Beyond Text Compression: Evaluating Tokenizers Across Scales

Jonas F. Lotz[*]
University of Copenhagen, Denmark &
ROCKWOOL Foundation Research Unit
jonasf.lotz@di.ku.dk

António V. Lopes **Stephan Peitz**
Hendra Setiawan **Leonardo Emili**
Apple
{avlopes, speitz, hendra, lemili}
@apple.com

Abstract

The choice of tokenizer can profoundly impact language model performance, yet accessible and reliable evaluations of tokenizer quality remain an open challenge. Inspired by scaling consistency, we show that smaller models can accurately predict significant differences in tokenizer impact on larger models at a fraction of the compute cost. By systematically evaluating both English-centric and multilingual tokenizers, we find that tokenizer choice has negligible effects on tasks in English but results in consistent performance differences in multilingual settings. We propose new intrinsic tokenizer metrics inspired by Zipf's law that correlate more strongly with downstream performance than text compression when modeling unseen languages. By combining several metrics to capture multiple aspects of tokenizer behavior, we develop a reliable framework for intrinsic tokenizer evaluations. Our work offers a more efficient path to informed tokenizer selection in future language model development.

1 Introduction

Language models rely on tokenizers to convert text into machine-interpretable tokens (Grefenstette, 1999). As tokenizers determine how text is segmented, typically into subword units (Sennrich et al., 2016; Kudo, 2018), they fundamentally shape the statistical patterns that language models learn to estimate, thereby impacting both efficiency and downstream performance (Domingo et al., 2019; Bostrom and Durrett, 2020; Ali et al., 2024). Since updating the tokenizer after model training is more cumbersome than ablating other architectural or training decisions (Yong et al., 2023; Zhao et al., 2024), understanding tokenizer impact on model performance prior to large-scale training is crucial.

Tokenizer design and evaluation remain open challenges in NLP (Gowda and May, 2020; Cognetta et al., 2024). *Extrinsic* tokenizer assessments, which involve training a model to measure the impact on performance, are prohibitively expensive for rapid iteration. As a result, *intrinsic* indicators are commonly utilized, with text compression often presented as a strong predictor of performance (Gallé, 2019; Klein and Tsarfaty, 2020; Rust et al., 2021). However, recent studies question the robustness of only considering text compression (Zouhar et al., 2023; Ali et al., 2024; Schmidt et al., 2024), motivating the search for more reliable frameworks.

In this work, we address the practical question of how to select a tokenizer for training a decoder-only language model. We focus on how significant differences in tokenizer quality manifest across both smaller and larger models, informed by evidence that variations in design choices can be traced across model scales (Zohar et al., 2024; Choshen et al., 2024). Concretely, we examine whether performance patterns observed in 350M-parameter models, varying only in their choice of tokenizer, can predict those at the 2.7B-parameter scale. This approach reduces the computational cost of extrinsic evaluation by 85% while isolating tokenizer impact, allowing for a methodical analysis of the relationship between intrinsic tokenizer characteristics and downstream performance.

Recent work has largely confined tokenizer evaluations to monolingual settings (Goldman et al., 2024; Schmidt et al., 2024) or limited multilingual comparisons to classification tasks (Ali et al., 2024). However, as large language models (LLMs) emerge as universal task solvers (OpenAI, 2023), we argue that tokenizer evaluation should also extend to a broader range of applications (Dagan et al., 2024). To understand *when* tokenizer choice matters, we first pretrain 350M-parameter and 2.7B-parameter models on English-centric data. We then systematically evaluate tokenizer impact across four English-centric and two multilingual

[*]Work done during an internship at Apple.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 32155–32173
July 27 - August 1, 2025

tokenizers on multiple-choice benchmarks, summarization, and machine translation into and out of multiple languages and scripts. Our experiments show that tokenizer choice does not have a scale-consistent impact on English-language tasks (likely due to sufficient vocabulary coverage), yet produces persistent differences in translation scenarios. We find that a 350M-parameter model with a multilingual tokenizer can outperform a 2.7B-parameter model that uses an English-centric tokenizer, demonstrating that careful tokenizer selection can offset substantial increases in model size.

To address the need for more reliable intrinsic evaluations, we hypothesize that tokenizers yielding token distributions closely aligned with the statistical patterns of natural language are especially well-suited for generative tasks in that language. Accordingly, we propose four new metrics based on different properties of the token distributions produced on downstream tasks. Although these metrics do not significantly correlate with generative performance in English, they offer more reliable predictions of performance than text compression when modeling previously unseen languages. Finally, we propose a two-stage predictive framework for intrinsic evaluation, combining several tokenizer metrics to produce consistent tokenizer rankings. Our findings offer a practical path toward better-informed tokenizer selection in future language model development.

2 Tokenizer Choice

The choice of tokenizer is a fundamental decision when developing a new language model. Although several aspects of tokenizer creation remain active areas of research (Schmidt et al., 2024; Ali et al., 2024), there is broad consensus that a language model and its tokenizer should ideally be trained on the same data distribution (BigScience Workshop et al., 2023; Hayase et al., 2024). Simply adopting an existing tokenizer can jeopardize this alignment, potentially leading to sub-optimal text encoding (Ahia et al., 2023) and, in severe cases, degraded performance and unintended model behavior (Land and Bartolo, 2024; Geiping et al., 2024). At the same time, developing a custom tokenizer without careful design and validation can introduce severe inequalities and limitations across languages (Ahia et al., 2023; Petrov et al., 2023).

In practice, the resource cost of designing and thoroughly evaluating a new tokenizer likely explains why some models simply borrow a pretrained one from an existing language model. Unfortunately, the exact rationale behind such decisions is rarely documented, leaving the impression that a choice was made after little or no systematic testing. This lack of transparency underscores the need for more reliable yet low-cost methods to guide tokenizer selection.

For our experiments, we evaluate the tokenizers from the following published language models:

PHI-3-MINI English-centric; used by Llama and Llama 2 (Touvron et al., 2023a,b), ALMA (Xu et al., 2024a), Mistral (Jiang et al., 2023), OpenELM (Mehta et al., 2024), and Phi-3-mini (Abdin et al., 2024).[1]

GPT-2 English-centric; used by GPT-2 (Radford et al., 2019), GPT-3 (Brown et al., 2020), Megatron (Shoeybi et al., 2020), OPT (Zhang et al., 2022), and RoBERTa (Liu et al., 2019).

GPT-NEOX English-centric; used by GPT-NeoX (Black et al., 2022), DCLM (Li et al., 2024), OLMo (Groeneveld et al., 2024), and Pythia (Biderman et al., 2023).

FALCON English-centric; used by the Falcon models (Almazrouei et al., 2023).

TIKTOKEN Multilingual; used by GPT-4 (OpenAI, 2023) and the basis for Llama 3 (Dubey et al., 2024).

AYA 23 Multilingual coverage spanning 23 major Asian, European, and Middle Eastern languages; used by Aya 23 (Aryabumi et al., 2024).

3 Proposed Approach

Our experimental approach involves pretraining language models with the chosen tokenizers at two scales, assessing their downstream performance across multiple tasks, and comparing these results with intrinsic tokenizer metrics.

3.1 Model Architecture and Scales

NLP has largely shifted toward decoder-only architectures, driven by their emergent applicability to diverse tasks (Wei et al., 2022; Chowdhery et al., 2023; Tay et al., 2023). Notably, recent advances in machine translation (Xu et al., 2024a,b) challenge

[1]For our experiments, we rely on the MIT-licensed implementation provided with the Phi-3-mini model.

	Vocab size	350M		2.7B	
		$\lvert\theta\rvert$	Hours	$\lvert\theta\rvert$	Hours
PHI-3-MINI	32k	337M	220	2.6B	1840
GPT-2	50k	356M	210	2.7B	1650
GPT-NEOX	50k	356M	210	2.7B	1650
FALCON	65k	371M	210	2.7B	1670
TIKTOKEN	100k	407M	220	2.8B	2050
AYA 23	256k	566M	490	3.2B	2120

Table 1: Vocabulary size, number of trainable parameters ($|\theta|$), and cost of pretraining measured in H100 GPU hours. For simplicity, we refer to the smaller and larger model scales as *"350M"* and *"2.7B"* respectively.

the traditional dominance of encoder-decoder systems (Bojar et al., 2018; Barrault et al., 2019, 2020; Akhbardeh et al., 2021; Kocmi et al., 2022). Following this trend, we restrict our focus to decoder-only transformer models (Vaswani et al., 2017).

