From Bias to Balance: How Multilingual Dataset Composition Affects Tokenizer Performance and Computational Equity

Anonymous Author(s)
Affiliation
Address
email

Abstract

Tokenization serves as a crucial preprocessing step in multilingual language models, affecting performance in both high-resource and low-resource languages. However, current tokenizers seem to adopt language biases due to unbalanced training datasets, leading to a poorly optimized tokenizer for underrepresented languages. This research examines the impact of balanced multilingual datasets on the performance of tokenizers trained with the Byte Pair Encoding, WordPiece, and Unigram Language Model algorithms. We build balanced corpora from various sources to study the impact of vocabulary size on 15k, 30k, 50k dataset scales. The trained tokenizers are assessed through intrinsic metrics, including Subword Fertility and Normalized Sequence Length, as well as through extrinsic performance on downstream tasks like Part-of-Speech tagging, Named Entity Recognition, and Machine Translation. We build custom data sets along with customized evaluation pipelines to enable consistent comparisons across nine languages using models built into standard NLP frameworks. Our observations reinforce the importance of a balanced dataset when training tokenizers and, in turn, advance the development of equitable and robust multilingual NLP systems.

1 Introduction

Tokenization serves as the critical bridge between raw text and model input in NLP, particularly challenging in multilingual settings where vocabulary overlap is limited Conneau et al. [2020]. Subword tokenization strategies: BPE Sennrich et al. [2016], WordPiece Devlin et al. [2019], and Unigram Kudo [2018], address the out-of-vocabulary problem by segmenting rare words into known units, each optimizing different trade-offs between frequency, coverage, and segmentation granularity. However, existing tokenization strategies disproportionately favor high-resource and Latin-script languages, leading to over-segmentation of low-resource languages and inflating sequence lengths Petrov et al. [2023]. This bias results in up to 68% additional training costs Ali et al. [2023] and creates societal inequalities through higher API costs and slower processing for marginalized communities Rust et al. [2021]. While multilingual models like mBERT Devlin et al. [2018] and XLM-R Conneau et al. [2020] use shared vocabularies, they suffer from token collisions and inconsistent granularity that disadvantage underrepresented languages Xiang Zhang [2024]. This work systematically investigates how balanced multilingual corpora can mitigate tokenizer performance disparities. We analyze nine typologically diverse languages (Yoruba, Arabic, Mandarin Chinese, Russian, Hindi, Japanese, Swahili, Bengali, Turkish) representing different language families, scripts, and morphological complexity. Our evaluation combines intrinsic metrics (subword fertility, normalized sequence length) with downstream performance on POS tagging, NER, and machine translation using balanced Wikipedia and OSCAR datasets.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

We focus on subword tokenizers, excluding character-level and neural approaches. While downstream evaluation uses curated datasets rather than full-scale pretrained models, and data limitations affect some low-resource language tasks, this work provides empirical evidence for balanced data's importance in fair tokenizer design.

2 Methodology

2.1 Tokenizers

We constructed a typologically diverse, balanced corpus from Wikipedia dumps[1] and the OSCAR [2] (Common Crawl) for nine languages: Yoruba, Arabic, Mandarin Chinese, Russian, Hindi, Japanese, Swahili, Bengali, and Turkish, allocating equal characters per language to reduce high-resource bias Zhang et al. [2022]. To study corpus size effects, three datasets of approximately 100M, 200M, and 400M characters were created. All corpora underwent a uniform normalization pipeline consisting of text repair using `ftfy`, Unicode normalization to NFKC, removal of non-printable characters, and whitespace standardization, with only non-empty cleaned lines retained. Using Hugging Face and SentencePiece, we trained BPE, WordPiece, and Unigram tokenizers with vocabulary sizes of 15k, 30k, and 50k; BPE and WordPiece applied whitespace pre-tokenization, while Unigram operated on raw text to better handle non-whitespace scripts. Special tokens ([PAD], [UNK], [CLS], [SEP], [MASK] or equivalents) and full character coverage were enforced, enabling direct comparison across algorithms and vocabulary sizes. Tokenizer performance was assessed using Normalized Sequence Length (NSL), the average tokens per character indicating segmentation granularity, and Subword Fertility, the average tokens per whitespace-delimited word reflecting subword splits. Evaluation was conducted on balanced samples of 50 sentences per language from TATOEBA[3] (Yoruba, Bengali) and TED2020 [4] (others), with results summarized in Table 3.

