

What Do We Talk About When We Talk About Tokenization?

Gianluca Vico and Jindřich Libovický

Charles University, Faculty of Mathematics and Physics,
Institute of Formal and Applied Linguistics
Malostranské náměstí 25, 118 00 Praha, Czech Republic
{vico, libovicky}@ufal.mff.cuni.cz

Abstract

Tokenization appears across a large and heterogeneous body of NLP literature, most often as a design choice rather than the primary focus. We identify more than 800 such papers published over the last decade and use an LLM-based methodology to survey them at a scale that would not be feasible through manual review. We analyze four dimensions: the languages studied, the evaluation tasks, the intrinsic metrics used, and the granularity of the tokenization units considered. We find that machine translation is the most frequently studied topic when tokenization is discussed. Despite the multilingual focus of much of this work, a strong English and Eurocentric bias persists. Subword tokenization dominates in practice despite sustained research interest in characters, bytes, and morphemes. We release the extracted dataset to support further analysis¹.

1 Introduction

Tokenization is one of many design choices when building a natural language processing (NLP) system, including large language models (LLMs). Recent work has studied a range of tokenization-specific problems: the so-called *token premium* problem (Ahia et al., 2023), which leads to substantially different LLM inference costs across languages; tokenization of code and numerical data (Wallace et al., 2019; Rozière et al., 2023); improvements to existing tokenization algorithms (Chizhov et al., 2024; Bauwens and Delobelle, 2024; Cognetta et al., 2024); and proposals for alternative tokenization approaches (Libovický and Helcl, 2024; Schmidt et al., 2024a).

We identify more than 800 papers discussing tokenization published over the last decade and use an LLM-based methodology for large-scale evidence

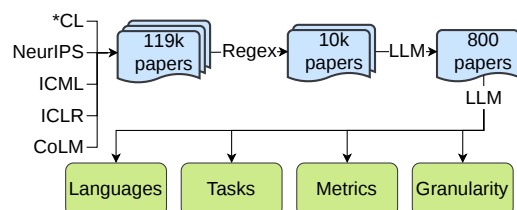


Figure 1: Pipeline used to filter the papers and extract information.

extraction to synthesize them into an empirical survey of how tokenization research has evolved over time.

We make three contributions. First, we use an *LLM-based survey methodology* that can process a body of literature at a scale that would not be feasible through manual review. We provide a replicable framework and quantify its reliability through manual annotation, finding 98% sensitivity and 92% specificity on a test sample. Second, we apply this methodology to produce *empirical findings on trends in tokenization research*, including the languages studied, the linguistic units considered, the evaluation tasks used, and the distribution of work across venues, as summarized in Table 1. Third, we *release the resulting dataset* to support further analysis by the community.

Our main findings are as follows: the corpus shows a strong English and Eurocentric bias. Extrinsic evaluation dominates, with machine translation and question answering as the most common tasks. Subword tokenization remains the dominant unit in the literature and, overwhelmingly, in practice, despite sustained research interest in characters, bytes, and morphemes. The remainder of the paper presents this methodology (Section 2), followed by our empirical findings (Sections 4–5).

¹<https://github.com/GianlucaVico/TalkAboutTokenization>

2 Methods

Large-scale literature analysis using LLMs has already been used several times as an alternative to manual reviewing. For example, in the biomedical domain, LLMs have been used to automate paper screening and data extraction in systematic reviews, achieving high accuracy, though with caveats regarding reliability (Schmidt et al., 2024b). Similar pipelines have been applied within NLP to study field-wide trends, including research on LLM limitations (Kostikova et al., 2025) and model interpretability (Calderon and Reichart, 2025). We adopt a similar approach and apply it to tokenization research, processing over 100,000 papers from a decade of NLP publications (see Appendix D for the sources).

Our pipeline has four stages: we first retrieve papers from relevant venues; we apply regular-expression matching to abstracts to obtain an initial candidate set; we then use an LLM to refine this selection; finally, we prompt the LLM with targeted questions about each paper and aggregate the responses. The pipeline is summarized in Figure 1.

We focus on text tokenization in NLP, though adjacent domains are occasionally relevant. We therefore collect papers from ACL Anthology², NeurIPS, ICLR, ICML, and CoLM conferences and their co-located workshops, covering 2016–2025³, i.e., from the year BPE was introduced in NLP (Sennrich et al., 2016). This yields 119k papers in total, the majority from *CL venues; the breakdown by venue is shown in Table 1.

