When Tokenization Becomes a Theory of Language: English, Chinese, and the Representation of Meaning in LLMs

Steven Blank

May 2026

Abstract

This paper investigates whether apparent English advantages over Mandarin Chinese in large language models arise from intrinsic properties of the languages or from contingent design choices in the model interface. I separate three possible sources of disparity: token compression, fixed-context consequences, and narrower behavioral effects of token identity. First, a tokenization audit of 1,012 aligned English–Chinese sentence pairs from FLORES-200 shows that the Chinese token burden varies sharply across tokenizers: mean Chinese-to-English tokenization parity ranges from 1.89 under `cl100k_base` to 1.01 under Qwen 2.5. Second, fixed-budget XNLI experiments show that this compression gap translates into unequal few-shot capacity, with Chinese prompts fitting substantially fewer demonstrations at the same token budget. Third, an exploratory character-similarity experiment finds a strong effect of shared radicals and a smaller, suggestive effect of shared tokenizer identity. Together, the results support a design-choice explanation for much of the observed English–Chinese disparity, while also showing that tokenization interacts with script structure, training exposure, and model-specific factors rather than providing a complete explanation on its own.

Introduction

English occupies a privileged position in contemporary science, business, and technology, and that linguistic dominance naturally carries over into generative AI. Large language models (LLMs) emerged primarily from American and English-speaking AI research labs, trained on corpora with heavy English representation and were first deployed at scale to largely English-speaking user bases. It is therefore unsurprising that many bilingual users report the impression that English "works better" than many other languages in current LLMs.

This impression is not obviously false, but it is conceptually sloppy. There is serious ambiguity in stating that English "works better" than many other languages in current LLMs. This could mean using fewer tokens (more efficient encoding), cheaper inference, more context being available in the same window or chat, better downstream performance, better reasoning, or more stable and reliable generated outputs. Crucially, these are different phenomena that should not be collapsed into one claim. They are separate performance issues that ought to be treated separately and investigated to understand what actual language-performance barriers

exist and what is causing them.

Recent technical literature suggests that this distinction matters. Petrov et al. argue that tokenizers can introduce substantial unfairness across languages by assigning semantically equivalent text radically different token lengths [2]. Ahia et al. show that these disparities become practical inequalities under token-priced commercial systems and fixed context windows [3]. Rust et al. and Ali et al. further show that tokenizer choice materially affects downstream performance rather than merely changing pre-processing convenience [4, 5]. The relevant question, then, is not simply whether English outperforms Chinese in current systems, but what explains that difference.

This distinction also matters for broader questions about language and thought. Questions about the relation between words, thought, and representation long predate contemporary AI. Aristotle, for example, describes spoken words as symbols of mental experience and written words as symbols of spoken words [1]. Read in that broader frame, multilingual LLM performance gaps should not be naively interpreted as evidence about differences in thought-like capacity. They may instead reflect disparities in the symbolic interface through which models encounter language.

This paper argues that many observed English advantages in current LLMs are better explained by contingent design choices—especially tokenization, vocabulary allocation, context-window economics, and training-data imbalance—than by any intrinsic superiority of English as a language for machine reasoning. The evidence reviewed and the experiments conducted here favor a design-choice explanation over an intrinsic-language explanation. I first clarify the key terms needed to make that distinction precise, then review relevant existing literature, and finally present a set of experiments designed to separate token compression, context-window consequences, and narrower behavioral effects of token identity.

Key Terms

A language should count as intrinsically advantageous for LLMs only if its apparent advantage survives controls for the engineered interface through which the model processes it. This definition of "intrinsic language quality" clarifies that a language being better-suited to LLMs would require surviving controls for technical aspects like tokenizer design, vocabulary size and allocation, context-budget normalization, model family, and training-data exposure. The bar set here is intentionally high: otherwise, a contingent property of the model can be misdescribed as a property of the language.