2.2 Downstream Tasks

POS Tagging We compiled a balanced dataset for nine languages from Universal Dependencies (Arabic, Mandarin Chinese, Russian, Hindi, Japanese, Turkish), MasakhaPOS (Yoruba, Swahili), and the NLTK Indian corpus (Bengali). Sentences with aligned tokens and POS tags were retained, reformatted into a unified JSONL structure (`tokens`, `tags`, `lang`), shuffled, and saved in UTF-8. A BERT-based token classification model (bert-base-cased) was fine-tuned with each tokenizer configuration. Tokenization was word-aligned, with subwords inheriting parent word labels and non-aligned tokens masked. Training used batch size 16, learning rate 5e-5, and 3 epochs.

Named Entity Recognition (NER) For NER, Yoruba data came from MasakhaneNER (CoNLL), and the remaining eight languages from WikiANN. All datasets were normalized to a shared schema (`tokens`, `ner_tags`, `language`), with Yoruba tags mapped to the WikiANN label set. Each language was downsampled to balance representation, shuffled, and split 80/20. A BERT-based token classification model was fine-tuned with all tokenizer variants. Subword alignment followed POS procedures, and class weights mitigated label imbalance. Evaluation used seqeval metrics: precision, recall, F1, and accuracy.

Machine Translation Parallel corpora were sourced from OPUS100 and TED2020, with TED2020 filling low-resource gaps (e.g., Swahili). Preprocessing removed duplicates, noisy strings, extreme-length sentences, and number/punctuation-heavy lines. Datasets were balanced across languages, shuffled, and split 90/10. We trained multilingual BART-large models with different tokenizer configurations. Training used Seq2SeqTrainer for 5 epochs (batch size 8, gradient accumulation 8, lr 1e-4, warmup 10%, weight decay 0.01, FP16, label smoothing 0.1, gradient clipping 1.0). Inputs were truncated/padded to 256 tokens; generation used beam search (beam=2, max length=128). Evaluation employed BLEU and exact match accuracy.

[1] `https://dumps.wikimedia.org/`
[2] `https://oscar-project.org/`
[3] `https://tatoeba.org/`
[4] `https://opus.nlpl.eu/TED2020.php`

3 Experimental Results

3.1 Intrinsic Tokenizer Evaluation

As shown in Appendix Table 3, larger vocabularies consistently reduce both metrics, with the strongest effect in logographic languages like Mandarin Chinese and Japanese Kudo and Richardson [2018], Conneau et al. [2020]. Across tokenizers, BPE yields the most compact sequences, WordPiece the least (especially at smaller vocabularies), and Unigram lies in between Sennrich et al. [2016], Wu et al. [2016], Kudo and Richardson [2018]. Typology also matters: morphologically complex languages benefit more from larger vocabularies, while simpler languages like Yoruba and Swahili show little change Bostrom and Durrett [2020], Wang et al. [2020]. Overall, larger vocabularies improve efficiency across algorithms, reducing sequence length and fragmentation and thus lowering computational cost in downstream tasks Qiu et al. [2020].

3.2 POS Tagging Results

We further compare tokenizers on POS tagging across vocabulary sizes using accuracy and F1 metrics. From Table 1, it can be inferred that wordPiece achieves the highest overall performance, with a test accuracy of 0.7830 and weighted F1 of 0.7722 at 15k, consistently outperforming BPE and SentencePiece Unigram. This can be attributed to WordPiece's ability to preserve morphologically meaningful units, which benefits POS tagging where syntactic boundaries are crucial Straka et al. [2016]. In contrast, BPE prioritizes frequency-based merges, often splitting or merging across morpheme boundaries, which reduces efficiency for this task despite shorter sequences Sennrich et al. [2016], Kudo [2018]. SentencePiece Unigram shows intermediate behavior, offering slightly higher macro-F1 than BPE but lacking the stability of WordPiece Kudo and Richardson [2018]. Notably, increasing vocabulary size does not improve results and in some cases reduces accuracy, as larger vocabularies can overspecialize subword units and lose the generalization capacity needed for POS tagging Nivre et al. [2016], Kann and Schütze [2016], Mielke et al. [2021].

Table 1: POS Performance comparison across tokenizers, vocabulary sizes, and metrics. Best values per block are highlighted in bold.

