Tokenization Disparities as Infrastructure Bias: How Subword Systems Create Inequities in LLM Access and Efficiency

1st Hailay Kidu Teklehaymanot
L3S Research Center
Leibniz University Hannover
Hannover, Germany
teklehaymanot@L3S.de

2nd Wolfgang Nejdl
L3S Research Center
Leibniz University Hannover
Hannover, Germany
nejdl@L3S.de

***Abstract*—Tokenization disparities pose a significant barrier to achieving equitable access to artificial intelligence across linguistically diverse populations. This study conducts a large-scale cross-linguistic evaluation of tokenization efficiency in over 200 languages to systematically quantify computational inequities in large language models (LLMs). Using a standardized experimental framework, we applied consistent preprocessing and normalization protocols, followed by uniform tokenization through the `tiktoken` library across all language samples. Comprehensive tokenization statistics were collected using established evaluation metrics, including Tokens Per Sentence (TPS) and Relative Tokenization Cost (RTC), benchmarked against English baselines. Our cross-linguistic analysis reveals substantial and systematic disparities: Latin-script languages consistently exhibit higher tokenization efficiency, while non-Latin and morphologically complex languages incur significantly greater token inflation, often 3–5 times higher RTC ratios. These inefficiencies translate into increased computational costs and reduced effective context utilization for underrepresented languages. Overall, the findings highlight structural inequities in current AI systems, where speakers of low-resource and non-Latin languages face disproportionate computational disadvantages. Future research should prioritize the development of linguistically informed tokenization strategies and adaptive vocabulary construction methods that incorporate typological diversity, ensuring more inclusive and computationally equitable multilingual AI systems.**

***Index Terms*—Multilingual Models, Tokenization, Infrastructure Bias, Subword System, Large Language Models**

I. Introduction

Recent advances in large language models (LLMs) have transformed natural language processing [1], [2], yet these developments remain disproportionately concentrated on high-resource languages, particularly English, leaving the majority of the world's languages underrepresented in both research and technological deployment. While multilingual LLMs (mLLMs) theoretically enable cross-lingual knowledge transfer from high-resource to low-resource languages through shared linguistic representations [3], substantial performance disparities persist across linguistically diverse populations [4]. Recent analyses of code-switching datasets [5] reveal systemic biases toward English and a lack of sociolinguistic representativeness in data collection and annotation, reflecting broader disparities in multilingual modeling. The global NLP research landscape exhibits a significant bias toward high-resource languages, resulting in substantial performance gaps for underrepresented languages due to limited annotated data and inadequate computational support [6], [7]. As highlighted by [5], the lack of representativeness in English-dominant and unbalanced datasets reflects broader structural inequities in multilingual NLP. These disparities parallel the biases embedded in tokenization systems, underscoring how insufficiently representative data undermines fairness and inclusivity across language technologies. Despite advances in transfer learning and transformer architectures that partially address these disparities through cross-lingual knowledge transfer [8], the extent and limitations of multilingual generalization remain under investigation.

A critical but underexplored factor contributing to these disparities is tokenization, the fundamental preprocessing step that transforms raw text into subword units for model input. Tokenization algorithms, predominantly optimized for high-resource languages, may inadequately handle the morphological complexity and structural properties of low-resource languages, thereby exacerbating existing inequalities [9]. This study presents a systematic investigation of tokenization disparities across over 200 languages using the FLORES-200 benchmark. By analyzing state-of-the-art tokenizers, we quantify variations in cross-lingual efficiency and identify tokenization as a key driver of performance inequities in multilingual NLP. Our findings lay the groundwork for developing more equitable tokenization strategies that enhance fairness and inclusivity across diverse linguistic contexts.

II. Related Work

A. Background

Tokenization constitutes a fundamental preprocessing step in natural language processing systems [10], [11], transforming raw text into discrete units for model consumption. Contemporary multilingual language models predominantly

employ subword tokenization strategies to address vocabulary constraints while maintaining representational efficiency across linguistically diverse corpora.