Since LLMs do not process natural language directly, instead processing a tokenized representation of it, tokenization is not a peripheral implementation detail but part of the model's theory of linguistic units. Tokenization is the mapping from text strings to the discrete units consumed by the model. A byte measures storage; a character, subword, morpheme, or radical is a linguistic or orthographic unit; and a token is the unit that a specific tokenizer actually hands to the model. Models do not process "language" in the abstract. They process a tokenizer-specific representation of language.

This paper uses the term "efficiency" in three ways. *Character efficiency* refers to how much information can be packed into a character string. *Token efficiency* refers to how many model tokens are required to encode a message. *Task efficiency* refers to how much cost or context budget is required to achieve a correct result. In the English–Chinese comparison that follows, these distinctions matter because a script like Mandarin Chinese can look compact visually while still being costly in tokens under a given tokenizer.

Literature Review

A first major result in recent literature is that multilingual disparity can emerge before any inference occurs. Petrov et al. show that semantically matched text can differ dramatically in tokenized length across languages, with disparities reaching an order of magnitude in some settings [2]. This matters because token length is not a neutral bookkeeping detail: it affects cost, latency, and the amount of content that can fit inside a fixed context window. A tokenizer can therefore disadvantage a language before any higher-level reasoning or generation takes place.

Ahia et al. make the practical consequences of this disparity explicit. They show that tokenization inequality becomes an allocation problem in commercial systems (like ChatGPT), because token-based pricing and fixed context windows convert representational inefficiency into unequal access and unequal utility across languages [3]. This is a concrete way in which current systems privilege some languages. Petrov identifies the tokenization problem; Ahia shows that the expected consequences of that problem are already visible in real deployment settings.

The strongest evidence against a simple intrinsic-language story is causal: when the tokenizer is changed, the model's performance changes with it. Rust et al. find that replacing a multilingual tokenizer with a language-specialized one improves downstream performance for almost every task-language pair they study, even while pretraining data size remains important [4]. Ali et al. similarly argue that tokenizer choice is not a negligible engineering detail but a consequential component of LLM training and evaluation [5]. If changing the tokenizer changes the outcome, then at least some of the observed gap must be attributed to the interface rather than to the language alone.

A careful consideration of LLM performance across languages should not replace one mono-causal story with another: tokenization matters greatly, but it interacts with pretraining exposure, vocabulary allocation, and model-scale decisions rather than entirely replacing them as explanations. This is a point reinforced in many instances of multilingual tokenizer design and language-specific adaptation, including the Llama 3 technical report, MYTE, and Tejaswi et al., all of whom treat vocabulary and tokenization as adjustable design levers rather than fixed properties of natural language [6–8].

The English–Chinese contrast is especially informative because it reveals both differences in token burden and insight into the deeper question about what kinds of written substructure a model is permitted to treat as semantically meaningful. The contrast between Mandarin Chinese and English is useful due to the two languages' different script traditions, segmentation

behavior, and the relation between character, morpheme, and token. Chinese is therefore more than a merely convenient non-English comparison case; it is a language in which semantically relevant information often exists below the level of the character. Si et al. argue for sub-character tokenization precisely because standard Chinese tokenization discards information present in glyph- and pronunciation-level structure [9]. ChineseBERT similarly improves Chinese language understanding by explicitly incorporating glyph and pinyin information that ordinary tokenization neglects [10].

Considered together, the existing literature supports a strong but bounded conclusion: many current cross-linguistic LLM disparities are best understood as consequences of representational design rather than as evidence that some natural languages are intrinsically more suitable for machine reasoning. The current evidence supports a design-choice explanation for cross-linguistic LLM disparities more strongly than an intrinsic-language explanation. This conclusion motivates the experiments below, which examine the problem first at the level of token compression, then at the level of context-window consequences, and finally at the level of more tentative behavioral effects.