Given the volume of papers, we do an initial filtering. We apply a regular expression (token|sub-?word|segment|sub-?tok) to titles and abstracts, under the assumption that papers discussing tokenization mention related terms already in the abstract. We then pass the filtered set through the LLM to remove incorrectly selected papers and discard papers with missing or empty PDFs. We use GPT-OSS 120B⁴, an open-weight instruction-tuned model and the largest we could run at this scale. The model is prompted to recognize tokenization papers based on their title and abstract.

²Note that ACL Anthology collects not only proceedings of conferences organized by ACL (such as ACL, NAACL, EACL, AACL, EMNLP, INLG), journals (TACL, CL), and workshops, but also proceedings of many events not organized by ACL (such as LREC, COLING, AMTA).

³We extend the time span to 2014–2025 when discussing tokenization units (see Section 5)

⁴From HuggingFace: openai/gpt-oss-120b

Venue	Papers	Regex	LLM
*CL	74,668	6,469	732
NeurIPS	24,606	2,044	32
ICML	11,049	942	56
ICLR	8,471	925	19
CoLM	781	166	7
Total	119,575	10,546	846

Table 1: Selected papers from 2016 to 2025 from each conference. *Regex* is the first filter used to reduce the total number of papers. *LLM* removes papers with invalid PDFs and refines the selection to the final dataset.

After an initial inspection of the selected papers, we refine the prompt to focus on tokenizer-free methods, as this topic was incorrectly discarded. We keep papers selected by either prompt. After this combined filtering, we reduce the set of papers from 119,575 to 846.

The LLM filtering step is sensitive to prompt formulation: certain topics, such as language modeling, may be over-represented in the selected set, while others, such as tokenizer-free methods, may be under-represented despite being relevant. We quantify this by manually annotating a random sample of 100 papers from each of the selected and discarded sets. We find 1 false negative and 8 false positives. This indicates that the pipeline reliably identifies tokenization papers, but also retains a non-trivial number of unrelated ones, which introduce noise into our analysis. Upon inspecting the false positives, we find that they mention tokenization only in passing, not as a central concern. Note that estimating recall directly would require annotating a large random sample of all 119,575 papers; we therefore follow the practice common in medical research and report recall ($98.95\% \pm 2.10\%$, 95% confidence interval with bootstrap) and specificity ($92.46 \pm 4.94\%$) instead. The specificity means that about 1 in 12 selected papers may not actually discuss tokenization in depth. Since these papers mention tokenization only briefly and likely make mainstream design choices, we expect their impact on our results to be small.

After selecting the papers, we convert each PDF to Markdown, remove sections such as the references and the appendix, and prompt the LLM to extract four types of information: the languages discussed, the evaluation tasks, the intrinsic metrics used, and the segmentation granularity adopted.

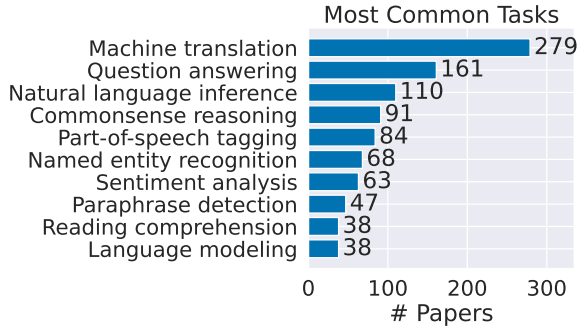


Figure 2: Machine translation and question answering are the two most popular tasks, but papers can discuss multiple tasks. Less frequent tasks are not shown.

The exact prompts used for filtering and extraction are given in Appendix B. We explicitly allow the LLM to generate a negative answer, e.g., if the paper does not use any intrinsic metric. We use 10 papers as a validation set to estimate reliability for each category.

3 Tasks & Metrics

Tokenizer quality can be measured intrinsically (e.g., by properties of the segmentation itself) or extrinsically (i.e., by performance on a downstream task). We prompt the LLM to list the downstream tasks used to evaluate the tokenizer and the intrinsic metrics. Overall, 25.3% of the papers have both intrinsic and extrinsic evaluation; 38.6% have only extrinsic; 25.6% have only intrinsic; and for 10.5% we did not retrieve any evaluation. We then group similar entries to account for spelling variations and closely related tasks or metrics (e.g., *NER* and *Named-entity recognition* are merged, as are *Rényi efficiency* and *Rényi entropy*). The LLM often answers with the datasets used for the evaluation instead of the downstream task: in these cases, we select the datasets that are mentioned more than 5 times and group them with the tasks. In Appendix E–F, we provide more detail on the grouping.