Tokenizer	Voc size	Epoch	Train Acc	Train F1 Macro	Train F1 Weighted	Test Acc	Test F1 Macro	Test F1 Weighted
BPE		3	0.7847	0.2957	0.7722	0.7207	0.2973	0.7057
WordPiece	15k	3	**0.8413**	**0.3889**	**0.8325**	**0.7830**	**0.3952**	**0.7722**
SentencePiece Unigram		3	0.8067	0.3209	0.7947	0.7484	0.3299	0.7340
BPE		3	0.7526	0.2686	0.7366	0.6932	0.2724	0.6735
WordPiece	30k	3	**0.7964**	0.2877	**0.7839**	**0.7325**	0.2915	**0.7174**
SentencePiece Unigram		3	0.7752	**0.3141**	0.7625	0.7211	**0.3218**	0.7061
BPE		3	0.7643	0.2719	0.7484	0.7015	0.2775	0.6812
WordPiece	50k	3	**0.8001**	0.2848	**0.7880**	**0.7335**	0.2883	**0.7187**
SentencePiece Unigram		3	0.7662	**0.3085**	0.7528	0.7107	**0.3141**	0.6944

3.3 NER Results

Table 2 presents the NER performance across tokenizers, vocabulary sizes, and evaluation metrics. With 15k vocabulary size, WordPiece significantly outperforms BPE and SentencePiece Unigram, achieving the highest Test F1 (0.5844) and Test Accuracy (0.7644). This suggests that WordPiece is particularly effective in low-vocabulary regimes, where its ability to balance word-level and subword-level information aids entity boundary recognition Devlin et al. [2019], Wu et al. [2016], Li et al. [2019]. As vocabulary size increases to 30k and 50k, BPE shows competitive performance, especially in Test Accuracy (0.7032 at 50k), while SentencePiece occasionally surpasses BPE in terms of F1 score. However, WordPiece maintains overall superiority, albeit with diminishing margins, likely due to reduced fragmentation and more stable subword segmentation as vocabulary size grows Klein et al. [2017]. Overall, these results indicate that WordPiece offers the best generalization for NER at smaller vocabulary sizes, while BPE and SentencePiece Unigram become more competitive at larger vocabularies, reflecting the trade-off between segmentation granularity and contextual representation in sequence labeling tasks Peters et al. [2018], Akbik et al. [2018], Mielke et al. [2021].

Table 2: NER Performance comparison across tokenizers, vocabulary sizes, and metrics

Tokenizer	Voc size	Epoch	Train Precision	Train Recall	Train F1	Train Accuracy	Test Precision	Test Recall	Test F1	Test Accuracy
BPE	15k	3	0.3113	0.6453	0.4200	0.6878	0.2452	0.5088	0.3309	0.6324
WordPiece		3	**0.5990**	**0.8601**	**0.7062**	**0.8374**	**0.4937**	**0.7157**	**0.5844**	**0.7644**
SentencePiece Unigram		3	0.4949	0.8398	0.6228	0.7856	0.3769	0.6441	0.4756	0.7102
BPE	30k	3	0.3730	0.7452	0.4971	0.7604	0.2640	0.5393	0.3545	0.6828
WordPiece		3	**0.4432**	**0.8106**	**0.5731**	**0.7898**	**0.3251**	**0.6063**	**0.4233**	**0.7014**
SentencePiece Unigram		3	0.4120	0.7810	0.5394	0.7449	0.2920	0.5690	0.3860	0.6532
BPE	50k	3	0.3951	0.7533	0.5183	0.7902	0.2650	0.5290	0.3531	0.7032
WordPiece		3	**0.4266**	0.7842	0.5525	**0.7927**	**0.3034**	**0.5767**	**0.3976**	**0.7153**
SentencePiece Unigram		3	0.4258	**0.7885**	**0.5530**	0.7599	0.2914	0.5515	0.3813	0.6694

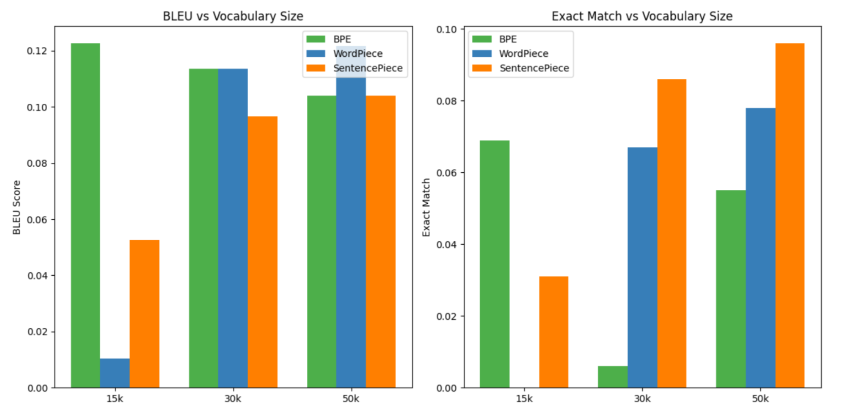

Figure 1: Performance comparison of tokenization methods (BPE, WordPiece, SentencePiece Unigram) across vocabulary sizes (15k, 30k, 50k) using BLEU and Exact Match metrics on multilingual BART-large.

