola, Peiqin Lin, Pinzhen Chen, Dayyán O'Brien, Hengyu Luo, Hinrich Schütze, Jörg Tiedemann, and Barry Haddow. 2025. Emma-500: Enhancing massively multilingual adaptation of large language models. *Preprint*, arXiv:2409.17892.

Jaap Jumelet, Leonie Weissweiler, Joakim Nivre, and Arianna Bisazza. 2026. MultiBLiMP 1.0: A massively multilingual benchmark of linguistic minimal pairs. *Transactions of the Association for Computational Linguistics*, 14:193–216.

Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, Gaël Liu, and 196 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *Preprint*, arXiv:2001.08361.

Jin-woo Lee, Junhwa Choi, Bongkyu Hwang, Jinho Choo, Bogun Kim, Jeongseon Yi, Joonseok Lee, DongYoung Jung, Jaeseon Park, Kyoungwon Park, and Suk-hoon Jung. 2026. SCALE: Upscaled continual learning of large language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 41006–41020, San Diego, California, United States. Association for Computational Linguistics.

Tal Linzen, Emmanuel Dupoux, and Yoav Goldberg. 2016. Assessing the ability of lstms to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535.

Yihong Liu, Chunlan Ma, Haotian Ye, and Hinrich Schuetze. 2024. TransliCo: A contrastive learning framework to address the script barrier in multilingual pretrained language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2476–2499, Bangkok, Thailand. Association for Computational Linguistics.

Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. 2024. FineWeb-Edu: the finest collection of educational content.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, and 88 others. 2024. Gemma: Open models based on Gemini research and technology. *Preprint*, arXiv:2403.08295.

Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Nouamane Tazi, Aleksandra Piktus, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. 2023. Scaling data-constrained language models. In *Proceedings of the Thirty-seventh Conference on Neural Information Processing Systems*.

Benjamin Muller, Antonios Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2021. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 448–462, Online. Association for Computational Linguistics.

Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. 2025. FineWeb2: One pipeline to scale them all – adapting pre-training data processing to every language. *Preprint*, arXiv:2506.20920.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-Based

Framework for Multi-Task Cross-Lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673, Online. Association for Computational Linguistics.

Evgeniia Razumovskaia, Ivan Vulić, and Anna Korhonen. 2025. Analyzing and adapting large language models for few-shot multilingual NLU: Are we there yet? *Transactions of the Association for Computational Linguistics*, 13:1096–1120.

Mohammad Samragh, Seyed Iman Mirzadeh, Keivan Alizadeh-Vahid, Fartash Faghri, Minsik Cho, Moin Nabi, Devang Naik, and Mehrdad Farajtabar. 2024. Scaling smart: Accelerating large language model pre-training with small model initialization. In *Proceedings of The 4th NeurIPS Efficient Natural Language and Speech Processing Workshop*, volume 262 of *Proceedings of Machine Learning Research*, pages 1–13. PMLR.

David Samuel, Vladislav Mikhailov, Erik Velldal, Lilja Øvrelid, Lucas Georges Gabriel Charpentier, Andrey Kutuzov, and Stephan Oepen. 2025. Small languages, big models: A study of continual training on languages of Norway. In *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, pages 573–608, Tallinn, Estonia. University of Tartu Library.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Fine-tuned language models are continual learners. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6107–6122, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Qwen Team. 2026. Qwen3.5: Towards native multimodal agents.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Alexander Tsvetkov and Alon Kipnis. 2024. Information parity: Measuring and predicting the multilingual capabilities of language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7971–7989, Miami, Florida, USA. Association for Computational Linguistics.

Haiyang Wang, Yue Fan, Muhammad Ferjad Naeem, Yongqin Xian, Jan Eric Lenssen, Liwei Wang, Federico Tombari, and Bernt Schiele. 2025. TokenFormer: Rethinking transformer scaling with tokenized model parameters. In *Proceedings of The Thirteenth International Conference on Learning Representations*.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Zheng Xin Yong, Hailey Schoelkopf, Niklas Muennighoff, Alham Fikri Aji, David Ifeoluwa Adelani, Khalid Almubarak, M Saiful Bari, Lintang Sutawika, Jungo Kasai, Ahmed Baruwa, Genta Winata, Stella Biderman, Edward Raff, Dragomir Radev, and Vassilina Nikoulina. 2023. BLOOM+1: Adding language support to BLOOM for zero-shot prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11682–11703, Toronto, Canada. Association for Computational Linguistics.