Figure 2 shows the most frequent tasks. Machine translation is the most popular task, followed by question answering, natural language inference, and commonsense reasoning. However, many datasets for these tasks can be used for question answering as well.

It might appear surprising that POS tagging and NER are among the most frequent evaluation tasks. While these tasks are considered largely solved for high-resource languages such as English (Manning,

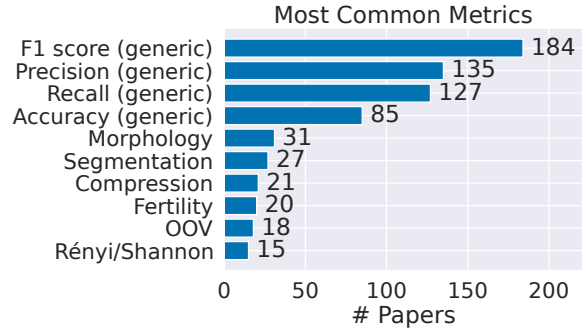


Figure 3: The frequency of the most commonly used intrinsic metrics. Metrics marked with *(generic)* can refer to multiple task. Some metrics are grouped together when they are measuring similar things (e.g., morphological boundary recall and precision).

2011; Strubell et al., 2017), they still occur frequently for several reasons: they are simple to set up and evaluate, have been widely used in interpretability studies (Belinkov, 2022), and remain challenging for low-resource languages. Since our corpus covers the last ten years, it is not surprising to see them. However, as tokenization research moves toward LLM-focused settings, we expect these tasks to become less common in future work.

Figure 3 shows the most frequent intrinsic metrics. F1, precision, recall and accuracy are the four most widely used metrics overall, as they apply across a wide range of tasks and are straightforward to compute. Then, we see metrics specific to tokenization. In particular, metrics evaluating morphology or, more in general, segmentation. Other notable metrics include compression metrics (like fertility (Rust et al., 2021)), out-of-vocabulary rate and distribution metrics such as Rényi efficiency (Zouhar et al., 2023).

From the validation set, we notice that three papers mention datasets instead of tasks, and one paper over-generalized a task. For the intrinsic metric, the LLM fails to recognize the metrics in two cases; in one case, it reports the right acronym but the wrong name; and in another case, the metrics are incorrect.

4 Languages

Multilinguality is a central concern in tokenization research, as already suggested by the popularity of translation in the previous section. At the same time, the NLP field is known to be biased towards English (Schwartz, 2022). We extract the languages mentioned in each paper, without

distinguishing whether a language is the primary focus or simply used for evaluation. We also count programming languages as languages but do not attribute any geographical or typological features to them. Sometimes the language is not mentioned explicitly, but the model is prompted to mention it if it can be understood from the paper or to say that it is not specified.

Figure 4 shows that most papers focus on a single language, though this varies across venues. ML-oriented venues (NeurIPS, ICML, ICLR) tend to focus on fewer languages, while *CL venues tend to be more multilingual (with workshops specializing in the processing of less-resourced language families). Still, even at *CL venues, most papers study only one language. A small number of papers mention no language at all, which may reflect extraction errors, a tokenization from a theoretical point of view, or work on non-linguistic data such as protein sequences. The median number of languages is 2 and remains stable over the years.

We align each language with Glottolog (Hammarström et al., 2025) to obtain language-family and geographical information. Languages not found by exact name matching are aligned manually. Dialects or creoles not listed in Glottolog are mapped to their parent language (e.g., *Singaporean English* is treated as *English*). When geographical information is missing, we fall back to the parent or grandparent language in the Glottolog hierarchy.

Glottolog provides six macroareas: Eurasia, Africa, North America, South America, Papunesia, and Australia. Since we want to distinguish European from Asian languages, we split the Eurasia macroarea accordingly. This is not straightforward for transcontinental countries, so we consider a language European if it is spoken in at least one European country (see Appendix C for the list of countries). As a result, languages spoken in the easternmost part of Russia are classified as European as well. For dialects missing geographical information, we use the parent language (e.g., Mexican Spanish is treated as Spanish and classified as European). The full list of European countries is in Appendix C.

Some papers refer to writing systems rather than specific languages, particularly for Chinese languages. In such cases, we still assign geographical information where possible. Similarly, when papers mention unspecified Indian or African languages without naming them, we retain only the geographical information.