Subword Tokenization

Subword tokenization decomposes words into semantically meaningful subunits, effectively addressing the open vocabulary problem in neural language models [12]. This approach enables efficient processing of both frequent and rare lexical items by representing common words directly while decomposing infrequent terms into familiar subword components [13]. Operating at an intermediate granularity between character-level and word-level representations, subword methods capture morphological regularities particularly beneficial for morphologically rich languages [9]. Predominant subword algorithms include Byte Pair Encoding (BPE) [14], which iteratively merges frequent symbol pairs; WordPiece [15], employing probabilistic merging strategies; and SentencePiece [16], treating input as raw Unicode sequences without explicit word boundaries. While transformer-based multilingual models rely extensively on these approaches, vocabulary construction often reflects high-resource language dominance, potentially introducing systematic biases for underrepresented languages [11].

Tokenization Efficiency Disparities

Recent investigations have revealed substantial tokenization disparities across languages, with significant implications for computational equity. [17] demonstrated efficiency variations up to 13-fold between English and other languages across 108 languages, highlighting how existing tokenization schemes systematically favor high-resource languages. [18] and [19] further documented tokenization length variations reaching 15-fold for semantically equivalent texts, particularly affecting non-Latin scripts and morphologically complex languages. These disparities carry practical consequences beyond performance metrics, directly impacting computational costs and accessibility. Token-based pricing models in commercial language services result in disproportionately higher usage costs for speakers of underrepresented languages, creating economic barriers to AI access. Proposed solutions include multilingually fair subword algorithms [19] and adaptive gradient-based approaches such as MAGNET [20], designed to minimize over-segmentation while accommodating diverse morphological and orthographic characteristics. Multilingual Vocabulary Allocation Multilingual models face vocabulary allocation challenges, distributing limited vocabulary capacity across multiple languages [21]. Inductive biases encoded in tokenizers reflect training corpus distributions, often favoring high-resource languages and resulting in suboptimal representation quality for underrepresented scripts [22], [23]. Poor vocabulary allocation leads to excessive subword fragmentation, particularly detrimental for word-level tasks requiring accurate morphological segmentation [13].

III. Methodology

A. Dataset

This study employs the FLORES-200 dataset [24], providing standardized parallel text across 200 languages representing diverse linguistic families, scripts, and morphological structures. We utilize the devtest split (1,012 sentences per language) to ensure cross-linguistic consistency and eliminate potential data contamination effects.

B. Tokenization Model

We employ the cl100k [1] base tokenizer via the tiktoken library, representing production-grade Byte Pair Encoding implementations deployed in state-of-the-art large language models, including GPT-3.5 and GPT-4. While computationally efficient, this tokenizer's training data exhibits high-resource language bias, potentially introducing systematic inefficiencies for underrepresented languages.

IV. Experimental Procedure

For this study, tokenization experiments are conducted using the tiktoken library, which provides access to OpenAI's production-grade Byte Pair Encoding (BPE) tokenizers. The tiktoken library implements highly efficient tokenization routines specifically designed for transformer-based language models, including those deployed in OpenAIs GPT-3.5 and GPT-4 architectures. In particular, we employ the cl100kbase encoding, a subword vocabulary widely adopted across OpenAI models that supports a broad range of languages and scripts, though it is primarily optimized for high-resource languages. The experimental procedure consists of the following steps:

a) Preprocessing and Normalization: All text samples from the FLORES-200 devtest set are first normalized using Unicode Normalization Form C (NFC). This normalization step ensures consistency in character representation across languages, which is particularly critical for scripts that allow multiple valid Unicode encodings of the same grapheme cluster. Such normalization is essential to eliminate artifacts that could distort tokenization behavior, especially in non-Latin and morphologically complex languages.

b) Tokenization with tiktoken: Following normalization, each sentence is tokenized using the cl100kbase tokenizer provided by tiktoken[2]. This tokenizer employs a data-driven subword segmentation strategy optimized for compression and computational efficiency. However, due to its construction based primarily on high-resource language corpora, it may introduce biases when applied to underrepresented languages that diverge significantly in morphological or orthographic structure.