Experimental Results

The experiments in this paper were designed to separate three distinct sources of cross-linguistic disparity in LLM use: (1) token compression, (2) context-window consequences, and (3) behavioral effects of token identity. They are presented in that order, moving from the cleanest and most interpretable findings to the most exploratory ones. Overall, the experimental results support the claim that much of the apparent English–Chinese disparity arises at the language-model interface rather than reflecting any simple intrinsic superiority of English as a language for LLMs.

Experiment 1: Tokenization Audit

Experiment 1 conducted a tokenization audit on aligned English–Chinese sentence pairs from FLORES-200 devtest, using *tokenization parity*, defined as TP = zh_tokens/en_tokens, as the primary metric. Under this metric, TP = 1 indicates parity; TP $>$ 1 indicates that Chinese consumes more tokens than its English translation; and TP $<$ 1 indicates the reverse. The results showed that Chinese incurred a substantial token burden under some tokenizers but not others.

The largest penalty appeared under cl100k_base, where mean TP was 1.89, meaning that a Chinese sentence consumed nearly 1.9$\times$ the context budget of its English counterpart. This disparity narrowed substantially under o200k_base and the Llama 3 tokenizer [6], which yielded mean TPs of 1.28 and 1.31 respectively. Most strikingly, Qwen 2.5 approached parity at mean TP = 1.01, indicating that the English–Chinese gap is highly tokenizer-dependent rather than fixed across model families.

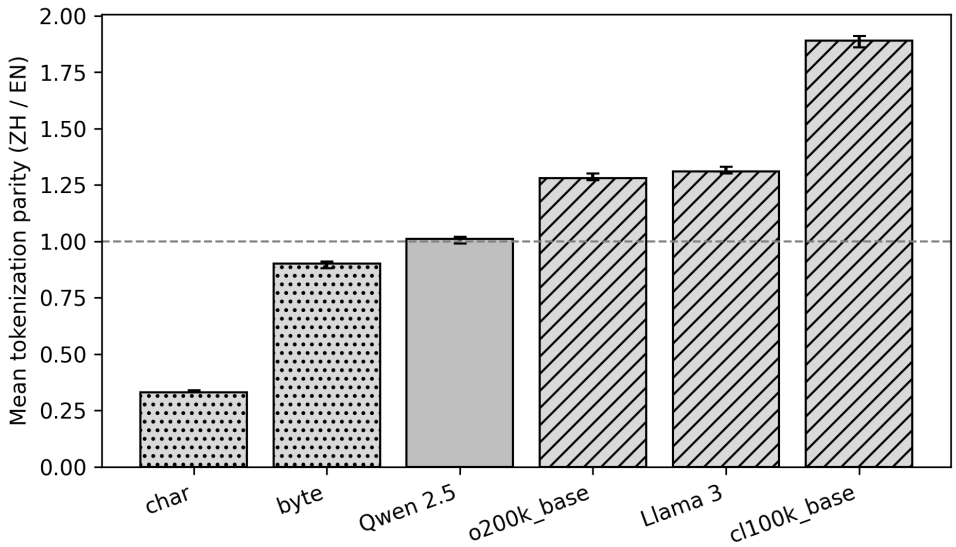

Figure 1: Mean tokenization parity (ZH / EN) on FLORES-200 EN–ZH devtest ($N = 1012$), broken out by tokenizer. The dashed line marks parity (TP = 1.0); error bars show 95% paired-bootstrap confidence intervals. Chinese incurs a much larger token burden under `cl100k_base` than under `o200k_base`, Llama 3, or Qwen 2.5—a tokenizer-dependent gap, not a property of the language itself.