Figure 5 shows that research is concentrated on European languages, followed by Asian ones. We did not retrieve any papers covering Australian languages. Regions like Papunesia receive little attention despite their language diversity. The languages of Russia have little influence regardless of their continental classification, as this accounts for just 12 papers. Figure 6 further shows that even papers covering non-European languages almost always include at least one European language, likely due to the near-universal inclusion of English.

Figure 7 shows a similar pattern at the level of language families: Indo-European languages dominate, followed by Sino-Tibetan. This is consistent with the broader English bias in NLP documented by Schwartz (2022).

Figure 8 shows that English dominates overwhelmingly, appearing in 536 papers. English appears in more than twice as many papers as Chinese (234), the second-most-frequent language. German (210) ranks third, substantially ahead of French (144) and Japanese (141); this likely reflects both the strong German NLP research community and the historical centrality of German in machine translation benchmarks (Kocmi et al., 2025, *inter alia*). The dominance of national and official languages confirms the field’s bias, with minority and low-resource languages largely absent from the most frequent ones.

Out of the 10 papers inspected, the LLM correctly identifies the languages of 7 of them; in one case, the languages are only briefly mentioned in the appendix and the LLM hallucinates the languages, although some are correct; in another paper, some languages are added; in one case, languages are added because the paper contains specific characters, even though it is not experimenting with them.

5 The Basic Unit

Tokenization methods differ in the granularity at which they split text. Common methods such as BPE (Sennrich et al., 2016) and UnigramLM (Kudo, 2018) operate at the subword level, while others such as ByT5 (Xue et al., 2022) and Canine (Clark et al., 2022) use bytes or characters directly. We prompt the LLM to extract the units discussed in each paper, focusing on *bytes*, *characters*, *subwords*, *words*, *morphemes*, *patches*, or *pixels*, but with the option of adding other units if relevant. The LLM returns a list of units per paper, which

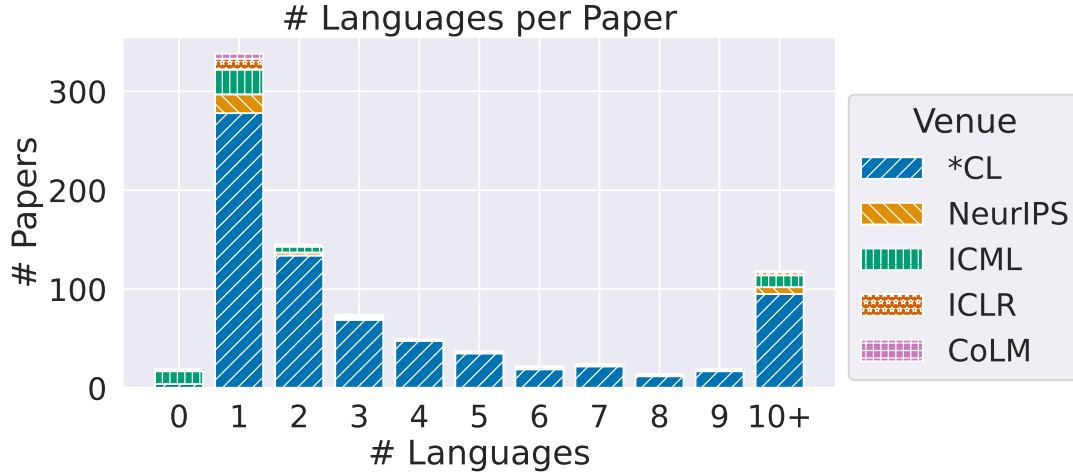


Figure 4: Number of languages per paper. The co-located workshops are grouped together with the main conference. Using one language is the most frequent setting. Papers with zero languages are, for example, theoretical papers without experiments on natural languages.

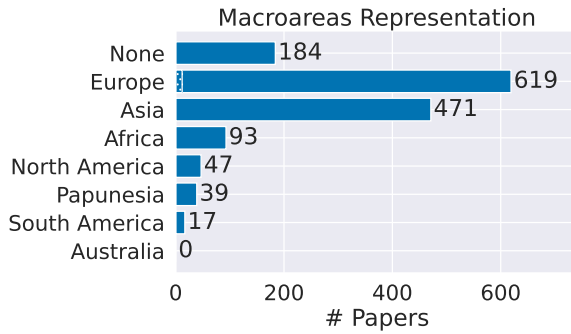


Figure 5: Frequency of each continent/macroarea. The dotted bar represents Russia, which spans part of Asia. *None* refers to cases where we could not retrieve this information or programming languages.