[1]OpenAI's tiktoken
[2]OpenAI's tiktoken

c) Tokenization Statistics Extraction: After tokenization, we compute and record several key statistics for each sentence, including: The total number of tokens generated. The total number of characters processed. Per language aggregated metrics such as: Tokens Per Sentence (TPS): the average number of tokens per sentence. Characters Per Token (CPT): the average number of characters represented by each token.

d) Cross-Linguistic Efficiency Analysis: To quantify cross-linguistic disparities, we calculate the Relative Tokenization Cost (RTC) for each language. This metric compares each language's TPS value to that of English, which serves as the reference baseline. The RTC provides a normalized indicator of tokenization efficiency, allowing us to evaluate how much more or less efficiently each language is tokenized relative to English. This experimental setup enables a systematic, language-agnostic evaluation of tokenization efficiency across all 200 FLORES languages. By employing a consistent preprocessing pipeline and a single tokenizer configuration, we aim to isolate and quantify the extent of tokenization inefficiencies that may arise from applying widely used subword algorithms to linguistically diverse languages in state-of-the-art multilingual models.

V. Tokenizer Evaluation Metrics

To systematically assess tokenization efficiency across diverse languages, we adopt several quantitative metrics that capture both absolute and relative characteristics of tokenized outputs. These metrics allow for cross-linguistic comparisons that highlight disparities introduced by subword segmentation strategies in multilingual language models.

A. Tokens Per Sentence (TPS)

The **Tokens Per Sentence (TPS)** metric computes the average number of tokens generated per sentence after tokenization for each language. Formally, for a language L with N sentences:

$$TPS(L) = \frac{1}{N} \sum_{i=1}^{N} T_i$$

where T_i denotes the number of tokens in sentence i. This metric reflects the granularity of subword segmentation and provides a language-specific view of tokenization density.

B. Characters Per Token (CPT)

The **Characters Per Token (CPT)** metric evaluates the average number of characters represented by each token:

$$CPT(L) = \frac{\sum_{i=1}^{N} C_i}{\sum_{i=1}^{N} T_i}$$

where C_i is the character count of sentence i, and T_i is its corresponding token count. CPT indicates how efficiently the tokenizer compresses character sequences into tokens. Lower CPT values typically reflect finer-grained segmentations, while higher values may suggest more compact tokenization.

C. Relative Tokenization Cost (RTC)

To directly quantify cross-linguistic disparities, we define the **Relative Tokenization Cost (RTC)** using English as the baseline reference language. For any language L, RTC is computed as:

$$RTC(L) = \frac{TPS(L)}{TPS(English)}$$

An RTC value greater than 1 indicates that language L requires more tokens to represent an equivalent sentence compared to English, signaling potential inefficiencies in tokenization for that language. Conversely, an RTC value less than 1 suggests that language L is tokenized more compactly than English.

D. Aggregate Efficiency Comparison

We conduct aggregate efficiency comparisons across the entire FLORES-200 [25] language set to quantify global disparities and assess the extent to which low-resource languages are disproportionately affected. This comprehensive analysis facilitates the identification of systematic biases in existing subword tokenization schemes and quantifies linguistic inclusivity across languages with varying resource availability, orthographic systems, and morphological complexity.

VI. Results and Evaluation

Our comprehensive evaluation encompasses 200+ languages representing diverse script families, including Latin, Arabic, Devanagari, Ethiopic, Cyrillic, and numerous others. The multilingual corpus exhibits substantial imbalance, with Latin-script languages contributing the largest proportion of content, followed by Arabic-script, Devanagari, and Ethiopic language families.