We conjecture that if a tokenizer significantly affects model quality, its impact will manifest consistently across different model scales. Although differences only revealed at specific scales may be relevant (Tao et al., 2024), we are primarily concerned with identifying consistent patterns since the same tokenizer is often employed for models of various sizes (Brown et al., 2020; Touvron et al., 2023b; Dubey et al., 2024). Smaller models, with limited representational capacity (Kaplan et al., 2020), are less able to compensate for sub-optimal tokenization (Chai et al., 2024a), making them particularly effective at revealing differences in tokenizer quality. This emphasis on scaling consistency is key to efficient model development (Choshen et al., 2024).

We consider two architecture configurations: a 350M-parameter model aligned with GPT-3 Medium and a model following the GPT-3 configuration for 2.7B trainable parameters. The exact parameter count, shown in Table 1, varies with vocabulary size. Our choice of 350M parameters is inspired by Li et al. (2024), who show that 400M-parameter models can effectively forecast performance trends in larger architectures. Meanwhile, 2.7B-parameter models align with recent studies demonstrating the practical efficiency of the 3B-parameter range (Gunasekar et al., 2023; Li et al., 2023; Abdin et al., 2024; Mehta et al., 2024).

3.2 Pretraining

For each tokenizer, we pretrain models at the two scales. Our pretraining methodology follows the GPT-3 configurations for next-token prediction, with modifications for improved stability and performance. For 350M-parameter models, we increase the maximum learning rate from $3e{-}4$ to $9e{-}4$ based on pilot results showing improved downstream performance. We adopt the weight initialization scheme of Le Scao et al. (2022), whose setup closely resembles ours. Unlike the original GPT-3 architecture, which employs alternating dense and locally banded sparse attention (Child et al., 2019), all models utilize full attention in every layer. All models are trained on the 100B GPT-2 tokens subset of the English-centric FineWeb dataset (Penedo et al., 2024) with a fixed batch size of 2M tokens, exceeding the $20\times$ parameter-count guideline of Hoffmann et al. (2022) and Besiroglu et al. (2024). To improve training stability, beyond the increased batch size, we add an auxiliary z-loss to constrain logit magnitudes (Chowdhery et al., 2023).[2] As reported in Table 1, most 350M-parameter models require around 85% fewer H100 GPU hours to train than their 2.7B-parameter counterpart. Table 7 (Appendix A) presents a complementary analysis based on FLOPs, leading to the same insights.

3.3 Downstream Tasks

We evaluate our models on three task categories: multiple-choice benchmarks, summarization, and machine translation. The emphasis on generative tasks is motivated by their demonstrated sensitivity to tokenizer quality (Goldman et al., 2024).

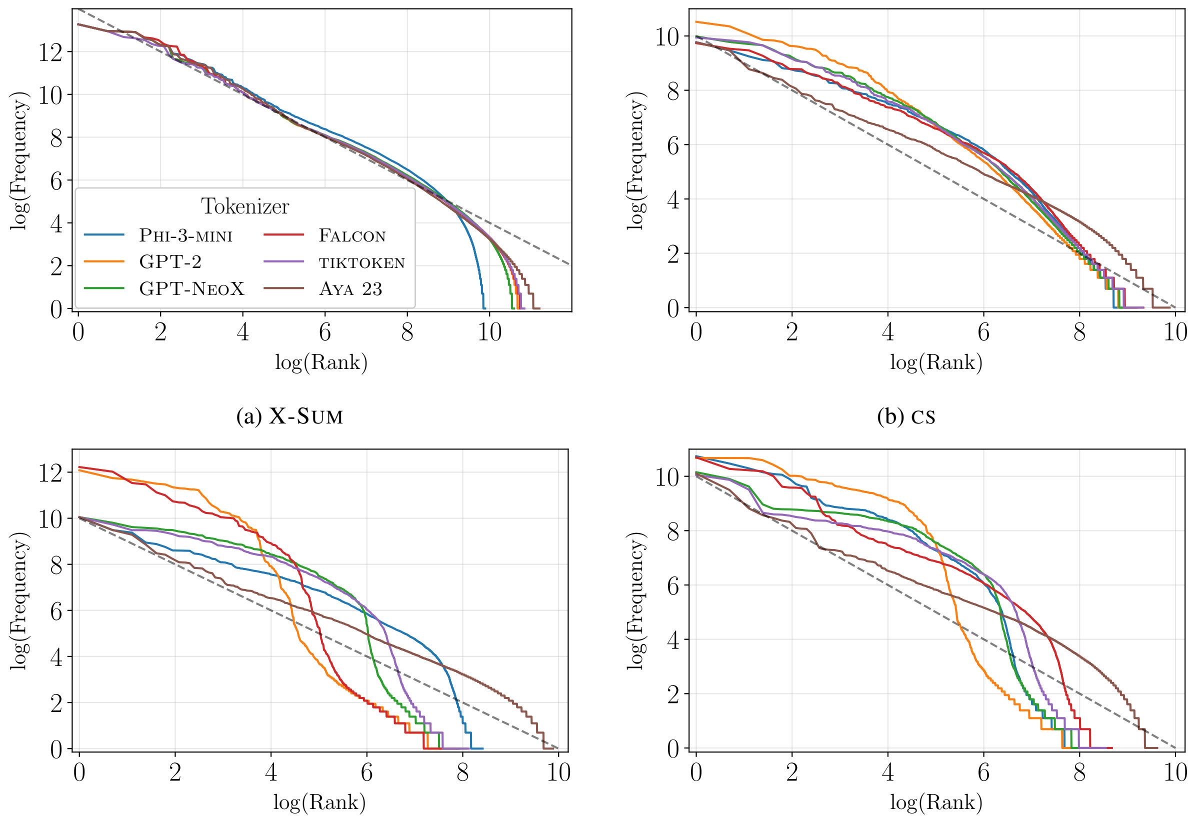
Multiple-choice Benchmarks Multiple-choice tasks represent a cornerstone of modern language model evaluations, serving as the primary framework for assessing zero- and few-shot capabilities across LLMs. We take inspiration from the Open LLM Leaderboard (v1)[3] and LLM360 (Liu et al., 2024) and rely on the LM Evaluation Harness (Gao et al., 2024)[4] to measure performance on:

- Reasoning (R): ARC (Clark et al., 2018), HELLASWAG (Zellers et al., 2019), PIQA (Bisk et al., 2020), and WINOGRANDE (Sakaguchi et al., 2021).

[2] We explored scaling the embeddings by $\sqrt{d}$ as suggested by Takase et al. (2024), but found this approach significantly degraded downstream performance on generative tasks.

[3] This version of the Open LLM Leaderboard was archived in June 2024. Relevant information can be found at `https://huggingface.co/docs/leaderboards/en/open_llm_leaderboard/archive`.

[4] Commit dc90fec at `https://github.com/EleutherAI/lm-evaluation-harness`.



(a) X-Sum (b) cs (c) ru (d) zh

Figure 1: Token frequency plotted against frequency rank in log-log scale for English (X-Sum), Czech (cs), Russian (ru), and Chinese (zh). The dashed lines with a slope of -1 reference a Zipfian power-law distribution.

- Knowledge understanding (KU): MMLU (Hendrycks et al., 2021), and RACE (Lai et al., 2017).
- Misinformation and bias (M&B): CrowS-Pairs (Nangia et al., 2020), and TruthfulQA (Lin et al., 2022).

We rely on the same experimental settings as used by Mehta et al. (2024).

Summarization We finetune our models on the X-Sum dataset (Narayan et al., 2018), which tasks models with generating single-sentence summaries of news articles in English. We evaluate the summaries with SL-BLEURT (Amplayo et al., 2023; Sellam et al., 2020), which offers robust measures of coherence, factuality, fluency, and informativeness. See Table 9 (Appendix A) for details.

Machine Translation To evaluate a tokenizer's contribution to downstream performance when modeling unseen languages and scripts, we perform bi-directional translation experiments across four language pairs with English: Czech (cs $\leftrightarrow$en), German (de $\leftrightarrow$en), Russian (ru $\leftrightarrow$en), and Chinese (zh $\leftrightarrow$en). We finetune our models on the translation data from Xu et al. (2024a) and evaluate performance on the WMT21 test set. Translation quality is assessed using two complementary metrics: the model-based MetricX (Juraska et al., 2023) and the string-based chrF (Popović, 2015). See Table 10 (Appendix A) for details.