The byte and character baselines are especially important because they show that the observed penalties are not straightforwardly inherited from the script itself. At the byte level, Chinese was actually cheaper than English, with mean TP = 0.90. At the character level, the asymmetry was even stronger, with mean TP = 0.33, reflecting the fact that a single Han character often corresponds to substantially more English orthographic material. These controls matter because they show that the large Chinese penalty under tokenizers such as cl100k_base is not an inevitable property of Chinese writing. The Chinese penalty is rather induced by the tokenizer's merge structure and vocabulary allocation. Experiment 1 therefore provides the clearest evidence in this work that cross-linguistic disparity can arise at the representational interface before any downstream reasoning or task performance is measured.

Table 1: Tokenization audit summary on FLORES-200 EN–ZH devtest ($N = 1012$ paired sentences). TP = mean ZH tokens / EN tokens; 95% CI from a paired bootstrap with 1000 resamples.

Tokenizer	Mean TP	95% CI	Mean EN tokens	Mean ZH tokens
char	0.33	[0.33, 0.34]	130.4	42.8
byte	0.90	[0.88, 0.91]	130.5	116.0
Qwen 2.5	1.01	[0.99, 1.02]	27.3	27.4
o200k_base	1.28	[1.27, 1.30]	26.6	33.8
Llama 3	1.31	[1.30, 1.33]	26.9	35.0
cl100k_base	1.89	[1.86, 1.91]	26.9	50.3

Experiment 2: Fixed-context fairness

Experiment 2 tested the downstream consequence of this compression gap in a fixed-context, few-shot setting. Using XNLI with English and Chinese prompts, I measured how many

demonstrations could be packed into a shared budget of 512, 1024, or 2048 tokens. The served model was gpt-4o-mini and the tokenizer used to measure the budget was tiktoken_cl100k. At every budget, Chinese prompts fit fewer demonstrations than English prompts.

At 512 tokens English fit an average of 11.8 demonstrations while Chinese fit only 6.4, an approximately 45% reduction. Similar reductions were observed at 1024 and 2048 tokens, but the relative difference shrank as the budget grew. The central result of Experiment 2 is therefore not primarily about accuracy but about unequal few-shot capacity under a shared token budget.

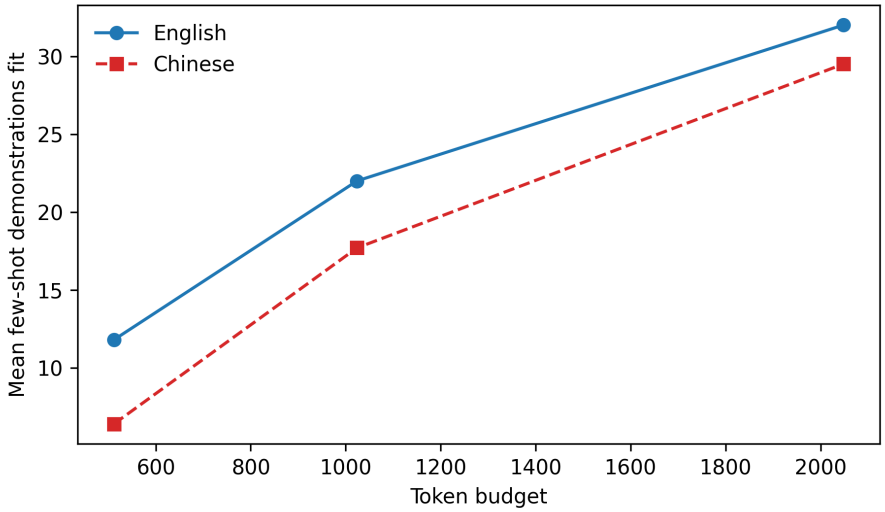

Figure 2: Mean number of few-shot demonstrations that fit inside a fixed token budget on XNLI, by language. At every shared budget the Chinese prompt fits fewer demonstrations than the English one. The figure visualizes the context-budget consequence of the parity gap in Figure 1; it is not an accuracy claim.