Script Distribution Analysis

Figure 1 illustrates the distribution of languages across script families, revealing significant concentration in Latin-based writing systems. This imbalance reflects historical and contemporary patterns of digital language representation, with implications for tokenization algorithm development and deployment. Scripts employed by single languages including Armenian (Armn), Georgian (Geor), and Japanese (Jpan) represent particularly vulnerable cases where tokenization inefficiencies may disproportionately impact linguistic accessibility.

Cross-Linguistic Tokenization Efficiency

Our empirical analysis reveals substantial disparities in tokenization efficiency across languages and script families, as demonstrated through multiple complementary analytical perspectives.

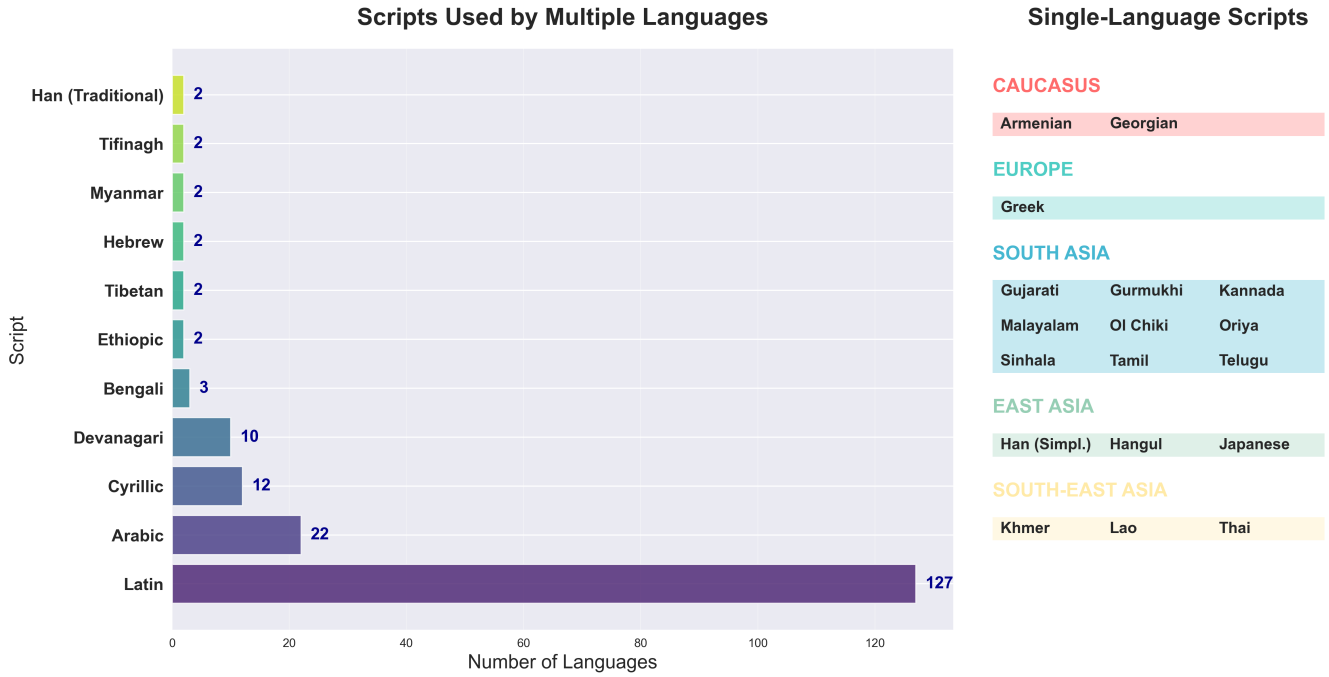

Fig. 1. Comparison of scripts used by multiple languages. The bar chart shows the number of languages using each script, with Latin (Latn) being the most widely adopted. An annotation box lists scripts used by only a single language, such as Armenian (Armn), Georgian (Geor), and Japanese (Jpan). This figure highlights the distribution and concentration of script usage across languages.