3.4 Intrinsic Metrics

Intrinsic evaluation offers a computationally efficient way to assess tokenizer quality without the cost of model training. Throughout, we measure text compression as the number of tokens after tokenizing a sequence (compression). To broaden our understanding of tokenizer behavior, we examine multiple aspects of tokenization beyond compression alone.

From statistical linguistics, word frequency distributions of natural language follow Zipf's law (Zipf, 1935, 1949), a pattern that holds across languages (Piantadosi, 2014). However, it remains unclear how well tokenizers capture this fundamental corpus statistic (Gerz et al., 2018),

	Model size	0-shot						5-shot	25-shot	Avg. R	Avg. KU	Avg. M&B
		ARC	HELLASWAG	PIQA	WINOGRANDE	RACE	TRUTHFULQA	MMLU	CROWS-PAIRS			
PHI-3-MINI	350M	25.3	44.9	69.8	51.9	31.1	38.5	25.9	**63.9**	48.0	28.5	51.2
GPT-2	350M	25.3	45.9	**71.2**	**53.7**	31.4	**40.7**	26.2	63.1	49.0	28.8	**51.9**
GPT-NEOX	350M	25.4	45.9	70.5	53.2	**32.8**	37.4	**26.7**	62.4	48.8	**29.8**	49.9
FALCON	350M	24.7	46.7	70.1	53.4	31.9	39.1	25.8	62.6	48.7	28.9	50.9
TIKTOKEN	350M	25.2	45.8	71.2	53.4	31.8	39.5	24.7	59.3	48.9	28.3	49.4
AYA 23	350M	**26.7**	**46.9**	70.7	52.3	32.2	40.2	25.1	63.0	**49.2**	28.7	51.6
PHI-3-MINI	2.7B	28.7	60.0	74.2	56.0	34.6	36.0	25.4	67.1	54.7	30.0	51.6
GPT-2	2.7B	28.9	60.2	75.1	56.8	35.2	34.1	26.8	67.1	55.3	31.0	50.6
GPT-NEOX	2.7B	28.2	60.4	75.3	58.3	35.6	34.0	26.8	66.9	55.6	31.2	50.5
FALCON	2.7B	28.6	**61.9**	**75.8**	**59.0**	**35.9**	**36.3**	**27.0**	**68.5**	**56.3**	**31.5**	**52.4**
TIKTOKEN	2.7B	30.0	60.6	75.4	57.5	36.9	37.3	26.7	62.5	55.9	31.2	49.9
AYA 23	2.7B	**30.4**	61.4	75.6	56.6	34.4	35.5	26.0	68.0	56.0	30.2	51.8

Table 2: Downstream results on the multiple-choice benchmarks (higher is better). The metrics are normalized accuracy for ARC (on the Challenge Set), HELLASWAG, and PIQA; accuracy for WINOGRANDE, RACE, TRUTHFULQA (on the multi-true/MC2 task), and MMLU; and PCT stereotype for CROWS-PAIRS (English version).

and whether preserving the natural rank-frequency distribution in the token space benefits model learning (Wei et al., 2021). We hypothesize that tokenizers yielding token distributions closely aligned with a Zipfian power law may be better suited for downstream modeling of natural language. Additionally, the number of unique tokens produced during tokenization can be seen as an indicator for subword coverage and reliance on fallback units, such as byte-level representations. A higher number of unique tokens might therefore indicate a closer match between the tokenizer vocabulary and the original text distribution.[5]

Figure 1 illustrates token frequencies against frequency rank on log-log scales for the training split of X-SUM and for CS, RU, and ZH from Xu et al. (2024a).[6] Rank-frequency patterns for X-SUM differ minimally across tokenizers, suggesting a potentially smaller performance gap on this task. In contrast, some variation among tokenizers can be seen for CS, while the two non-Latin scripts RU and ZH exhibit the largest distributional differences. Notably, among the six tokenizers, the AYA 23 tokenizer most closely follows a Zipfian pattern for ZH without an overrepresentation of high-frequency tokens (Zouhar et al., 2023), indicating a potential advantage on Chinese text.

Motivated by these linguistic considerations and the observed tokenizer behaviors illustrated in Figure 1, we extend our evaluations beyond text compression with four additional intrinsic metrics:

Number of unique tokens (CARDINALITY) The cardinality of the token set after tokenization.

Rank-frequency AUC (AUC) The area under the curve of sorted token frequency against frequency rank in log-log scale (as illustrated in Figure 1), computed using Simpson's rule.

Slope of linear function (SLOPE) The slope β_1 from estimating a linear function $f(x) = \beta_0 + \beta_1 x$ of token frequency as a function of frequency rank in log-log scale, approximating Zipf's law.

Deviation from linear function (POWER LAW) The mean absolute error from the estimated linear function $f(x)$, $\frac{1}{n}\sum_{i=1}^{n} |\beta_0 + \beta_1 x_i - y_i|$. This metric quantifies how closely the token distribution aligns with a Zipfian power law.[7]

4 Experimental Results

For each task, we evaluate whether intrinsic metrics and model performance at the 350M-parameter scale can reliably predict the relative downstream performances of 2.7B-parameter models. As the models vary only in their choice of tokenizer, the experimental setting isolates tokenizer impact on downstream performance. We measure the monotonic relationship between intrinsic metrics and downstream performance using Spearman's ρ, and compare rankings across scales with Kendall's τ.

4.1 Multiple-choice Benchmarks

Table 2 reports the results for the multiple-choice tasks. As expected, larger models outperform their

[5] Additional aspects of tokenizer quality, such as handling rare words, morphological segmentation, and pre-tokenization (Schmidt et al., 2024; Dagan et al., 2024), warrant dedicated investigations in future work.

[6] DE is plotted in Figure 2a (Appendix A).

[7] For AUC, POWER LAW, and SLOPE, motivated by the evidence that power laws only apply above some minimum (Newman, 2005; Clauset et al., 2009; Moreno-Sánchez et al., 2016), we restrict our analysis to tokens with $\log(\text{rank}) \leq 6$.

	Multiple-choice	Summarization	Machine translation
COMPRESSION	**−0.59****	−0.09	0.77**
CARDINALITY	0.29*	−0.09	**−0.79****
AUC	0.19	0.14	0.77**
POWER LAW	0.0	0.14	0.78**
SLOPE	0.0	−0.43	−0.44*
Across scales	0.33	−0.07	0.87*

Table 3: Correlation analysis for all downstream tasks: Spearman's ρ coefficients between intrinsic metrics and downstream performance at 2.7B parameters (top); Kendall's τ coefficients comparing ranked downstream performances between the two scales (bottom). Statistical significance is denoted as: $^{*}p < 0.05$; $^{**}p < 0.01$.

	Model size	X-SUM	WMT21		
			EN→XX	XX→EN	Avg.
PHI-3-MINI	350M	36.0	11.0	8.9	10.0
GPT-2	350M	37.0	16.9	12.1	14.5
GPT-NEOX	350M	37.3	13.0	9.5	11.3
FALCON	350M	**37.4**	11.9	9.5	10.7
TIKTOKEN	350M	36.5	12.4	10.0	11.2
AYA 23	350M	37.3	**9.3**	**8.0**	**8.7**
PHI-3-MINI	2.7B	42.3	8.5	5.8	7.2
GPT-2	2.7B	**43.0**	11.9	7.2	9.6
GPT-NEOX	2.7B	42.7	10.8	6.5	8.7
FALCON	2.7B	42.1	10.2	6.2	8.2
TIKTOKEN	2.7B	39.0	9.4	6.2	7.8
AYA 23	2.7B	42.8	**8.0**	**5.5**	**6.8**

Table 4: Downstream results on the generative tasks. The metrics are SL-BLEURT (↑) for X-SUM and MetricX (↓) for the WMT21 test set. The results for XX→EN and EN→XX present averages over all tasks for translating into and out of English, respectively.

smaller counterparts on reasoning tasks (Wei et al., 2022), but the gap narrows for knowledge-based tasks and becomes negligible for misinformation and bias tasks. The reduced performance on TRUTHFULQA for larger models is attributable to the *U-shaped scaling* properties of the task as identified by Wei et al. (2023). At 2.7B parameters, the model trained with the FALCON tokenizer generally outperforms the others; however, no clear winner emerges at the 350M scale.