3.4 Machine Translation Results

Figure 1 compares tokenization methods on multilingual BART-large across vocabulary sizes (15k, 30k, 50k) . At 15k, BPE achieved the best BLEU (0.1226), reflecting efficient rare word segmentation, while WordPiece performed poorly (0.0103) due to insufficient coverage of morphologically rich words. At 30k, WordPiece (0.1136 BLEU) nearly matched BPE (0.1135) and aligned better with reference lengths, while SentencePiece Unigram achieved the highest Exact Match (0.086) but produced shorter sequences. At 50k, WordPiece led in BLEU (0.1218) and Unigram in Exact Match (0.096), showing a trade-off between fluency and precision. Overall, BPE is strongest at smaller vocabularies, WordPiece scales better with larger ones, and Unigram favors exact matching but tends to under-generate. These trends highlight how subword granularity, vocabulary coverage, and sequence length jointly shape BLEU and Exact Match performance.

4 Discussion and Conclusion

This study investigated the impact of balanced multilingual datasets on tokenizer performance across nine typologically diverse languages, showing that balanced data improves both efficiency and fairness. BPE consistently yielded the lowest Normalized Sequence Length and Subword Fertility, while logographic languages like Chinese and Japanese benefited most from larger vocabularies (e.g., NSL for Chinese dropped from 0.85 to 0.70 between 15k and 50k). Downstream performance was task-dependent: WordPiece excelled in POS tagging (accuracy 0.7830) and NER (F1 0.5844), while BPE led in machine translation at smaller vocabularies (BLEU 0.1226). Balanced datasets reduced over-segmentation in low-resource languages, mitigating computational inequities where underrepresented languages can otherwise incur up to 68% extra processing costsAli et al. [2023]. While our results are promising, several limitations must be acknowledged. Future work should extend evaluation to character-level and neural tokenization methods, explore family-specific balancing, and assess generative tasks where tokenizer choice may influence quality differently. Multi-tokenizer approaches within a single model also offer promise for leveraging complementary strategies. Overall, balanced training data is critical for fair and efficient multilingual tokenization. While optimal choice remains task-dependent, balanced datasets consistently enhance performance, advancing more equitable multilingual NLP systems that move beyond one-size-fits-all paradigms.

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, 2020. doi:10.18653/v1/2020.acl-main.747.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016. doi:10.18653/v1/p16-1162.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019. doi:10.18653/v1/n19-1423.

Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018. doi:10.18653/v1/p18-1007.

Aleksandar Petrov, La Malfa Emanuele, Philip H. S. Torr, and Adel Bibi. Language model tokenizers introduce unfairness between languages, 2023.

Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, Jasper Ebert, Niclas Doll, Jan Stanislaus Buschhoff, et al. Tokenizer choice for llm training: negligible or crucial?, 2023.

Phillip Rust, 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*, pages 3118–3135, 2021. doi:10.18653/v1/2021.acl-long.243.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

Chenyu You Xiang Zhang, Juntai Cao. Counting ability of large language models and impact of tokenization, 2024. URL `https://arxiv.org/html/2410.19730v2`. arXiv:2410.19730v2.

Shiyue Zhang, Vishrav Chaudhary, Naman Goyal, James Cross, Guillaume Wenzek, Mohit Bansal, and Francisco Guzmán. How robust is neural machine translation to language imbalance in multilingual tokenizer training? In *Proceedings of the 15th biennial conference of the Association for Machine Translation in the Americas*, 2022. URL `https://aclanthology.org/2022.amta-research.8/`.

Taku Kudo and John Richardson. Sentencepiece: a simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 66–71, 2018. doi:10.18653/v1/d18-2012.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*, 2016.

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

Changhan Wang, Kyunghyun Cho, and Jiatao Gu. Neural machine translation with byte-level subwords. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9154–9160, 2020.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10): 1872–1897, 2020.

Milan Straka, Jan Hajic, and Jana Strakova. Udpipe: trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 4290–4297, 2016.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, et al. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, pages 1659–1666, 2016.