Tokens Per Sentence (TPS) Analysis

Figure 2 quantifies the extreme range of tokenization burdens across writing systems. Myanmar script requires the highest token density (357.2 TPS), followed by Ol Chiki (342.1 TPS) and Oriya (334.0 TPS), representing nearly 7-fold higher tokenization costs compared to the most efficient scripts. Latin script achieves optimal efficiency (50.2 TPS), with Han Traditional (56.1 TPS) and Han Simplified (56.8 TPS) also demonstrating relatively compact tokenization. The language-wide mean of 90.3 TPS serves as a reference point, revealing that numerous scripts require 3-4 times more tokens than this average, creating substantial computational inequalities.

TPS measurements demonstrate significant variation across languages, with morphologically complex languages exhibiting substantially higher token densities. Languages employing agglutinative morphological processes require extensive subword fragmentation, resulting in elevated TPS values that directly impact computational resource requirements and context window utilization.

Characters Per Token (CPT) Patterns

Figure 3 illustrates dramatic script-dependent variations in compression efficiency. Latin script achieves the highest compression efficiency (2.61 CPT), followed by Cyrillic (1.58 CPT) and Greek (1.14 CPT). Non-Latin scripts consistently demonstrate lower efficiency: Arabic (1.28 CPT), Devanagari (0.99 CPT), and numerous Asian scripts falling below 1.0 CPT. Notably, Tibetan (0.49 CPT), Oriya (0.40 CPT), and Ol Chiki (0.41 CPT) exhibit severe tokenization inefficiencies, indicating excessive subword fragmentation for these writing systems.

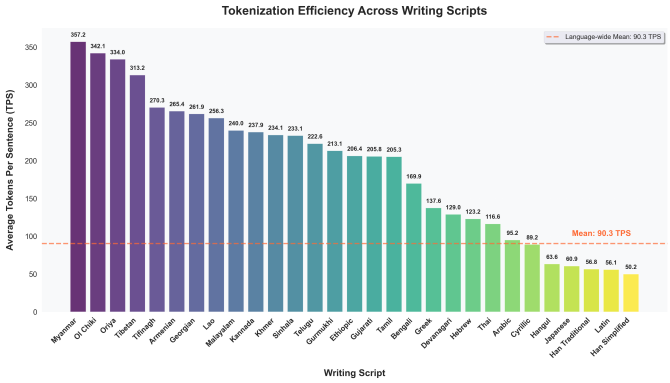

Fig. 2. Tokenization Efficiency Across Writing Scripts (Tokens Per Sentence - TPS). Cross-script comparison quantifies extreme tokenization burden variations, with the Myanmar script requiring the highest token density (357.2 TPS) and the Latin script achieving optimal efficiency (50.2 TPS). The language-wide mean of 90.3 TPS serves as a reference, revealing that numerous scripts require 3-4 times more tokens than average, with some scripts demanding nearly 7-fold higher computational costs for equivalent semantic content.

Characters Per Token (CPT) Analysis by Language Family

Figure 4 demonstrates substantial variation in CPT values across language families, ranging from 0.55 (Kannada) to 2.85 (Creole languages). Creole languages exhibit the highest compression efficiency (2.85 CPT), followed by Isolate languages