Table 3 summarizes the intrinsic and extrinsic evaluations across all tasks. For these multiple-choice benchmarks, performance at the 2.7B scale cannot be reliably extrapolated from 350M-parameter models, and COMPRESSION emerges as the most significant predictor of average downstream performance.

4.2 Summarization

Results for X-SUM are shown in Table 4. At both model scales, most tokenizers yield similar performances, except for TIKTOKEN underperforming especially at 2.7B. Moreover, the AYA 23 tokenizer demonstrates that multilingual coverage does not hinder English performance compared to English-centric tokenizers.

In Table 3, we observe an insignificant rank correlation between the two scales, and none of the intrinsic metrics emerge as predictive of downstream performance. Indeed, Zipfian patterns may be less informative in English natural language domains, where all evaluated token distributions follow similar power law trends. This result challenges previous findings suggesting that a tokenizer's compression efficiency strongly predicts success in English generation (Goldman et al., 2024). A plausible explanation is that once compression surpasses a certain threshold, further reductions yield diminishing returns and other factors become more decisive.

4.3 Machine Translation

Table 4 also summarizes our machine translation results (chrF scores and detailed outcomes for all translation directions are presented in Table 11 in Appendix A). AYA 23 consistently outperforms the other tokenizers at both model scales for translating into and out of English. Furthermore, the 350M-parameter model using the AYA 23 tokenizer performs comparably to GPT-NEOX at 2.7B and even surpasses GPT-2, which features five times more trainable parameters. This underscores that an appropriate tokenizer can compensate for a substantially smaller parameter count. However, in addition to incurring higher pretraining costs, a larger vocabulary also increases inference time, as detailed in Table 12 (Appendix A).

In Table 3, we observe a significant rank correlation across scales, where only FALCON and TIKTOKEN change places, indicating scale-consistent performance trends when processing non-English data. In contrast to the English tasks, all intrinsic metrics exhibit significant correlations with 2.7B-parameter performance, with CARDINALITY demonstrating the strongest correlation overall.[8]

[8]Performing the same analysis based on chrF instead of MetricX yields similar results. Since MetricX achieves a stronger correlation with human judgment (Freitag et al., 2023), we base the remainder of our analyses on MetricX.

	PHI-3-MINI	GPT-2	GPT-NEOX	FALCON	TIKTOKEN	AYA 23	Avg.
COMPRESSION	0.56	0.0	0.86	0.17	0.67	**0.92**	0.53
AUC	0.56	0.0	0.40	0.31	0.33	**0.92**	0.42
CARDINALITY (C)	0.36	**0.10**	0.33	0.73	0.64	**0.92**	0.51
POWER LAW (P)	**0.76**	0.0	**1.0**	**0.78**	**0.80**	0.85	**0.70**
SLOPE (S)	0.71	0.0	0.67	0.59	0.67	0.64	0.55
C + P + S	0.43	0.67	0.80	0.83	0.89	0.92	0.76

Table 5: Predicting the best tokenizer in pairwise comparisons. For each tokenizer, we report the F_1 score from a logistic regression model estimated on the remaining tokenizers. The best performing setting combines CARDINALITY, POWER LAW, and SLOPE (C + P + S) and is estimated using an SVM with a linear kernel.

	CS	DE	RU	ZH
1st place	PHI-3-MINI AYA 23	PHI-3-MINI **PHI-3-MINI**	PHI-3-MINI **PHI-3-MINI**	AYA 23 **AYA 23**
2nd place	AYA 23 PHI-3-MINI	AYA 23 **AYA 23**	AYA 23 **AYA 23**	FALCON PHI-3-MINI
3rd place	FALCON **FALCON**	FALCON TIKTOKEN	TIKTOKEN **TIKTOKEN**	PHI-3-MINI TIKTOKEN
4th place	TIKTOKEN **TIKTOKEN**	TIKTOKEN FALCON	GPT-NEOX FALCON	TIKTOKEN FALCON
5th place	GPT-2 GPT-NEOX	GPT-NEOX **GPT-NEOX**	FALCON GPT-NEOX	GPT-NEOX **GPT-NEOX**
6th place	GPT-NEOX GPT-2	GPT-2 **GPT-2**	GPT-2 **GPT-2**	GPT-2 **GPT-2**
Kendall's τ	0.73^{*}	0.87^{**}	0.87^{**}	0.73^{*}

Table 6: Bradley-Terry ranking of tokenizers for each language. For every rank, we report the ground truth ranking above (marked in grey) and the prediction below. Correct predictions are emphasized in **bold**. Statistical significance is denoted as: $^{*}p < 0.1$; $^{**}p < 0.05$.

5 Predicting Relative Performances

Our extrinsic evaluations indicate that tokenizer choice does not consistently affect English-centric tasks but plays a more decisive role in multilingual settings. Based on our experimental results in machine translation, we propose a framework to identify the optimal tokenizer from intrinsic metrics.

The correlation analysis in Section 4 suggests that our proposed intrinsic metrics can be as informative as COMPRESSION. However, correlations alone lack nuance; for instance, Table 3 might favor the tokenizer that produces the largest set of unique tokens. This criteria, taken to the extreme, would imply constructing vocabularies based on whole words rather than subwords.

To address these limitations, we propose a two-stage predictive framework that first models pairwise differences and then aggregates these into global rankings.

5.1 Pairwise Comparisons

We assume that more informative metrics lead to better predictive performances in pairwise comparisons. For every pair of tokenizers (i, j), we consider the difference in a given intrinsic metric $X_i - X_j$ and define a binary outcome variable Y_{ij} that equals 1 if tokenizer i outperforms tokenizer j and 0 otherwise. The log-odds of this outcome is modeled via logistic regression:

$$\log\left(\frac{\Pr(Y_i > Y_j)}{\Pr(Y_i < Y_j)}\right) = \beta_0 + \beta_1(X_i - X_j)$$

This formulation allows us to infer the probability that tokenizer i outperforms tokenizer j given their difference in tokenizer characteristic. We iteratively leave out one tokenizer for validation and estimate the logistic model on the remaining five; with four languages (averaging results over both translating into and out of English), this yields 40 pairwise comparisons per metric.[9] This setup simulates the scenario of comparing a new tokenizer against established baselines.

Because the same tokenizers are evaluated across multiple languages, the outcomes are correlated; we therefore focus on predictive success rather than statistical inference. Table 5 reports the F_1 score (the harmonic mean of precision and recall) on the held-out tokenizer for each individual metric. POWER LAW proves the most informative predictor on average, a finding that contradicts the simpler correlation patterns in Table 3. Moreover, combining CARDINALITY, POWER LAW, and SLOPE in a support vector machine (SVM) estimated with a linear kernel improves generalization, particularly when evaluating GPT-2. This underscores the potential of more nuanced intrinsic evaluations that capture multiple aspects of tokenizer behavior.

[9]For every model, we perform a cross-validated search for optimal hyperparameters.

5.2 Global Tokenizer Ranking

To extend pairwise outcomes into a transitive global ranking, we adopt the Bradley-Terry (BT) model (Bradley and Terry, 1952). BT is widely used to rank agents in pairwise competitions, including LLM leaderboards and RLHF algorithms (Christiano et al., 2017; Ouyang et al., 2022).[10,11] Under BT, pairwise outcomes are aggregated into latent skill ratings $\lambda_i > 0$ and the probability that tokenizer i outperforms tokenizer j is modeled as

$$\Pr(i > j) = \frac{\lambda_i}{\lambda_i + \lambda_j} .$$

We derive the BT parameters by first estimating the probability that tokenizer i outperforms tokenizer j using an SVM with an RBF kernel over all intrinsic metrics, applying Platt scaling (Platt, 1999) for calibration. We iteratively leave out one language from the estimation for evaluation. Table 6 compares the resulting BT rankings to the ground-truth, demonstrating how intrinsic metrics can be used to effectively predict tokenizer performance across languages. This approach is particularly appealing when extensive extrinsic evaluation would be computationally prohibitive. One limitation is that the framework does not accommodate ties, which may produce more granular rankings than practically meaningful.

6 Discussion

The better predictive performance of more nuanced intrinsic evaluations in §5 emphasizes the importance of capturing multiple aspects of tokenizer behavior. However, in practice, it is often simpler to assess tokenizer quality based on a single metric when the optimal weighting or combination of multiple metrics is unclear. Our results indicate that deviations from a Zipfian (power-law) distribution serve as the single most informative predictor of multilingual performance (Table 5).