A UniMorph BLiMP Dataset

We provide details on our custom UniMorph BLiMP dataset in Table 2: Number of samples, number of subsets, and comments on where we did manual improvements on known weaknesses of either the UniMorph in the given language or of our automatic approach of creating the datasets. We release the dataset under the Creative Commons Attribution-ShareAlike 4.0 International License (CC BY-SA), which we inherit from Wikipedia.

B SIB-200 Fine-Tuning

To test the models' performance under task-specific fine-tuning, we fine-tune all models on the topic classification dataset SIB-200 (Adelani et al., 2023), using the original train–development–test split. To control for differences in the number of trainable parameters during task-specific fine-tuning, we always add a classification head with a hidden size of 256 on top of the frozen model models. We use a learning rate of 5e-4, selection of the best checkpoint over 20 epochs, an effective batch size of 16, a cosine learning rate schedule with a warmup ratio of 0.05. We train all models with 10 seeds and report mean and standard deviation. Results are in Figure 6.

Scales We see that the **2×** models consistently outperform the **1×** models, confirming that upscaling benefits target language performance. The role of the **1× cloned** models is less clear: For the monolingual models and the lower-resource languages, they often underperform **1×**. This is in line with our findings from Section 4.2 where **1× cloned** also underperformed for smaller languages. For the larger languages however, the **1× cloned** models are sometimes even better than **2×**.

Language	Rows	Subsets	Comments
ckb	907	11	–
ekk	5182	28	Uses EESTHETIC (Beniamine et al., 2024) instead of UniMorph for broader and higher-quality coverage.
fao	7907	40	–
fas	1584	8	UniMorph entries post-edited; missing non-compound verbs added.
fin	10033	64	–
isl	5200	26	–
pus	3211	19	–
sna	397	10	–
sme	6,125	47	–
swe	2000	10	Archaic forms filtered out.
swh	527	8	–
zul	937	11	–

Table 2: Overview of the custom BLiMP datasets. # Subsets refers to the number of categories of morphological corruptions included in the dataset.

Multilingual versus Bilingual Models For the lower-resource languages (Central Kurdish, Southern Pashto, Faroese, Shona and Zulu), the multilingual models outperform their single-target counterparts by a wide margin. This contrasts with the intrinsic evaluations in Section 4.3, where multilingual models scored similarly or below single-target models of the same scale. We conclude that the multilingual models, trained on more data, are a better base for fine-tuning, although they underperform out of the box as the single-target models are more specialized. Topic classification relies less on fine-grained morphosyntactic knowledge than linguistic acceptability and more on robust semantic representations, which the lowest-resource languages appear to acquire through transfer from other languages in the mixture.

C Extended ABX Results

In the main paper, we focused on the multilingual models under the different scaling setups. Here, we provide additional analyses of the ABX results, including layer-wise behavior, training dynamics, and per-language breakdowns.

Per-layer analysis We recall from Figure 5 that MD scores rise rapidly in the lower layers, plateau in the middle layers, and decrease again in the uppermost layers. While **2×** and **1× cloned** initially behave similarly, the **1× cloned** model begins to decline around layer 12 and recovers only slowly, whereas **2×** continues improving. The **1×** model consistently obtains lower MD scores. LD exhibits the inverse trend. Scores are highest in the first and last layers, indicating stronger language-specific information. The **1×** and **2×** models show a rapid early decrease in LD, whereas **1× cloned** preserves higher LD for longer before dropping sharply around layers 12–13. It then remains below the other models until the final layers. To better understand this behavior, Figure 7 breaks the LD results down by language pair type: within-family versus cross-family, same-script versus different-script, and English versus non-English pairs. The same overall pattern appears in all settings, suggesting that the unusual behavior of the **1× cloned** model is not driven by one specific group of language pairs. Some groups exhibit the pattern stronger than others (in particular, pairs not including English more strongly than pairs including English, and across-family pairs more strongly than within-family pairs), but it is present in all groups analyzed, suggesting a general effect.