Katharina Kann and Hinrich Schütze. Single-model encoder-decoder with explicit morphological representation for reinflection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 555–560, 2016.

Sabrina J Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y Lee, Benoît Sagot, et al. Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*, pages 5647–5664, 2021.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. A unified mrc framework for named entity recognition. *arXiv preprint arXiv:1910.11476*, 2019.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M Rush. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, 2017.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237, 2018.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, 2018.

A Computational Resources

All experiments in this study were conducted using two distinct computational environments. Tokenizer training for all nine languages across three vocabulary sizes (15k, 30k, 50k) and three algorithms (BPE, WordPiece, Unigram), along with Part-of-Speech tagging and Named Entity Recognition model training, were performed using Google Colab with Tesla T4 GPU (16GB VRAM) and Python 3.10 with CUDA 11.8. Machine Translation experiments using multilingual BART-large required enhanced computational capacity and were conducted on a local workstation equipped with NVIDIA GeForce RTX 4090 (24GB VRAM) and CUDA 11.8. The choice of computational environments was determined by the memory requirements of each task, with the larger BART-large models for machine translation necessitating the higher VRAM capacity of the RTX 4090 GPU.

Table 3: NSL and Subword Fertility across tokenizers, languages, and vocabulary sizes.

Language	Tokenizer	15k voc size		30k voc size		50k voc size	
		NSL	Subword Fertility	NSL	Subword Fertility	NSL	Subword Fertility
Yoruba	BPE	0.4584	1.9911	0.3993	1.7278	0.3668	1.5783
	WordPiece	0.8137	3.5557	0.4584	1.9867	0.3827	1.6498
	SentencePiece Unigram	0.5471	2.3723	0.4178	1.8040	0.3963	1.7086
Arabic	BPE	0.4908	2.8232	0.3726	2.1427	0.3316	1.9056
	WordPiece	0.8353	4.8084	0.4485	2.5773	0.3567	2.0482
	SentencePiece Unigram	0.4859	2.7878	0.3879	2.2264	0.3427	1.9706
Mandarin Chinese	BPE	0.8479	8.7353	0.7438	7.6769	0.6985	7.2173
	WordPiece	0.8391	8.4954	0.7628	7.7758	0.6786	6.9250
	SentencePiece Unigram	0.9393	9.6524	0.8556	8.7974	0.8166	8.4003
Russian	BPE	0.4621	3.3589	0.3367	2.4322	0.2990	2.1634
	WordPiece	0.8701	6.3412	0.4406	3.1862	0.3276	2.3668
	SentencePiece Unigram	0.4879	3.5411	0.3414	2.4549	0.3091	2.2264
Hindi	BPE	0.4727	2.5893	0.3566	1.9619	0.3258	1.8028
	WordPiece	0.7793	4.1668	0.4210	2.2823	0.3377	1.8444
	SentencePiece Unigram	0.5174	2.8608	0.3819	2.1243	0.3497	1.9463
Japanese	BPE	0.7739	11.0098	0.5978	8.4776	0.5290	7.4856
	WordPiece	0.9168	13.1019	0.7082	10.0753	0.5775	8.1898
	SentencePiece Unigram	0.8368	11.8092	0.6681	9.3931	0.5973	8.3685
Swahili	BPE	0.4008	2.6293	0.2953	1.9350	0.2620	1.7163
	WordPiece	0.8573	5.6392	0.3759	2.4664	0.2910	1.9076
	SentencePiece Unigram	0.4068	2.6739	0.3031	1.9849	0.2632	1.7273
Bengali	BPE	0.4270	2.9301	0.2897	1.9807	0.2479	1.6934
	WordPiece	0.8569	5.8917	0.3897	2.6693	0.2759	1.8865
	SentencePiece Unigram	0.4621	3.1692	0.2958	2.0243	0.2512	1.7167
Turkish	BPE	0.4295	3.3339	0.3166	2.4487	0.2807	2.1667
	WordPiece	0.8805	6.8531	0.3937	3.0524	0.3050	2.3543
	SentencePiece Unigram	0.4482	3.4741	0.3262	2.5227	0.2826	2.1694

Table 4: Performance comparison of different tokenization methods on multilingual BART-large. Best scores per vocabulary size are in **bold**.