Text compression is a practical measure of efficiency, directly impacting generation speed and computational cost. Meanwhile, the deviation of a token distribution from a Zipfian power law benchmarks a tokenizer's alignment with the structure of natural language. Data-driven subword tokenizers whose token frequencies approximate a Zipfian distribution typically achieve efficient compression by reflecting the natural frequencies of words and phrases. Such tokenizers must support adequate vocabulary coverage to avoid overly relying on a small set of high-frequency subword tokens (Zouhar et al., 2023) and an excessively long tail of low-frequency tokens (Gowda and May, 2020).

As illustrated in Figure 1a for X-SUM, all evaluated tokenizers fall along a similar distributional curve, accurately indicating minimal differences for English generation tasks. While our findings emphasize prioritizing an appropriate token distribution, once tokenizers surpass a certain threshold of distributional alignment—where the choice of tokenizer becomes less critical—optimizing for text compression can become a secondary focus to further improve decoding efficiency.

Future work could explore interactions between these intrinsic metrics to provide more detailed guidance. Moreover, our analyses could be extended to investigate when and how the relative importance of these metrics changes for specialized downstream tasks, such as code generation or biomedical text analysis, where syntactic or domain-specific properties may take precedence.

7 Related Work

The most widely adopted algorithms for training a tokenizer include byte-pair encoding (Sennrich et al., 2016) and unigram language modeling (Kudo, 2018). Recently, vocabulary-free approaches for decoder-only models have been proposed (Tai et al., 2024; Chai et al., 2024b) by rendering text as images (Salesky et al., 2021; Rust et al., 2023). However, these approaches only allow for continuous input representations and still rely on a vocabulary and softmax layer for text generation tasks. Alternatively, byte-based tokenizers (Xue et al., 2022) avoid large vocabularies but produce prohibitively long sequences (Mielke et al., 2021). Larger, multilingual vocabularies, while potentially beneficial for generalization, can be slower during inference (Hofmann et al., 2022; Sun et al., 2023; Petrov et al., 2023); our findings highlight this trade-off as well (Table 12, Appendix A).

Tokenizers are traditionally evaluated by their impact on downstream tasks (Provilkov et al., 2020; Saleva and Lignos, 2023; Yehezkel and Pinter, 2023) or by how well they meet specific design criteria (Klein and Tsarfaty, 2020; Hofmann et al., 2021; Beinborn and Pinter, 2023). For instance,

[10] `https://lmsys.org/blog/2023-12-07-leaderboard/`

[11] The main difference between BT and Elo rating, which has also been utilized for ranking language models (Askell et al., 2021; Bai et al., 2022), is the assumption that skill levels remain static.

Schmidt et al. (2024) focus on English multiple-choice benchmarks, whereas Goldman et al. (2024) include generation tasks and find text compression to be a strong predictor of performance. In contrast, Ali et al. (2024) report that compression is not always reliable for multilingual tasks, challenging its viability as a sole merit for multilingual tokenizers (Stollenwerk, 2023; Martins et al., 2024). Dagan et al. (2024) further discuss how to overcome potential pitfalls when applying a tokenizer to a domain for which it was not designed.

Gowda and May (2020) recommend ensuring that tokens in the long tail of infrequent vocabulary items from a Zipfian distribution are observed at least 100 times during training, enabling the model to effectively learn their distributional properties. Complementary, Zouhar et al. (2023) propose to use Rényi entropy, a generalization of Shannon entropy (Shannon, 1948), as an intrinsic metric for tokenizer evaluation, arguing that efficient tokenizers produce balanced token distributions by avoiding an overrepresentation of high-frequency tokens. However, Cognetta et al. (2024) present counterexamples showing that increasing Rényi effiency by eliminating high-frequency tokens and redistributing their probability mass can negatively correlate with downstream performance. Furthermore, Dagan et al. (2024) find that, contrary to expectations, higher Rényi entropy correlates with lower performance in code generation.

8 Conclusion

We presented a cost-effective approach to tokenizer selection by training 350M-parameter decoder-only models that differ only in tokenizer choice, serving as reliable proxies for 2.7B-scale performance. Our experiments indicate that tokenizer choice is more critical in multilingual scenarios than in tasks limited to the pretraining language (English).

We proposed new intrinsic tokenizer metrics that capture how closely token distributions align with a Zipfian power law. These metrics proved especially useful for determining performance on previously unseen languages. Our results highlight the importance of distinguishing between different experimental settings when evaluating tokenizers, and emphasized that comprehensive intrinsic evaluations should consider multiple aspects of tokenizer behavior. Finally, we presented a reliable framework for ranking tokenizers based on their intrinsic metrics.

Limitations

Our study focuses on decoder-only models up to 2.7B parameters, chosen for their practical relevance. Although our findings provide a strong basis for evaluating tokenizer performance at this scale, we have not verified whether these trends hold for larger architectures. Prior work (Tao et al., 2024) indicates that vocabulary size may need to grow with model size, suggesting that conclusions could differ for models beyond the scales explored here.

Furthermore, while we systematically evaluate tokenizer performance on five different languages, covering three different scripts, the scope of our multilingual experiments remains limited. A wider range of languages could yield different outcomes, especially for scripts or morphological structures not represented in our training data.

We also note that the considered multiple-choice benchmarks are known to exhibit inherent variance (Madaan et al., 2024; Alzahrani et al., 2024), which may amplify or mask performance differences between tokenizers. The results presented here should thus be interpreted with caution and ideally verified by training multiple models with different random seeds.

Finally, we did not explore the sensitivity of our results to multiple random seeds, hyperparameter configurations during downstream tasks, or variations in the pretraining pipeline. Although these choices kept computational demands in check, they may limit the generality of our conclusions. Future work could address these gaps by investigating larger model sizes, additional languages, and more exhaustive hyperparameter searches.

Acknowledgements

We thank Matthias Sperber, Sarthak Garg, Frederik Kølby Christensen, and Desmond Elliott for helpful discussions and feedback. Jonas F. Lotz is funded by the ROCKWOOL Foundation (grant 1242).

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, and 110 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint*.

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. 2023. Do all languages cost the same? tokenization in the era of commercial language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.

Farhad Akhbardeh, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, and 17 others. 2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, Jan Ebert, Niclas Doll, Jasper Buschhoff, Charvi Jain, Alexander Weber, Lena Jurkschat, Hammam Abdelwahab, Chelsea John, Pedro Ortiz Suarez, Malte Ostendorff, Samuel Weinbach, Rafet Sifa, and 2 others. 2024. Tokenizer choice for LLM training: Negligible or crucial? In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3907–3924, Mexico City, Mexico. Association for Computational Linguistics.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models. *arXiv preprint*.

Norah Alzahrani, Hisham Alyahya, Yazeed Alnumay, Sultan AlRashed, Shaykhah Alsubaie, Yousef Almushayqih, Faisal Mirza, Nouf Alotaibi, Nora Al-Twairesh, Areeb Alowisheq, M Saiful Bari, and Haidar Khan. 2024. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13787–13805, Bangkok, Thailand. Association for Computational Linguistics.

Reinald Kim Amplayo, Peter J Liu, Yao Zhao, and Shashi Narayan. 2023. SMART: Sentences as basic units for text evaluation. In *The Eleventh International Conference on Learning Representations*.

Viraat Aryabumi, John Dang, Dwarak Talupuru, Saurabh Dash, David Cairuz, Hangyu Lin, Bharat Venkitesh, Madeline Smith, Kelly Marchisio, Sebastian Ruder, Acyr Locatelli, Julia Kreutzer, Nick Frosst, Phil Blunsom, Marzieh Fadaee, Ahmet Üstün, and Sara Hooker. 2024. Aya 23: Open weight releases to further multilingual progress. *arXiv preprint*.

Amanda Askell, Yuntao Bai, Anna Chen, Dawn Drain, Deep Ganguli, Tom Henighan, Andy Jones, Nicholas Joseph, Ben Mann, Nova DasSarma, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Jackson Kernion, Kamal Ndousse, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, and 3 others. 2021. A general language assistant as a laboratory for alignment. *arXiv preprint*.

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

Loïc Barrault, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, and 2 others. 2020. Findings of the 2020 conference on machine translation (WMT20). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1–55, Online. Association for Computational Linguistics.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Lisa Beinborn and Yuval Pinter. 2023. Analyzing cognitive plausibility of subword tokenization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4478–4486, Singapore. Association for Computational Linguistics.