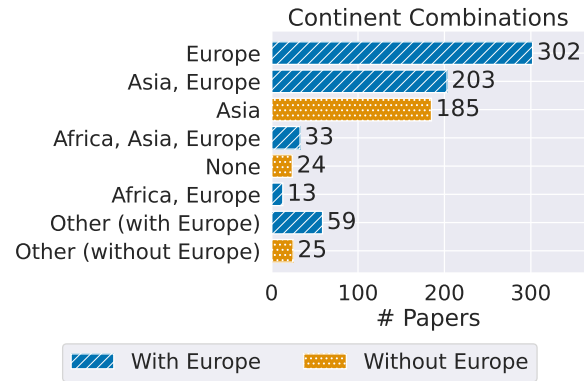


Figure 6: Sets of macroareas considered by the papers. During grouping, *None* is ignored unless it is the sole macroarea in the set. Europe is present in most of the works.

we group into broader categories (see Appendix G).

Figure 9 shows that subwords are the most common unit, followed by characters and words. Byte-based models are less frequent than character-based ones, and morpheme-based models represent a small niche. Word-based models are less used. The term *token* is ambiguous since it can refer to different units depending on the context. Two categories are harder to track reliably: pixel-based models such as PIXEL (Rust et al., 2023), which the LLM may confuse with other vision-language models, and “tokenizer-free” or latent tokenization methods, where non-standard terminology (e.g., *chunk* or *patch*) can be confused with unrelated uses of the same terms or with papers focusing on the model architecture.

Figure 10 shows that most papers discuss one unit. Papers discussing multiple units typically

either compare several tokenization methods or reflect extraction noise: the LLM sometimes attributes *character* and *byte* units to BPE-based methods, since BPE builds its vocabulary from individual characters, rather than labeling them as *subword* methods.

Figure 11 shows how the use of different units has changed over time. Here, we include papers from 2014 to see how introducing BPE changes the unit used. Subwords increased in popularity in 2019 following the introduction of UnigramLM (ULM) (Kudo, 2018). Words have steadily declined in favor of other methods. Characters rose in popularity after 2016, likely as a baseline for comparison with subword methods, but declined again. Morpheme-based models seem to be less popular in general, except for 2015 when most of

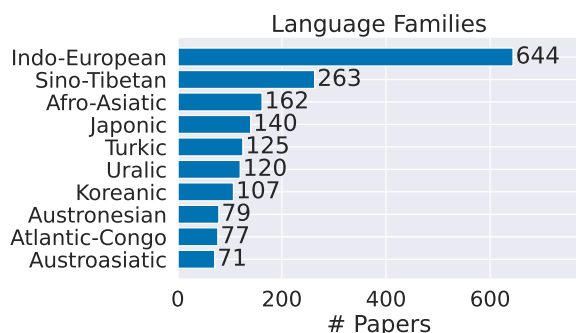


Figure 7: The most common language families included in the papers.

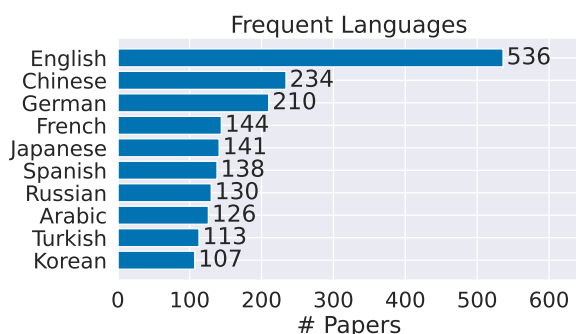


Figure 8: The most common languages used in the papers. *Chinese* is used to refer to Mandarin, Chinese writing systems, and unspecified Sino-Tibetan languages that use these scripts.

the papers retrieved used this method. Bytes are less common but show a gradual increase following the introduction of ByT5.

More broadly, by 2023–2025, all non-subword units are converging to similarly low prevalence levels, suggesting the field is consolidating around subword tokenization to an even greater degree than frequency counts alone imply. The rise-and-fall arc of characters, peaking around 2016–2019, aligns with the interest in character-level modeling (Kim et al., 2016; Sennrich, 2017; Lee et al., 2017; Cherry et al., 2018) that preceded the widespread adoption of BERT-style pretraining (Devlin et al., 2019).

To complement this analysis, we examine the 150 most popular text generation models on HuggingFace, 140 of which have accessible tokenizers. Of these, 130 use BPE, 9 use UnigramLM, and 1 uses WordPiece (Wu et al., 2016). This suggests that, despite the diversity of units studied in the literature, subword tokenization, especially BPE, has become the dominant choice in practice.