Although English accuracy was numerically higher in two of the three budget conditions, the experiment was not designed to support a strong claim about English superiority on XNLI as such. At this sample size, the accuracy differences are modest and should be interpreted cautiously. What the experiment does clearly establish is that tokenization changes how much few-shot scaffolding the model receives in each language under identical budget constraints. In that sense, Experiment 2 translates Experiment 1's compression disparity into a concrete opportunity-cost effect: what appears on the surface as a performance gap can begin as a bandwidth-allocation gap.

Experiment 3: Radical Sensitivity

Experiment 3 asked whether tokenization might affect model judgments through more than compression—specifically, through the structure of the input representation itself. To test this, I constructed controlled Chinese character pairs and triples that varied along two dimensions: whether the characters shared a semantic radical and whether they shared the same first token under cl100k_base. The choice of this tokenizer was deliberate: unlike many tokenizers which map each Han character to a single id, cl100k_base fragments many characters into multiple byte-level pieces and therefore creates meaningful variance in the same_token condition. The

main task was a similarity judgment asking whether two characters were similar in meaning, with odd-one-out judgments used as a secondary task.

The strongest effect in Experiment 3 came from visible script structure. The model judged same-radical pairs to be similar 26% of the time, compared with only 1% for different-radical pairs. This yields a Haldane-corrected odds ratio of 23.6 (see Table 2). Within same-radical pairs, however, token identity still appeared to matter: when two characters shared the same first tokenizer token, the probability of a "similar" judgment rose from 23% to 50%. The logistic regression results point in the same direction, with same_radical having a large positive coefficient and same_token a smaller positive coefficient ($\beta = 3.41$ vs. 1.19).

Interpreting this experiment requires care. The dominant driver of similarity judgments here is clearly radical identity—visible script structure—rather than tokenization. At the same time, the token effect suggests that the tokenizer may introduce an additional learned bias that cannot be reduced to radical identity. This does not reveal anything about the model's internal representations, and it does not show that tokenization is the primary driver of Chinese semantic behavior. What it does show is a narrower but still important finding: once script-family similarity is partially controlled for, token identity still appears capable of shifting outputs. Experiment 3 therefore provides the strongest evidence in this report that tokenization may influence behavior in addition to compression.

Table 2: Behavioral evidence that token identity modulates the model's character-similarity judgments after controlling for radical (script) identity.

Quantity	Estimate	Detail
P(predicted similar | same radical)	0.260	$n = 100$
P(predicted similar | different radical)	0.010	$n = 100$
Odds ratio (same vs. different radical)	23.60	Haldane–Anscombe corrected
Fisher exact p	7.3×10^{-8}	two-sided
Logit β(`same_radical`)	+3.41	$p = 0.001$
Logit β(`same_token`)	+1.19	$p = 0.080$

Synthesizing Experimental Results

Considered together, the experiments form a coherent progression. Experiment 1 shows that the English–Chinese gap is highly tokenizer-dependent at the level of compression. Experiment 2 shows that this compression gap becomes an inequality in few-shot capacity under fixed token budgets (512, 1024, 2048). Experiment 3 suggests that tokenization may also influence model judgments beyond compression itself, although visible script structure remains the dominant factor. Across the experimental section as a whole, the strongest pattern is that cross-linguistic disparity appears first and most clearly at the representational interface.

Conclusion, Philosophical Implications, and Future Work

From both the literature reviewed here and the experiments I conducted, a design-choice explanation for cross-lingual LLM performance differences is more accurate than an intrinsic-language explanation. Experiment 1 demonstrates that the English-Chinese gap at the level of compression is highly tokenizer dependent; the same pair of sentences can be near parity under one tokenizer yet almost two-to-one under another. Experiment 2 translated this compression gap into a concrete opportunity cost, showing that a shared token budget buys substantially fewer few-shot demonstrations in Chinese vs. English. Experiment 3 then provided behavioral evidence that tokenization can shift a model's outputs even after controlling for visible script structure. This suggests that the interface is not purely a compression layer but is partly constitutive of what the model treats as similar. Together, these experiments locate the largest and most actionable axis of English–Chinese disparity at the representational interface rather than in any property of Chinese as a language for machine reasoning.