Tamay Besiroglu, Ege Erdil, Matthew Barnett, and Josh You. 2024. Chinchilla scaling: A replication attempt. *arXiv preprint*.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, and 1 others. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, and 374 others. 2023. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint*.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 95–136, virtual+Dublin. Association for Computational Linguistics.

Ondřej Bojar, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, and Christof Monz. 2018. Findings of the 2018 conference on machine translation (WMT18). In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 272–303, Belgium, Brussels. Association for Computational Linguistics.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624, Online. Association for Computational Linguistics.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs i: The method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yekun Chai, Yewei Fang, Qiwei Peng, and Xuhong Li. 2024a. Tokenization falling short: On subword robustness in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1582–1599, Miami, Florida, USA. Association for Computational Linguistics.

Yekun Chai, Qingyi Liu, Jingwu Xiao, Shuohuan Wang, Yu Sun, and Hua Wu. 2024b. Autoregressive pretraining on pixels and texts. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3106–3125, Miami, Florida, USA. Association for Computational Linguistics.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. *arXiv preprint*.

Leshem Choshen, Yang Zhang, and Jacob Andreas. 2024. A hitchhiker's guide to scaling law estimation. *arXiv preprint*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, and 48 others. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems*, 30.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *arXiv preprint*.

Aaron Clauset, Cosma Rohilla Shalizi, and M. E. J. Newman. 2009. Power-law distributions in empirical data. *SIAM Review*, 51(4):661–703.

Marco Cognetta, Vilém Zouhar, Sangwhan Moon, and Naoaki Okazaki. 2024. Two counterexamples to tokenization and the noiseless channel. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16897–16906, Torino, Italia. ELRA and ICCL.

Gautier Dagan, Gabriel Synnaeve, and Baptiste Rozière. 2024. Getting the most out of your tokenizer for pre-training and domain adaptation. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Miguel Domingo, Mercedes García-Martínez, Alexandre Helle, Francisco Casacuberta, and Manuel Herranz. 2019. How much does tokenization affect neural machine translation? In *International Conference on Computational Linguistics and Intelligent Text Processing*, pages 545–554. Springer.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela

Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 516 others. 2024. The Llama 3 Herd of Models. *arXiv preprint*.

Markus Freitag, Nitika Mathur, Chi-kiu Lo, Eleftherios Avramidis, Ricardo Rei, Brian Thompson, Tom Kocmi, Frederic Blain, Daniel Deutsch, Craig Stewart, Chrysoula Zerva, Sheila Castilho, Alon Lavie, and George Foster. 2023. Results of WMT23 metrics shared task: Metrics might be guilty but references are not innocent. In *Proceedings of the Eighth Conference on Machine Translation*, pages 578–628, Singapore. Association for Computational Linguistics.

Matthias Gallé. 2019. Investigating the effectiveness of BPE: The power of shorter sequences. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1375–1381, Hong Kong, China. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024. A framework for few-shot language model evaluation.

Jonas Geiping, Alex Stein, Manli Shu, Khalid Saifullah, Yuxin Wen, and Tom Goldstein. 2024. Coercing LLMs to do and reveal (almost) anything. In *ICLR 2024 Workshop on Secure and Trustworthy Large Language Models*.

Daniela Gerz, Ivan Vulić, Edoardo Maria Ponti, Roi Reichart, and Anna Korhonen. 2018. On the relation between linguistic typology and (limitations of) multilingual language modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 316–327, Brussels, Belgium. Association for Computational Linguistics.

Omer Goldman, Avi Caciularu, Matan Eyal, Kris Cao, Idan Szpektor, and Reut Tsarfaty. 2024. Unpacking tokenization: Evaluating text compression and its correlation with model performance. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2274–2286, Bangkok, Thailand. Association for Computational Linguistics.

Thamme Gowda and Jonathan May. 2020. Finding the optimal vocabulary size for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3955–3964, Online. Association for Computational Linguistics.

Gregory Grefenstette. 1999. *Tokenization*, pages 117–133. Springer Netherlands, Dordrecht.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, and 24 others. 2024. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809, Bangkok, Thailand. Association for Computational Linguistics.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks are all you need. *arXiv preprint*.

Jonathan Hayase, Alisa Liu, Yejin Choi, Sewoong Oh, and Noah A. Smith. 2024. Data mixture inference attack: BPE tokenizers reveal training data compositions. In *ICML 2024 Workshop on Foundation Models in the Wild*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Thomas Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karén Simonyan, Erich Elsen, and 3 others. 2022. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*, volume 35, pages 30016–30030. Curran Associates, Inc.

Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. 2021. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3594–3608, Online. Association for Computational Linguistics.

Valentin Hofmann, Hinrich Schuetze, and Janet Pierrehumbert. 2022. An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 385–393, Dublin, Ireland. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *arXiv preprint*.

Juraj Juraska, Mara Finkelstein, Daniel Deutsch, Aditya Siddhant, Mehdi Mirzazadeh, and Markus Freitag. 2023. MetricX-23: The Google submission to the WMT 2023 metrics shared task. In *Proceedings of the Eighth Conference on Machine Translation*, pages 756–767, Singapore. Association for Computational Linguistics.

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. *arXiv preprint*.

Stav Klein and Reut Tsarfaty. 2020. Getting the ##life out of living: How adequate are word-pieces for modelling complex morphology? In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 204–209, Online. Association for Computational Linguistics.

Tom Kocmi, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Thamme Gowda, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Rebecca Knowles, Philipp Koehn, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Michal Novák, Martin Popel, and Maja Popović. 2022. Findings of the 2022 conference on machine translation (WMT22). In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1–45, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Sander Land and Max Bartolo. 2024. Fishing for magikarp: Automatically detecting under-trained tokens in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 11631–11646, Miami, Florida, USA. Association for Computational Linguistics.

Teven Le Scao, Thomas Wang, Daniel Hesslow, Stas Bekman, M Saiful Bari, Stella Biderman, Hady Elsahar, Niklas Muennighoff, Jason Phang, Ofir Press, Colin Raffel, Victor Sanh, Sheng Shen, Lintang Sutawika, Jaesung Tae, Zheng Xin Yong, Julien Launay, and Iz Beltagy. 2022. What language model to train if you have one million GPU hours? In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 765–782, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, and 40 others. 2024. Datacomp-lm: In search of the next generation of training sets for language models. In *Advances in Neural Information Processing Systems*, volume 37, pages 14200–14282. Curran Associates, Inc.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: phi-1.5 technical report. *arXiv preprint*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint*.

Zhengzhong Liu, Aurick Qiao, Willie Neiswanger, Hongyi Wang, Bowen Tan, Tianhua Tao, Junbo Li, Yuqi Wang, Suqi Sun, Omkar Pangarkar, Richard Fan, Yi Gu, Victor Miller, Yonghao Zhuang, Guowei He, Haonan Li, Fajri Koto, Liping Tang, Nikhil Ranjan, and 8 others. 2024. LLM360: Towards fully transparent open-source LLMs. In *First Conference on Language Modeling*.

Ilya Loshchilov and Frank Hutter. 2017. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Lovish Madaan, Aaditya K. Singh, Rylan Schaeffer, Andrew Poulton, Sanmi Koyejo, Pontus Stenetorp, Sharan Narang, and Dieuwke Hupkes. 2024. Quantifying variance in evaluation benchmarks. *arXiv preprint*.

Pedro Henrique Martins, Patrick Fernandes, João Alves, Nuno M. Guerreiro, Ricardo Rei, Duarte M. Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, Pierre Colombo, Barry Haddow, José G. C. de Souza, Alexandra Birch, and André F. T. Martins. 2024. Eurollm: Multilingual language models for europe. *arXiv preprint*.

Sachin Mehta, Mohammad Sekhavat, Qingqing Cao, Max Horton, Yanzi Jin, Frank Sun, Iman Mirzadeh, Mahyar Najibikohnehshahri, Dmitry Belenko, Peter Zatloukal, and Mohammad Rastegari. 2024. Openelm: An efficient language model family with open training and inference framework. In *ICML Workshop*.

Sabrina J. Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y. Lee, Benoît Sagot, and Samson Tan. 2021. Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp. *arXiv preprint*.

Isabel Moreno-Sánchez, Francesc Font-Clos, and Álvaro Corral. 2016. Large-scale analysis of zipf's law in english texts. *PLOS ONE*, 11(1):1–19.

Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. 2020. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1953–1967, Online. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

MEJ Newman. 2005. Power laws, pareto distributions and zipf's law. *Contemporary Physics*, 46(5):323–351.