Training dynamics across checkpoints. The drop that we observed in MD for the **1× cloned** model in Section 4.4 develops gradually during training and is mainly concentrated in the middle layers. Figure 8 shows there is a breaking point between layers 12–13, while the final layer is stable across models including the **1× cloned** model. This suggests that the instability mainly affects the *intermediate* representations rather than the final output layer, which explains why we do not observe a strong difference between the **1× cloned** and the **2×** model in other evaluations.

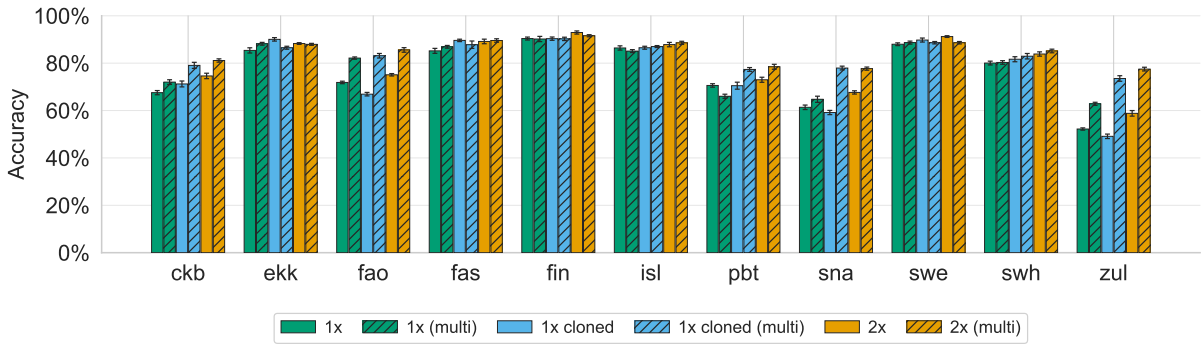


Figure 6: SIB-200 fine-tuning using a classification head.

	Multilingual (eng×lang)						Monolingual (eng×lang)						Multilingual (all pairs)					
	MD			LD			MD			LD			MD			LD		
	1×	**2×**	**1× cloned**	**1×**	**2×**	**1× cloned**	**1×**	**2×**	**1× cloned**	**1×**	**2×**	**1× cloned**	**1×**	**2×**	**1× cloned**	**1×**	**2×**	**1× cloned**
fao	0.645	0.639	0.649	0.803	0.770	0.755	0.629	0.644	0.710	0.791	0.772	0.792	0.773	0.789	0.773	0.790	0.778	0.764
swe	0.833	0.839	0.809	0.771	0.720	0.717	0.901	0.893	0.917	0.764	0.693	0.693	0.809	0.804	0.785	0.780	0.758	0.761
isl	0.618	0.649	0.647	0.788	0.751	0.734	0.710	0.704	0.748	0.780	0.741	0.752	0.784	0.798	0.781	0.806	0.777	0.767
fas	0.742	0.743	0.738	0.815	0.770	0.787	0.812	0.791	0.824	0.821	0.730	0.736	0.750	0.765	0.748	0.820	0.779	0.780
ekk	0.866	0.812	0.776	0.756	0.747	0.779	0.902	0.870	0.891	0.769	0.738	0.775	0.788	0.796	0.777	0.801	0.777	0.772
fin	0.677	0.702	0.679	0.782	0.744	0.736	0.741	0.734	0.772	0.775	0.702	0.692	0.787	0.804	0.785	0.795	0.767	0.759
sme	0.560	0.568	0.580	0.835	0.808	0.801	0.559	0.565	0.582	0.853	0.841	0.890	0.745	0.756	0.743	0.828	0.815	0.805
ckb	0.567	0.541	0.543	0.867	0.891	0.875	0.552	0.549	0.610	0.884	0.922	0.952	0.648	0.622	0.612	0.854	0.861	0.836
pbt	0.504	0.524	0.535	0.858	0.842	0.857	0.512	0.523	0.529	0.861	0.841	0.847	0.655	0.669	0.652	0.833	0.818	0.813
sna	0.580	0.595	0.602	0.840	0.793	0.788	0.602	0.613	0.636	0.810	0.780	0.827	0.749	0.770	0.751	0.828	0.809	0.795
swh	0.626	0.671	0.671	0.807	0.760	0.734	0.713	0.759	0.785	0.772	0.738	0.778	0.783	0.808	0.773	0.811	0.778	0.759
zul	0.553	0.572	0.580	0.843	0.802	0.795	0.570	0.588	0.628	0.803	0.792	0.837	0.724	0.751	0.732	0.853	0.836	0.815