Voc Size	Tokenizer Type	BLEU	Exact Match	Avg. Pred Len	Avg. Label Len
15k	BPE	**0.1226**	**0.069**	27.41	23.05
	WordPiece	0.0103	0	20	35.11
	SentencePiece Unigram	0.0526	0.031	9.14	7.37
30k	BPE	0.1135	0.006	17.99	18.55
	WordPiece	**0.1136**	0.067	23.30	21.65
	SentencePiece Unigram	0.0966	**0.086**	7.58	7.37
50k	BPE	0.1039	0.055	22.35	16.97
	WordPiece	**0.1218**	0.078	19.94	17.96
	SentencePiece Unigram	0.1039	**0.096**	7.13	7.37

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately reflect the paper's contributions and scope. The abstract clearly states the research examines tokenizer performance using balanced multilingual datasets across BPE, WordPiece, and Unigram algorithms, evaluated on 9 languages with both intrinsic metrics and downstream tasks (POS tagging, NER, machine translation). The introduction properly contextualizes the problem of tokenization bias and clearly delineates the study's scope, including explicit acknowledgment that the work focuses on subword-based tokenizers and excludes full-scale pretrained models.

   Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper explicitly discusses limitations in Section 4, acknowledging that the scope is deliberately focused on subword-based tokenizers (excluding character-level and neural approaches), etc. The authors also outline future research directions to address these limitations, including expanding to character-level methods and exploring intermediate balancing strategies.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA] .

   Justification: This paper is primarily an empirical study that does not include theoretical results, theorems, or mathematical proofs. The work focuses on experimental evaluation of tokenization algorithms across languages using established metrics (NSL and Subword Fertility). Along with the downstream evaluation for these tokenizers.

   Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper details dataset sources and balancing, preprocessing, tokenizer training configurations, and task-specific training setups (hyperparameters, model choices, and evaluation metrics) in Sections 2, with complete result tables in Section 3, enabling independent reproduction of the main findings. See "Dataset Generation," "Data Preprocessing," "Tokenizer training," and the POS/NER/MT training descriptions for exact settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Since this is a paper for double blind, the manuscript does not provides code repository links, thus once review is done, the authors are willing to add the repository link in the code. With that and the detailed processing steps in Sections 2, the results can be replicated.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The paper provides comprehensive experimental details across all tasks. Section 2 thoroughly specifies tokenizer configurations, preprocessing pipelines, training hyperparameters (batch size 16, learning rate 5e-5, 3 epochs for POS; batch size 8 with gradient accumulation for MT), data splits (80/20 for NER, 90/10 for MT), and evaluation metrics. The methodology sections clearly describe how datasets were constructed, balanced, and preprocessed for each downstream task.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We report mean values for NSL, Subword Fertility, and aggregate metrics (accuracy, F1, BLEU, Exact Match) over balanced test sets, which ensures reliable comparisons. However, explicit error bars or statistical significance tests were not included due to computational constraints.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provides detailed computational resource information in Appendix A Computational Resources. It specifies that tokenizer training and POS/NER experiments used Google Colab with Tesla T4 GPU (16GB VRAM) and Python 3.10 with CUDA 11.8, while machine translation experiments required a local workstation with NVIDIA GeForce RTX 4090 (24GB VRAM) and CUDA 11.8. The choice of different environments is justified by memory requirements, with BART-large models necessitating the higher VRAM capacity.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to NeurIPS ethics guidelines by using publicly available datasets, addressing fairness in multilingual NLP (which has positive societal implications), and conducting responsible empirical research without ethical concerns.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses positive societal impacts by addressing computational inequities and linguistic bias that affect underrepresented languages and communities. It acknowledges how tokenization disparities can impose additional costs and processing delays for marginalized communities, contributing to more equitable multilingual NLP systems.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper focuses on tokenization algorithms and evaluation metrics rather than releasing high-risk models or datasets. The work involves standard NLP evaluation tasks and does not pose significant misuse risks requiring special safeguards.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: The paper cites in the footnotes and acknowledges all dataset sources (e.g., Tatoeba, OSCAR, etc.) and tokenizer/modeling baselines in Section 2 and related work. While explicit license types (e.g., CC-BY, MIT) are not always included, references to original sources and their respective documentation ensure compliance with their usage terms.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: The paper does not release new datasets or models as assets; it focuses on evaluating existing tokenization algorithms on curated versions of publicly available datasets.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: The research uses existing publicly available datasets and does not involve crowdsourcing experiments or research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The research does not involve human subjects research, crowdsourcing, or data collection that would require IRB approval, as it uses existing publicly available datasets.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA] .

    Justification: The paper does not describe using LLMs as part of the core methodology; the research focuses on evaluating tokenization algorithms using standard NLP models like BERT and BART, not developing or using LLMs in novel ways.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.