After inspecting the validation set, we note that unit extraction is noisy. In one case, the LLM adds

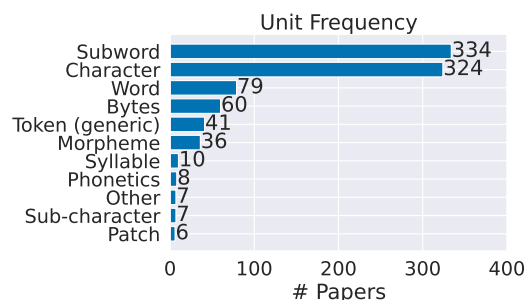


Figure 9: Frequency of the most common type of unit used in the papers. Subwords are the most common choice. *Token* is ambiguous because, depending on the time and context, may refer to different units.

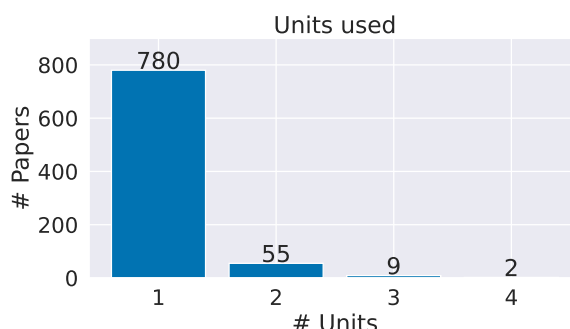


Figure 10: Number of basic units discussed in each paper.

one incorrect unit to the correctly detected unit; in three cases, the LLM fails to identify *subwords* when they are implied by the methods of the paper; in one case, it fails to identify *characters*.

6 Related Work

Surveying tokenization. Mielke et al. (2021) trace the history of open-vocabulary modeling from word-level approaches to subword and character-based methods, concluding that no single tokenization strategy is universally optimal. More recently, Gastaldi et al. (2025) provide a theoretical framework for tokenization, characterizing formal properties such as consistency, ambiguity, and sequentiality that apply across tokenization algorithms. We are not aware of any survey that quantitatively analyzes design decisions for tokenization across a large body of NLP literature.

Automated and LLM-assisted literature reviews. The use of LLMs to assist or automate systematic literature reviews has grown rapidly, particularly in the biomedical domain, where they have been applied to screening and structured data extraction (Schmidt et al., 2024b). A collaborative

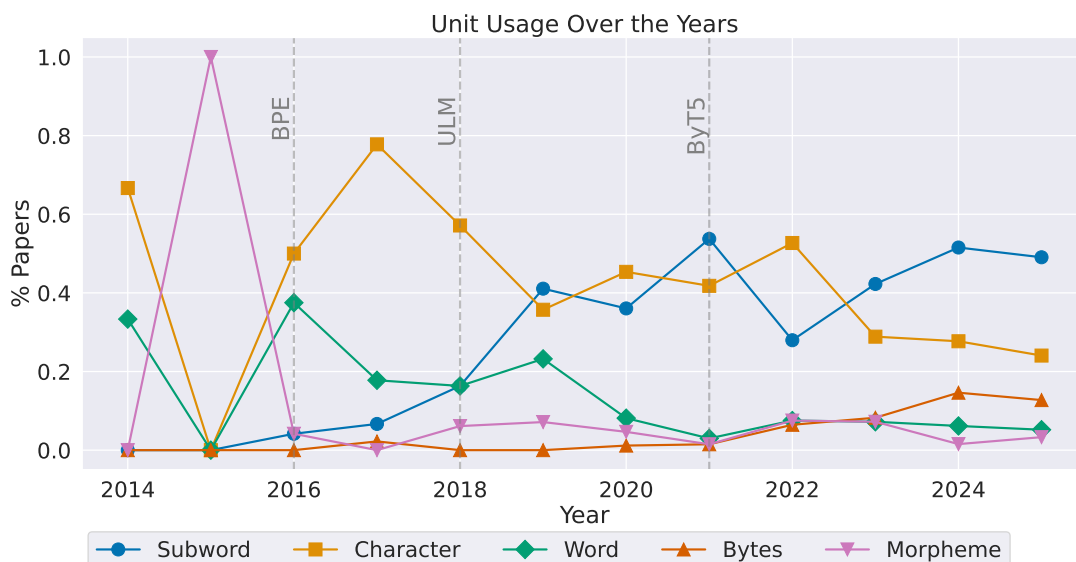


Figure 11: Percentage of papers using different types of units over time. Subwords gain popularity after ULM. Characters gain popularity after BPE, but lose it in the following years. We retrieved a small number of papers from 2015, and those are morpheme-based. Words lose popularity after BPE. Bytes see an increase of popularity after ByT5. We indicate methods (BPE Sennrich et al. (2016); ULM Kudo (2018); ByT5 Xue et al. (2022)) that might have affected the unit used.