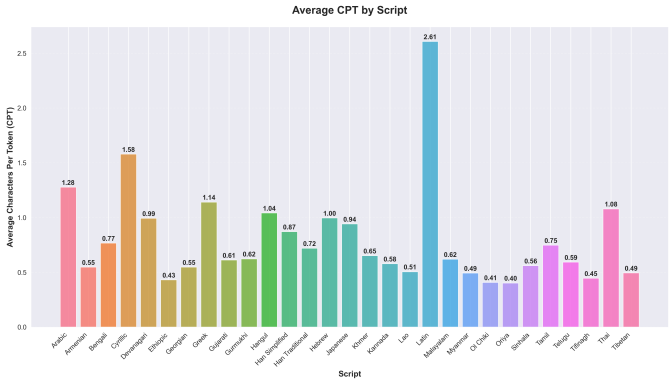

Fig. 3. Average Characters Per Token (CPT) by Writing Script. Script-based analysis reveals dramatic efficiency variations, with Latin script achieving optimal compression (2.61 CPT) compared to numerous Asian scripts falling below 1.0 CPT. Non-Latin scripts consistently demonstrate lower efficiency, with Tibetan (0.49 CPT), Oriya (0.40 CPT), and Ol Chiki (0.41 CPT) exhibiting severe tokenization inefficiencies. This pattern reflects the Latin-script optimization inherent in contemporary tokenization algorithms.

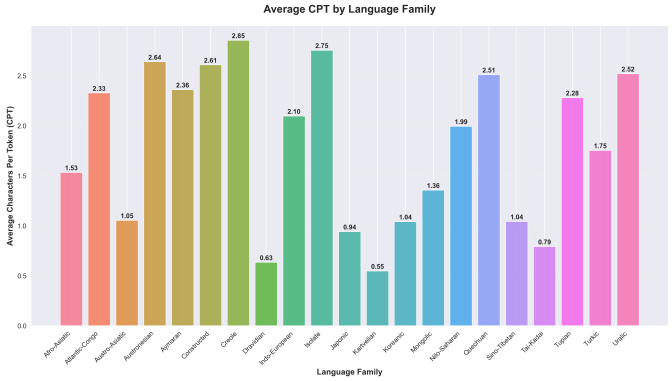

Fig. 4. Average Characters Per Token (CPT) by Language Family. The analysis reveals substantial variation across language families, with Creole languages demonstrating the highest tokenization efficiency (2.85 CPT) and Kannada exhibiting the lowest efficiency (0.55 CPT). Higher CPT values indicate more efficient character-to-token compression, while lower values reflect excessive subword fragmentation. Notable patterns include the superior performance of Creole, Isolate, and Austronesian families compared to morphologically complex families such as Dravidian and Tai-Kadai.

(2.75 CPT) and Austronesian languages (2.64 CPT). Conversely, Kannada (0.55 CPT), Dravidian (0.63 CPT), and Tai-Kadai (0.79 CPT) demonstrate substantially lower compression ratios, indicating excessive subword fragmentation that severely impacts computational efficiency for these language families.

Relative Tokenization Cost (RTC) Disparities

RTC measurements quantify the magnitude of cross-linguistic tokenization inequalities relative to English baseline performance. Our results demonstrate RTC values ranging from below 1.0 for closely related Indo-European languages to values exceeding 4.0 for morphologically rich and non-Latin script languages. These disparities translate directly into differential computational costs and accessibility barriers for speakers of underrepresented languages, with some languages requiring up to four times the computational resources for equivalent semantic processing.

Aggregate Efficiency Comparison (AEC)

The aggregate analysis confirms systematic tokenization biases favoring high-resource, Latin-script languages. Low-resource languages, particularly those employing unique scripts or complex morphological systems, consistently demonstrate reduced efficiency across all evaluation metrics. This pattern reflects the training data composition of contemporary tokenizers and highlights fundamental accessibility challenges in current multilingual AI systems.

VII. Impact on Model Performance and Resource Utilization

These tokenization disparities carry significant practical implications for multilingual language model deployment:

a) Computational Resource Inequality: Languages with elevated RTC values require disproportionate computational resources for equivalent semantic processing, creating systematic disadvantages for underrepresented language communities.

b) Context Window Limitations: Higher token densities reduce effective context utilization for complex languages, potentially degrading model performance on tasks requiring extended contextual understanding.

c) Economic Accessibility Barriers: Token-based pricing models in commercial language services amplify these disparities, resulting in substantially higher usage costs for speakers of inefficiently tokenized languages.

d) Performance Degradation: Excessive subword fragmentation may impair semantic representation quality, particularly for word-level tasks requiring accurate morphological analysis.