Table 3: ABX scores (last checkpoint, avg over layers).

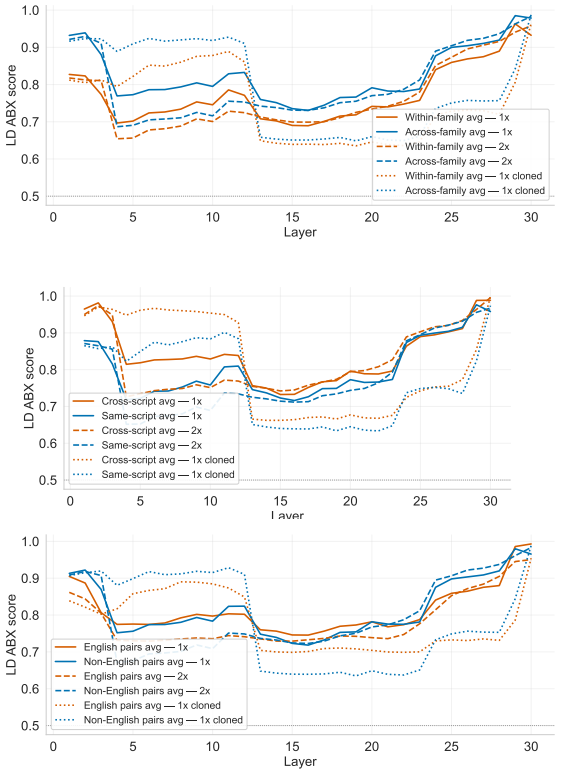


Figure 7: Ablation: LD ABX scores across layers for the three multilingual models, broken down by language pair type (within-family vs. cross-family, same-script vs. different-script, English vs. non-English pairs).

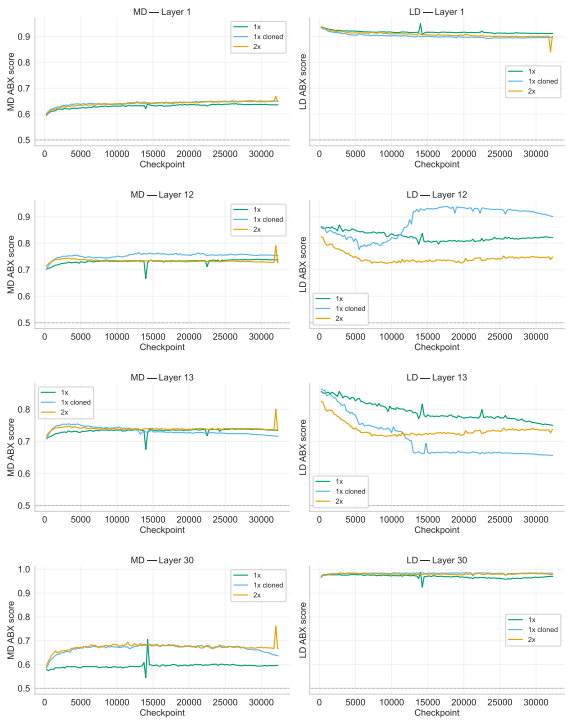


Figure 8: Ablation: A closer look into why the ABX scores for the **1× cloned** model drop during training.

Per-language results (Table 3). There is a notable difference between mono/bilingual and multilingual models. For mono/bilingual models, the **1× cloned** variant often achieves the highest MD scores, while for multilingual models, the **2×** variant tends to score highest. Combined with the checkpoint dynamics above, this may be because mono/bilingual models are trained on less data and thus stop training before the MD decline that affects the multilingual **1× cloned** model sets in.