two-LLM approach, in which discordant predictions between two models trigger cross-critique, has been proposed to improve reliability (Khan et al., 2025). A scoping review of the field found results to be mixed, with roughly half of the studies finding LLMs promising and a significant minority finding them not yet suitable for unsupervised use (Lieberum et al., 2025) in the biomedical domain. Within NLP, similar pipelines have been used to study field-wide trends: Kostikova et al. (2025) combine keyword filtering with LLM-based classification to survey research on LLM limitations, and Calderon and Reichart (2025) apply a comparable approach to model interpretability research. Our work follows the same methodology, extending it to the field of tokenization research.

7 Conclusions

We employed an LLM-based methodology for large-scale literature analysis and apply it to survey a decade of NLP research on text tokenization. Processing over 100,000 papers, we extract and analyze four dimensions: the languages studied, the evaluation tasks and intrinsic metrics used, and the granularity of the tokenization units considered.

Our analysis reveals that despite apparent multilinguality, the corpus exhibits a strong English and Eurocentric bias, with minority and low-resource languages largely absent (Joshi et al.,

2020; Schwartz, 2022; Kreutzer et al., 2022). English appears in more than twice as many papers as Chinese, the second-most-frequent language. This also highlights a broader tension between high and low-resource languages. While improving NLP for low-resource languages has important linguistic and social value (Magueresse et al., 2020; Bird, 2020), industry incentives favor high-resource languages with larger user bases (Petrov et al., 2023; Nag et al., 2024).

We also observe a gap between research and practice: while the literature explores a wide range of tokenization granularities—including characters, bytes, morphemes, and pixels—an analysis of the 150 most popular text generation models on HuggingFace reveals that BPE subword tokenization accounts for 93% of deployed models. This gap may reflect infrastructural inertia (Alqahtani et al., 2026) rather than inherent superiority: ByT5 (Xue et al., 2022) and PIXEL (Rust et al., 2023) demonstrate competitive performance in many settings, and the consolidation around BPE creates systematic blind spots, particularly for low-resource languages and scripts (Rust et al., 2021) where byte- or character-level representations may be more appropriate.

Our findings suggest several directions the community might prioritize: the development and standardization of intrinsic evaluation benchmarks for

tokenization, given that tokenization-specific metrics such as fertility and Rényi efficiency remain rare; and greater investment in tokenization for non-European scripts where subword methods are least well suited. The dataset we release is intended to support further quantitative analysis of tokenization research and to make future surveys of this kind easier to conduct and verify.

Limitations

The paper selection and the insights are affected by the LLM section, the prompt used, and the sampling method. While we try to mitigate hallucinations, we cannot reliably check every paper. Also, we used only a single model. Moreover, venue selection can bias the results, and we may introduce additional biases when grouping languages, tasks, and metrics.

Ethical Considerations

We do not anticipate any negative ethical implications arising directly from this study.

ACL Anthology has a license CC-BY 4.0⁵. OpenReview metadata has license CC0 1.0⁶, while the papers have license CC-BY 4.0, CC BY-NC 4.0⁷, CC BY-NC-ND 4.0⁸, CC BY-NC-SA 4.0⁹, CC BY-ND 4.0¹⁰, CC BY-SA 4.0¹¹, CC0 1.0¹², or none. These papers are used exclusively for research purposes, and material derived from papers with *No derivatives* clause is removed from the data release.

all-mpnet-base-v2 and gpt-oss-120b have license Apache 2.0¹³.

The paper dataset is not anonymous, as it contains research papers. Given a paper title, abstract, or URL, it is possible to retrieve the authors. However, this information is already public and widely available.

⁵<https://creativecommons.org/licenses/by/4.0/>

⁶<https://creativecommons.org/publicdomain/zero/1.0/>

⁷<https://creativecommons.org/licenses/by-nc/4.0/>

⁸<https://creativecommons.org/licenses/by-nc-nd/4.0/>

⁹<https://creativecommons.org/licenses/by-nc-sa/4.0/>

¹⁰<https://creativecommons.org/licenses/by-nd/4.0/>

¹¹<https://creativecommons.org/licenses/by-sa/4.0/>

¹²<https://creativecommons.org/publicdomain/zero/1.0/>

¹³<https://www.apache.org/licenses/LICENSE-2.0.txt>

We used AI-assisted coding (Copilot), and AI tools (Claude, and Grammarly) for proofreading.