OpenAI. 2023. Gpt-4 technical report.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. In *Advances in Neural Information Processing Systems*, volume 37, pages 30811–30849. Curran Associates, Inc.

Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. 2023. Language model tokenizers introduce unfairness between languages. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Steven T. Piantadosi. 2014. Zipf's word frequency law in natural language: A critical review and future directions. *Psychonomic Bulletin & Review*, 21(5):1112–1130.

John C. Platt. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press.

Maja Popović. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal. Association for Computational Linguistics.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Phillip Rust, Jonas F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. 2023. Language modelling with pixels. In *The Eleventh International Conference on Learning Representations*.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Elizabeth Salesky, David Etter, and Matt Post. 2021. Robust open-vocabulary translation from visual text representations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7235–7252, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jonne Saleva and Constantine Lignos. 2023. What changes when you randomly choose BPE merge operations? not much. In *Proceedings of the Fourth Workshop on Insights from Negative Results in NLP*, pages 59–66, Dubrovnik, Croatia. Association for Computational Linguistics.

Craig W Schmidt, Varshini Reddy, Haoran Zhang, Alec Alameddine, Omri Uzan, Yuval Pinter, and Chris Tanner. 2024. Tokenization is more than compression. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 678–702, Miami, Florida, USA. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Claude E. Shannon. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2020. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint*.

Felix Stollenwerk. 2023. Training and evaluation of a multilingual tokenizer for gpt-sw3. *arXiv preprint*.

Jimin Sun, Patrick Fernandes, Xinyi Wang, and Graham Neubig. 2023. A multi-dimensional evaluation of tokenizer-free multilingual pretrained models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1725–1735, Dubrovnik, Croatia. Association for Computational Linguistics.

Yintao Tai, Xiyang Liao, Alessandro Suglia, and Antonio Vergari. 2024. PIXAR: Auto-regressive language modeling in pixel space. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14673–14695, Bangkok, Thailand. Association for Computational Linguistics.

Sho Takase, Shun Kiyono, Sosuke Kobayashi, and Jun Suzuki. 2024. Spike no more: Stabilizing the pre-training of large language models. *arXiv preprint*.

Chaofan Tao, Qian Liu, Longxu Dou, Niklas Muennighoff, Zhongwei Wan, Ping Luo, Min Lin, and Ngai Wong. 2024. Scaling laws with vocabulary: Larger models deserve larger vocabularies. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yi Tay, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, Denny Zhou, Neil Houlsby, and Donald Metzler. 2023. UL2: Unifying language learning paradigms. In *The Eleventh International Conference on Learning Representations*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023b. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information ProcessingSystems 30: Annual Conference on Neural Information Processing Systems*, pages 5998–6008, Long Beach, CA, USA.

Shibo Wang and Pankaj Kanwar. 2019. Bfloat16: The secret to high performance on cloud tpus. *Blog Post*.

Jason Wei, Dan Garrette, Tal Linzen, and Ellie Pavlick. 2021. Frequency effects on syntactic rule learning in transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 932–948, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jason Wei, Najoung Kim, Yi Tay, and Quoc Le. 2023. Inverse scaling can become U-shaped. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15580–15591, Singapore. Association for Computational Linguistics.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent abilities of large language models. *Transactions on Machine Learning Research*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024a. A paradigm shift in machine translation: Boosting translation performance of large language models. In *The Twelfth International Conference on Learning Representations*.

Haoran Xu, Amr Sharaf, Yunmo Chen, Weiting Tan, Lingfeng Shen, Benjamin Van Durme, Kenton Murray, and Young Jin Kim. 2024b. Contrastive preference optimization: Pushing the boundaries of LLM performance in machine translation. In *Forty-first International Conference on Machine Learning*.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. ByT5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

Shaked Yehezkel and Yuval Pinter. 2023. Incorporating context into subword vocabularies. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 623–635, Dubrovnik, Croatia. Association for Computational Linguistics.

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.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pretrained transformer language models. *arXiv preprint*.

Jun Zhao, Zhihao Zhang, Luhui Gao, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. Llama beyond english: An empirical study on language capability transfer. *arXiv preprint*.

George K. Zipf. 1949. *Human Behaviour and the Principle of Least Effort*. Addison-Wesley.

George Kingsley Zipf. 1935. *The Psychobiology of Language*. Houghton-Mifflin, New York, NY, USA.

Orr Zohar, Xiaohan Wang, Yann Dubois, Nikhil Mehta, Tong Xiao, Philippe Hansen-Estruch, Licheng Yu, Xiaofang Wang, Felix Juefei-Xu, Ning Zhang, Serena Yeung-Levy, and Xide Xia. 2024. Apollo: An exploration of video understanding in large multimodal models. *arXiv preprint*.

Vilém Zouhar, Clara Meister, Juan Gastaldi, Li Du, Mrinmaya Sachan, and Ryan Cotterell. 2023. Tokenization and the noiseless channel. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5184–5207, Toronto, Canada. Association for Computational Linguistics.

A Pretraining and Experimental Details

	Vocab size	350M			2.7B		
		$\|\theta\|$	Hours	TFLOPs	$\|\theta\|$	Hours	TFLOPs
PHI-3-MINI	32k	337M	220	5.35	2.6B	1840	36.06
GPT-2	50k	356M	210	5.58	2.7B	1650	36.62
GPT-NEOX	50k	356M	210	5.58	2.7B	1650	36.62
FALCON	65k	371M	210	5.77	2.7B	1670	37.09
TIKTOKEN	100k	407M	220	6.21	2.8B	1540	38.19
AYA 23	256k	566M	490	8.17	3.2B	2120	43.10

Table 7: Vocabulary size, number of trainable parameters, and cost of pretraining measured in H100 GPU hours and TFLOPs following Chowdhery et al. (2023).

	350M	2.7B
Optimizer	AdamW (Loshchilov and Hutter, 2019)	
Adam β	(0.9, 0.999)	(0.9, 0.95)
Adam ε	$1e{-}8$	
Clip gradient norm	1.0	
Weight decay	0.1	
Peak LR	$9e{-}4$	$1.6e{-}4$
Minimum LR	$9e{-}5$	$1.6e{-}5$
LR schedule	Cosine Decay (Loshchilov and Hutter, 2017)	
LR warmup ratio	0.0	
Batch size	2M tokens	
Tied embeddings	Yes	
Precision	BFloat16 (Wang and Kanwar, 2019)	
Z-loss coefficient	$1e{-}4$	
Training duration	One epoch (100B GPT-2 tokens)	

Table 8: Pretraining details for both model scales. The implementation takes inspiration from `https://github.com/karpathy/nanoGPT`.

	350M	2.7B
Peak LR	$1e{-}4$	
Minimum LR	$1e{-}5$	
LR schedule	Cosine Decay	
LR warmup steps	1000	
Batch size	128	
Precision	BFloat16	
Training duration	10 epochs	
Source prefix	"Article: {`source`}"	
Target prefix	"Summary: {`target`}"	

Table 9: Finetuning details for X-SUM.

	350M	2.7B
Peak LR	$4.5e{-}4$	$8e{-}5$
LR schedule	Inverse Square-root	
Batch size	256	
Precision	BFloat16	
Training duration	3 epochs	
Source prefix	Translate this from $\{Lang_1\}$ to $\{Lang_2\}$: $\{Lang_1\}$: $\{Lang_1 sentence\}$	
Target prefix	$\{Lang_2\}$:	