These findings underscore the critical need for developing language-aware tokenization strategies that address systematic inequalities in multilingual AI system accessibility and performance across linguistically diverse populations.

VIII. Discussions

This study frames its findings as evidence of both **systematic algorithmic** and **infrastructural biases** rather than mere technical limitations. It reveals how existing tokenization algorithms inherently favor high-resource languages, while infrastructural disparities in data availability and tool support further exacerbate the gap. The analysis **connects these technical disparities** to **broader issues of social and linguistic equity**, emphasizing how such design choices can reinforce **economic and accessibility barriers**. Overall, the study **underscores the failure of current multilingual systems** to achieve **inclusive and equitable language representation**.

IX. Conclusions

Summarizes key findings: 7-fold efficiency disparities Connects technical inequalities to real-world barriers Calls for language-aware tokenization strategies Emphasizes need for equity over efficiency in multilingual AI.

LIMITATIONS

Acknowledges single tokenizer/dataset constraints Notes focus on efficiency rather than downstream performance recognizes potential oversimplification within script families Provides clear directions for future research.

ACKNOWLEDGMENT

This research was supported by the German Academic Exchange Service (DAAD) through the Hilde Domin Programme (funding no. 57615863). The authors gratefully acknowledge this support.

REFERENCES

[1] J. Armengol-Estapé, C. P. Carrino, C. Rodriguez-Penagos, O. de Gibert Bonet, C. Armentano-Oller, A. Gonzalez-Agirre, M. Melero, and M. Villegas, "Are multilingual models the best choice for moderately under-resourced languages? a comprehensive assessment for catalan," *arXiv preprint arXiv:2107.07903*, 2021.

[2] A. Muñoz Ortiz, "Dependency parsing as sequence labeling for low-resource languages," Universidad Nacional de Educación a Distancia (España), Escuela Técnica Superior de Ingenieros en Informática, Technical Report, 2021.

[3] Y. Xu, L. Hu, J. Zhao, Z. Qiu, Y. Ye, and H. Gu, "A survey on multilingual large language models: Corpora, alignment, and bias," *arXiv preprint arXiv:2404.00929*, 2024.

[4] S. Mutuvi, E. Boros, A. Doucet, G. Lejeune, A. Jatowt, and M. Odeo, "Analyzing the impact of tokenization on multilingual epidemic surveillance in low-resource languages," in *Document Analysis and Recognition - ICDAR 2023*, G. A. Fink, R. Jain, K. Kise, and R. Zanibbi, Eds. Cham: Springer Nature Switzerland, 2023, pp. 17–32.

[5] A. S. Doğruöz, S. Sitaram, and Z. X. Yong, "Representativeness as a forgotten lesson for multilingual and code-switched data collection and preparation," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5751–5767. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.382/

[6] A. Magueresse, V. Carles, and E. Heetderks, "Low-resource languages: A review of past work and future challenges," *arXiv preprint arXiv:2006.07264*, 2020.

[7] Y. Zhu, B. Heinzerling, I. Vulić, M. Strube, R. Reichart, and A. Korhonen, "On the importance of subword information for morphological tasks in truly low-resource languages," *arXiv preprint arXiv:1909.12375*, 2019.

[8] C. Theodoropoulos and M.-F. Moens, "An information extraction study: Take in mind the tokenization!" in *Conference of the European Society for Fuzzy Logic and Technology*. Springer, 2023, pp. 593–606.

[9] S. J. Mielke, Z. Alyafeai, E. Salesky, C. Raffel, M. Dey, M. Gallé, A. Raja, C. Si, W. Y. Lee, B. Sagot *et al.*, "Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp," *arXiv preprint arXiv:2112.10508*, 2021.

[10] M. Hassler and G. Fliedl, "Text preparation through extended tokenization," *WIT Transactions on Information and Communication Technologies*, vol. 37, 2006.