The primary philosophical implication of this work is that tokenization is not merely pre-processing but can be understood as a theory of language. By committing to a particular vocabulary and merge structure, a tokenizer commits to a particular partition of the input into semantically discoverable units. The Chinese radical result in Experiment 3 makes this concrete: when a tokenizer fragments a Han character into byte pieces, the radical—a meaningful sub-character unit in the script—is no longer a primitive that the model is permitted to see directly, and the model's similarity behavior shifts accordingly. Read this way, claims of the form "Language X is harder for LLMs than Language Y" are under-specified until the tokenizer, and more broadly the engineered interface, is held fixed.

Several limitations bound how strongly these results can be taken. Some of these limitations (like limited number of experiments run) were due to limited compute/time resources for experimentation. While more information on this can be found in this paper's codebase, some obvious limitations include sample size number (e.g. $n = 50$ XNLI queries per language-budget cell in experiment 2), the number of radical pairs used in experiment 3 ($n = 200$), and a single served model being used (gpt-4o-mini) for experiments 2 and 3. Future work could include testing more models, experimenting with additional languages for cross-linguistic result replication (e.g. Japanese, Arabic, Hindi, and Tamil), and extending the radical experiment to additional radicals and tokenizers.

Codebase and Generative AI Use

I formulated the research question, designed the experiments, interpreted the results, and wrote the report. I used Claude through Cursor as a coding assistant for implementation, refactoring, and syntax support, and used GPT for limited LaTeX and figure-formatting assistance. I reviewed and validated the resulting code and outputs. The codebase for this project is available on GitHub linked here.

Appendix A: Supporting Tables and Figures

This appendix contains the full per-tokenizer audit table (including context shrinkage and Wilcoxon statistics), the appendix accuracy figure for Experiment 2, the full fixed-context summary, and the raw logistic-regression coefficients for Experiment 3. These supplement, but do not change, the headline results in the main text.

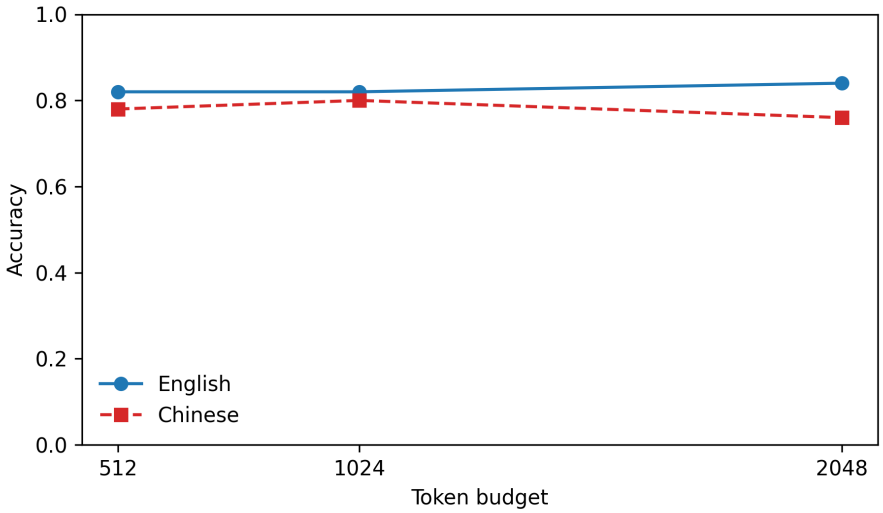

Figure 3: XNLI accuracy at fixed token budgets, by language. Within-row differences are modest and reported for completeness alongside Figure 2.

Table 3: Full per-tokenizer audit including context shrinkage $(1 - 1/\text{TP})$ and Wilcoxon signed-rank p-values testing EN-vs.-ZH token-count equality within each tokenizer.