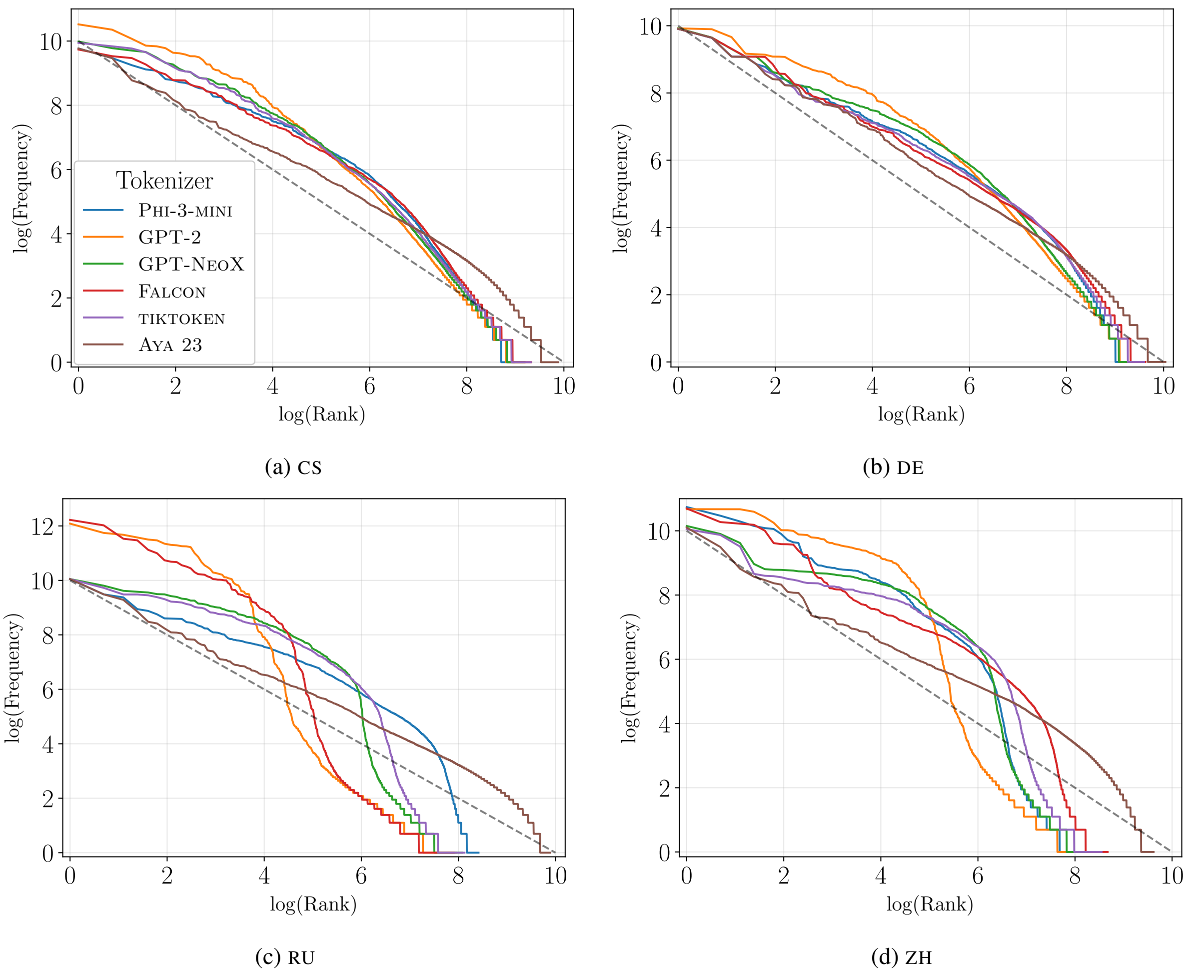
Table 10: Finetuning details for machine translation, where $\{Lang_1\}$ and $\{Lang_2\}$ are the source and target language, respectively, and $\{Lang_1 sentence\}$ is the source sentence.

	Model size	EN→CS		CS→EN		EN→DE		DE→EN		EN→RU		RU→EN		EN→ZH		ZH→EN		EN→XX		XX→EN		Avg.	
		chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX	chrF	MetricX
PHI-3-MINI	350M	**34.6**	10.87	30.5	8.2	48.9	4.95	40.7	5.88	**32.6**	14.02	33.9	8.39	13.2	14.22	26.1	13.24	**32.4**	11.0	32.8	8.9	32.6	10.0
GPT-2	350M	24.7	17.27	21.2	10.81	40.6	8.39	29.2	8.00	28.8	20.98	18.7	12.78	9.3	20.96	17.0	16.66	25.8	16.9	21.5	12.1	23.7	14.5
GPT-NEOX	350M	29.5	13.03	26.8	8.91	46.2	5.79	36.5	6.37	28.5	17.56	28.3	9.1	11.1	15.45	23.9	13.79	28.8	13.0	28.9	9.5	28.9	11.3
FALCON	350M	33.1	11.9	28.4	8.82	47.8	5.12	42.1	5.91	32.0	18.16	26.0	10.71	15.8	12.61	28.2	12.66	32.2	11.9	31.1	9.5	31.7	10.7
TIKTOKEN	350M	29	13.73	23.78	9.82	46	5.72	35.6	7.01	29.37	16.64	27.95	9.55	14.01	13.52	25.03	13.76	29.6	12.4	28.1	10.0	28.8	11.2
AYA 23	350M	32.5	**10.09**	**31.3**	**7.59**	**49.3**	**4.31**	**42.6**	**5.37**	27.1	**13.87**	**37.5**	**7.1**	**17.8**	**9.02**	**29.8**	**11.93**	31.7	**9.3**	**35.3**	**8.0**	**33.5**	**8.7**
PHI-3-MINI	2.7B	**37.2**	**8.98**	36.8	**5.29**	**52.3**	3.56	**49.4**	**3.39**	**37.0**	**11.64**	41.8	5.89	17.4	9.85	36.1	8.61	**36.0**	8.5	41.0	5.8	38.5	7.2
GPT-2	2.7B	34.3	11.12	32.8	6.52	50.5	4.23	47.1	4.03	31.2	18.08	32.0	7.9	14.0	14.33	29.5	10.46	32.5	11.9	35.4	7.2	33.9	9.6
GPT-NEOX	2.7B	30.6	12.28	35.3	5.91	49.2	4.32	47.3	3.84	31.0	14.92	37.4	6.53	15.3	11.67	32.2	9.75	31.5	10.8	38.0	6.5	34.8	8.7
FALCON	2.7B	34.7	10.04	**37.4**	5.38	52.1	**3.51**	48.5	3.77	28.7	17.83	35.5	7.34	18.2	9.37	37.3	8.24	33.4	10.2	39.7	6.2	36.5	8.2
TIKTOKEN	2.7B	35.14	9.97	38.02	5.62	51.69	3.75	49.49	3.78	34.34	13.68	41.29	6.23	17.42	10.1	35.26	9.18	34.6	9.4	41.0	6.2	37.8	7.8
AYA 23	2.7B	35.0	9.47	37.2	5.36	51.6	3.60	49.1	3.41	32.8	12.09	41.0	**5.74**	**21.2**	**6.99**	**40.7**	**7.44**	35.2	**8.0**	**42.0**	**5.5**	**38.6**	**6.8**

Table 11: Detailed machine translation results on the WMT21 test sets measured with chrF (higher is better; `nrefs:2|case:mixed|eff:yes|nc:6|nw:0|space:no|version:2.1.0`), and MetricX (lower is better; `version:metricX23|referenceless:no`). Language codes follow ISO 639-1.

Model	Model size	CS→EN	DE→EN	EN→CS	EN→DE	EN→RU	EN→ZH	RU→EN	ZH→EN
PHI-3-MINI	350M	92.1	95.2	**75.3**	**89.5**	**65.8**	61.1	90.7	79.5
GPT-2	350M	77.6	78.8	58.2	63.5	55.8	56.0	95.7	73.8
GPT-NEOX	350M	83.6	91.6	62.7	74.7	58.3	59.5	90.4	75.2
FALCON	350M	78.7	86.4	64.9	75.8	53.6	59.4	104.8	71.9
TIKTOKEN	350M	77.2	75.7	61.3	70.8	58.9	59.1	77.1	65.5
AYA 23	350M	70.0	76.1	69.4	83.8	60.3	**71.8**	71.6	61.4
PHI-3-MINI	2.7B	97.9	96.9	65.2	72.2	57.9	51.0	96.1	84.8
GPT-2	2.7B	**115.3**	**112.2**	53.0	61.2	44.0	45.7	**169.0**	**98.3**
GPT-NEOX	2.7B	99.0	95.6	48.8	61.6	44.1	44.9	97.3	71.3
FALCON	2.7B	100.6	96.2	57.7	71.9	43.8	50.5	154.3	80.6
TIKTOKEN	2.7B	91.1	87.4	51.3	62.3	43.9	48.5	97.1	73.1
AYA 23	2.7B	74.8	80.3	53.5	64.0	45.9	63.4	68.2	60.2

Table 12: Inference speed (tokens per second) on the WMT21 test set with a batch size of 1 on a single H100 GPU.



(a) CS (b) DE (c) RU (d) ZH

Figure 2: Token frequency plotted against frequency rank in log-log scale for Czech (CS), German (DE), Russian (RU), and Chinese (ZH). The dashed lines with a slope of -1 reference a Zipfian power-law distribution.