Results also vary considerably by language. Higher-resource languages like Swedish and Estonian show substantially higher MD scores (0.8–0.9), indicating stronger cross-lingual alignment. Lower-resource languages like Central Kurdish, Southern Pashto, and Northern Sámi have much lower MD scores (0.5–0.6) paired with high LD scores (0.85–0.95). This suggests that when the model has limited data for a language, it isolates that language in its representation space rather than aligning it with others.

D Data Matched Comparison Statistics

We show pairwise score differences between data-matched models with 95% confidence intervals from paired bootstrap resampling for sentence-level and paragraph level IP in Tables 4 and 5 respectively, as described in Section 3.2. Pairs are grouped by single-target and multilingual models. We find that most differences between models are statistically significant in this setup, supporting our results.

Model Family	Language	1x cloned – 1x	2x – 1x	2x – 1x cloned
Single-target	ckb	**0.048 [0.046, 0.05]**	**0.058 [0.056, 0.06]**	**0.01 [0.008, 0.011]**
	ekk	**0.041 [0.038, 0.045]**	**0.06 [0.057, 0.064]**	**0.019 [0.016, 0.022]**
	fao	**-0.033 [-0.035, -0.03]**	**0.037 [0.034, 0.04]**	**0.07 [0.066, 0.073]**
	fas	**0.009 [0.006, 0.012]**	**0.095 [0.091, 0.099]**	**0.086 [0.081, 0.091]**
	fin	**0.085 [0.081, 0.089]**	**0.024 [0.021, 0.028]**	**-0.061 [-0.065, -0.057]**
	isl	**0.006 [0.002, 0.009]**	**0.036 [0.032, 0.04]**	**0.03 [0.027, 0.033]**
	pbt	**0.009 [0.008, 0.01]**	**0.012 [0.01, 0.013]**	**0.003 [0.001, 0.004]**
	sme	-0.001 [-0.003, 0.001]	**0.03 [0.028, 0.031]**	**0.031 [0.029, 0.033]**
	sna	**0.004 [0.002, 0.006]**	**0.027 [0.025, 0.029]**	**0.024 [0.021, 0.026]**
	swe	**0.114 [0.11, 0.118]**	**0.162 [0.157, 0.167]**	**0.048 [0.045, 0.052]**
	swh	**0.021 [0.019, 0.024]**	**0.007 [0.004, 0.011]**	**-0.014 [-0.017, -0.011]**
	zul	**-0.017 [-0.02, -0.015]**	**0.006 [0.004, 0.008]**	**0.023 [0.021, 0.026]**
Multilingual	ckb	-0.001 [-0.003, 0.001]	**0.011 [0.009, 0.012]**	**0.011 [0.01, 0.013]**
	ekk	-0.004 [-0.009, 0.001]	**-0.025 [-0.03, -0.02]**	**-0.021 [-0.024, -0.017]**
	fao	**0.033 [0.029, 0.037]**	**0.021 [0.016, 0.026]**	**-0.012 [-0.015, -0.009]**
	fas	**0.012 [0.008, 0.015]**	0.002 [-0.002, 0.005]	**-0.01 [-0.012, -0.007]**
	fin	-0.0 [-0.005, 0.005]	**-0.019 [-0.024, -0.014]**	**-0.019 [-0.022, -0.016]**
	isl	0.001 [-0.004, 0.005]	**-0.013 [-0.018, -0.009]**	**-0.014 [-0.017, -0.011]**
	pbt	**0.005 [0.003, 0.006]**	**0.011 [0.01, 0.013]**	**0.007 [0.005, 0.008]**
	sme	**0.043 [0.04, 0.047]**	**0.036 [0.033, 0.04]**	**-0.007 [-0.01, -0.005]**
	sna	**0.025 [0.021, 0.029]**	**0.015 [0.011, 0.018]**	**-0.01 [-0.013, -0.007]**
	swe	**-0.01 [-0.015, -0.005]**	**-0.031 [-0.037, -0.026]**	**-0.022 [-0.025, -0.019]**
	swh	**0.014 [0.01, 0.019]**	-0.004 [-0.008, 0.0]	**-0.018 [-0.021, -0.015]**
	zul	**0.041 [0.037, 0.045]**	**0.029 [0.025, 0.033]**	**-0.012 [-0.015, -0.009]**

Table 4: Differences in sentence-level information parity and 95% confidence intervals for each pairwise comparison for each language, separated by single-target and multilingual models. Bold values indicate statistically significant differences ($p < 0.05$) after applying Benjamini-Hochberg correction for each model pair.