Tokenizer	Mean TP	95% CI	Context shrinkage	Mean EN	Mean ZH	Wilcoxon p
char	0.33	[0.33, 0.34]	-2.20	130.4	42.8	5.3×10^{-167}
byte	0.90	[0.88, 0.91]	-0.15	130.5	116.0	1.5×10^{-79}
Qwen 2.5	1.01	[0.99, 1.02]	-0.04	27.3	27.4	0.041
o200k_base	1.28	[1.27, 1.30]	+0.19	26.6	33.8	5.8×10^{-140}
Llama 3	1.31	[1.30, 1.33]	+0.21	26.9	35.0	7.4×10^{-149}
cl100k_base	1.89	[1.86, 1.91]	+0.45	26.9	50.3	3.4×10^{-167}

Table 4: Full Experiment 2 (fixed-context) summary across token budgets and languages on XNLI few-shot prompts.

Language	Budget	Demos fit	Accuracy	Cost / correct (USD)	n queries	Tokens in	Tokens out
English	512	11.8	0.820	8.82×10^{-5}	50	23626	118
English	1024	22.0	0.820	0.000174	50	47150	114
English	2048	32.0	0.840	0.000259	50	72000	120
Chinese	512	6.4	0.780	6.57×10^{-5}	50	16392	175
Chinese	1024	17.7	0.800	0.000129	50	33804	173
Chinese	2048	29.5	0.760	0.000270	50	67645	185

Table 5: Full logistic-regression coefficients for Experiment 3.

Term	β (coef)	p value	Regularized
const	-4.595	4.8×10^{-6}	no
same_radical	+3.406	0.001	no
same_token	+1.190	0.080	no

References

[1] Aristotle. *On Interpretation* (*De Interpretatione*).

[2] Petrov, Aleksandar, Emanuele La Malfa, Philip H. S. Torr, and Adel Bibi. "Language Model Tokenizers Introduce Unfairness Between Languages." *arXiv preprint* arXiv:2305.15425, 2023.

[3] Ahia, Orevaoghene, Sachin Kumar, Hila Gonen, Jungo Kasai, David R. Mortensen, Noah A. Smith, and Yulia Tsvetkov. "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*, 9904–9923, 2023.

[4] Rust, Phillip, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. "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)*, 3118–3135, 2021.

[5] Ali, Mehdi, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, et al. "Tokenizer Choice for LLM Training: Negligible or Crucial?" In *Findings of the Association for Computational Linguistics: NAACL 2024*, 3790–3825, 2024.

[6] Grattafiori, Aaron, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, et al. "The Llama 3 Herd of Models." *arXiv preprint* arXiv:2407.21783, 2024.

[7] Limisiewicz, Tomasz, Terra Blevins, Hila Gonen, Orevaoghene Ahia, and Luke Zettlemoyer. "MYTE: Morphology-Driven Byte Encoding for Better and Fairer Multilingual Language Modeling." In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 14926–14940, 2024.

[8] Tejaswi, Atula, Nilesh Gupta, and Eunsol Choi. "Exploring Design Choices for Building Language-Specific LLMs." In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 10493–10518, 2024.

[9] Si, Chenglei, Zhengyan Zhang, Yingfa Chen, Fanchao Qi, Xiaozhi Wang, Zhiyuan Liu, Yasheng Wang, Qun Liu, and Maosong Sun. "Sub-Character Tokenization for Chinese Pretrained Language Models." *arXiv preprint* arXiv:2106.00400, 2021.

[10] Sun, Zijun, Xiaoya Li, Xiaofei Sun, Yuxian Meng, Xiang Ao, Qing He, Fei Wu, and Jiwei Li. "ChineseBERT: Chinese Pretraining Enhanced by Glyph and Pinyin Information." *arXiv preprint* arXiv:2106.16038, 2021.