[11] C. Toraman, E. H. Yilmaz, F. Şahinuç, and O. Ozcelik, "Impact of tokenization on language models: An analysis for turkish," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 4, pp. 1–21, 2023.

[12] C. Park, Y. Yang, K. Park, and H. Lim, "Decoding strategies for improving low-resource machine translation," *Electronics*, vol. 9, no. 10, p. 1562, 2020.

[13] T. Limisiewicz, J. Balhar, and D. Mareček, "Tokenization impacts multilingual language modeling: Assessing vocabulary allocation and overlap across languages," *arXiv preprint arXiv:2305.17179*, 2023.

[14] R. Sennrich, B. Haddow, and A. Birch, "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)*, K. Erk and N. A. Smith, Eds. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725. [Online]. Available: https://aclanthology.org/P16-1162

[15] X. Song, A. Salcianu, Y. Song, D. Dopson, and D. Zhou, "Fast wordpiece tokenization," *arXiv preprint arXiv:2012.15524*, 2020.

[16] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," *arXiv preprint arXiv:1808.06226*, 2018.

[17] M. Asprovska and N. Hunter, "The tokenization problem: Understanding generative ai's computational language bias," *Ubiquity Proceedings*, vol. 4, no. 1, 2024.

[18] O. Ahia, S. Kumar, H. Gonen, J. Kasai, D. Mortensen, N. Smith, and Y. 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*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 9904–9923. [Online]. Available: https://aclanthology.org/2023.emnlp-main.614/

[19] A. Petrov, E. La Malfa, P. Torr, and A. Bibi, "Language model tokenizers introduce unfairness between languages," *Advances in neural information processing systems*, vol. 36, pp. 36 963–36 990, 2023.

[20] O. Ahia, S. Kumar, H. Gonen, V. Hoffman, T. Limisiewicz, Y. Tsvetkov, and N. A. Smith, "Magnet: Improving the multilingual fairness of language models with adaptive gradient-based tokenization," *arXiv preprint arXiv:2407.08818*, 2024.

[21] B. Zheng, L. Dong, S. Huang, S. Singhal, W. Che, T. Liu, X. Song, and F. Wei, "Allocating large vocabulary capacity for cross-lingual language model pre-training," *arXiv preprint arXiv:2109.07306*, 2021.

[22] P. Vyas, A. Kuznetsova, and D. S. Williamson, "Optimally encoding inductive biases into the transformer improves end-to-end speech translation." in *Interspeech*, 2021, pp. 2287–2291.

[23] F. Remy, P. Delobelle, H. Avetisyan, A. Khabibullina, M. de Lhoneux, and T. Demeester, "Trans-tokenization and cross-lingual vocabulary transfers: Language adaptation of llms for low-resource nlp," *arXiv preprint arXiv:2408.04303*, 2024.

[24] NLLB Team, M. R. Costa-jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard, A. Sun, S. Wang, G. Wenzek, A. Youngblood, B. Akula, L. Barrault, G. M. Gonzalez, P. Hansanti, J. Hoffman, S. Jarrett, K. R. Sadagopan, D. Rowe, S. Spruit, C. Tran, P. Andrews, N. F. Ayan, S. Bhosale, S. Edunov, A. Fan, C. Gao, V. Goswami, F. Guzmán, P. Koehn, A. Mourachko, C. Ropers, S. Saleem, H. Schwenk, and J. Wang, "Scaling neural machine translation to 200 languages," *Nature*, vol. 630, no. 8018, pp. 841–846, 2024. [Online]. Available: https://doi.org/10.1038/s41586-024-07335-x

[25] M. R. Costa-Jussà, J. Cross, O. Çelebi, M. Elbayad, K. Heafield, K. Heffernan, E. Kalbassi, J. Lam, D. Licht, J. Maillard *et al.*, "No language left behind: Scaling human-centered machine translation," *arXiv preprint arXiv:2207.04672*, 2022.