Model Family	Language	1x cloned – 1x	2x – 1x	2x – 1x cloned
Single-target	ckb	**0.018 [0.015, 0.02]**	**0.021 [0.019, 0.024]**	**0.004 [0.002, 0.006]**
	ekk	**0.067 [0.061, 0.073]**	**0.076 [0.071, 0.082]**	**0.009 [0.004, 0.014]**
	fao	**-0.01 [-0.012, -0.007]**	**0.029 [0.026, 0.032]**	**0.039 [0.035, 0.042]**
	fas	**0.064 [0.06, 0.069]**	**0.099 [0.093, 0.104]**	**0.034 [0.03, 0.039]**
	fin	**0.089 [0.083, 0.094]**	**0.076 [0.07, 0.081]**	**-0.013 [-0.017, -0.009]**
	isl	**0.038 [0.032, 0.043]**	**0.06 [0.054, 0.066]**	**0.022 [0.018, 0.027]**
	pbt	**-0.009 [-0.011, -0.006]**	**0.017 [0.015, 0.018]**	**0.025 [0.022, 0.028]**
	sme	**-0.015 [-0.017, -0.013]**	**0.015 [0.013, 0.016]**	**0.03 [0.028, 0.032]**
	sna	-0.002 [-0.004, 0.0]	**0.024 [0.021, 0.026]**	**0.026 [0.023, 0.028]**
	swe	**0.092 [0.087, 0.097]**	**0.111 [0.105, 0.116]**	**0.018 [0.014, 0.023]**
	swh	**0.027 [0.024, 0.03]**	**0.034 [0.03, 0.038]**	**0.007 [0.003, 0.011]**
	zul	**-0.012 [-0.015, -0.009]**	**0.013 [0.011, 0.015]**	**0.025 [0.022, 0.028]**
Multilingual	ckb	**0.08 [0.076, 0.084]**	**0.087 [0.084, 0.091]**	**0.008 [0.005, 0.011]**
	ekk	**0.065 [0.059, 0.07]**	**0.059 [0.053, 0.065]**	-0.005 [-0.01, -0.001]
	fao	**0.072 [0.067, 0.077]**	**0.07 [0.065, 0.075]**	-0.002 [-0.006, 0.002]
	fas	**0.062 [0.058, 0.066]**	**0.062 [0.058, 0.067]**	0.0 [-0.003, 0.004]
	fin	**0.064 [0.059, 0.07]**	**0.063 [0.057, 0.069]**	-0.002 [-0.006, 0.003]
	isl	**0.048 [0.043, 0.053]**	**0.052 [0.047, 0.057]**	0.004 [-0.001, 0.008]
	pbt	**0.064 [0.061, 0.068]**	**0.072 [0.068, 0.075]**	**0.008 [0.005, 0.01]**
	sme	**0.067 [0.063, 0.07]**	**0.067 [0.063, 0.071]**	0.0 [-0.003, 0.004]
	sna	**0.057 [0.054, 0.061]**	**0.056 [0.052, 0.06]**	-0.001 [-0.004, 0.002]
	swe	**0.052 [0.046, 0.057]**	**0.051 [0.045, 0.057]**	-0.001 [-0.005, 0.003]
	swh	**0.049 [0.044, 0.053]**	**0.048 [0.043, 0.053]**	-0.001 [-0.004, 0.003]
	zul	**0.075 [0.07, 0.08]**	**0.072 [0.067, 0.077]**	-0.003 [-0.007, 0.001]

Table 5: Differences in paragraph-level information parity and 95% confidence intervals for each pairwise comparison for each language, separated by single-target and multilingual models. Bold values indicate statistically significant differences ($p < 0.05$) after applying Benjamini-Hochberg correction for each model pair.