Acknowledgements

We thank Martin Popel and Dušan Variš for comments on the draft of this paper. This research was supported by the Czech Science Foundation project 25-16242S.

References

- Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David Mortensen, Noah Smith, and Yulia Tsvetkov. 2023. [Do all languages cost the same? tokenization in the era of commercial language models](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9904–9923, Singapore. Association for Computational Linguistics.
- Sawsan Alqahtani, Mir Tafseer Nayeem, Md. Tahmid Rahman Laskar, Tasnim Mohiuddin, and M. Saiful Bari. 2026. [Stop taking tokenizers for granted: They are core design decisions in large language models](#). *CoRR*, abs/2601.13260.
- Thomas Bauwens and Pieter Delobelle. 2024. [BPE-knockout: Pruning pre-existing BPE tokenisers with backwards-compatible morphological semi-supervision](#). In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5810–5832, Mexico City, Mexico. Association for Computational Linguistics.
- Yonatan Belinkov. 2022. [Probing classifiers: Promises, shortcomings, and advances](#). *Computational Linguistics*, 48(1):207–219.
- Steven Bird. 2020. [Decolonising speech and language technology](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3504–3519, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Nitay Calderon and Roi Reichart. 2025. [On behalf of the stakeholders: Trends in NLP model interpretability in the era of LLMs](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 656–693, Albuquerque, New Mexico. Association for Computational Linguistics.
- Colin Cherry, George Foster, Ankur Bapna, Orhan Firat, and Wolfgang Macherey. 2018. [Revisiting character-based neural machine translation with capacity and compression](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4295–4305, Brussels, Belgium. Association for Computational Linguistics.

- Pavel Chizhov, Catherine Arnett, Elizaveta Korotkova, and Ivan P. Yamshchikov. 2024. [BPE gets picky: Efficient vocabulary refinement during tokenizer training](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 16587–16604, Miami, Florida, USA. Association for Computational Linguistics.
- Jonathan H. Clark, Dan Garrette, Iulia Turc, and John Wieting. 2022. [Canine: Pre-training an efficient tokenization-free encoder for language representation](#). *Transactions of the Association for Computational Linguistics*, 10:73–91.
- Marco Cognetta, Tatsuya Hiraoka, Rico Sennrich, Yuval Pinter, and Naoaki Okazaki. 2024. [An analysis of BPE vocabulary trimming in neural machine translation](#). In *Proceedings of the Fifth Workshop on Insights from Negative Results in NLP*, pages 48–50, Mexico City, Mexico. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Juan Luis Gastaldi, John Terilla, Luca Malagutti, Brian DuSell, Tim Vieira, and Ryan Cotterell. 2025. [The foundations of tokenization: Statistical and computational concerns](#). In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net.
- Harald Hammarström, Robert Forkel, Martin Haspelmath, and Sebastian Bank. 2025. [glottolog/glottolog: Glottolog database 5.2.1](#).
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. [The state and fate of linguistic diversity and inclusion in the NLP world](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.
- Muhammad Ali Khan, Umair Ayub, Syed Arsalan Ahmed Naqvi, Kaneez Zahra Rubab Khakwani, Zaryab bin Riaz Sipra, Ammad Raina, Sihan Zhou, Huan He, Amir Saeidi, Bashar Hasan, et al. 2025. Collaborative large language models for automated data extraction in living systematic reviews. *Journal of the American Medical Informatics Association*, 32(4):638–647.
- Yoon Kim, Yacine Jernite, David A. Sontag, and Alexander M. Rush. 2016. [Character-aware neural language models](#). In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2741–2749. AAAI Press.
- Tom Kocmi, Ekaterina Artemova, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Konstantin Dranch, Anton Dvorkovich, Sergey Dukanov, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, Barry Haddow, Marzena Karpinska, Philipp Koehn, Howard Lakounga, Jessica Lundin, Christof Monz, Kenton Murray, and 10 others. 2025. [Findings of the WMT25 general machine translation shared task: Time to stop evaluating on easy test sets](#). In *Proceedings of the Tenth Conference on Machine Translation*, pages 355–413, Suzhou, China. Association for Computational Linguistics.
- Aida Kostikova, Zhipin Wang, Deidamea Bajri, Ole Pütz, Benjamin Paaßen, and Steffen Eger. 2025. [Llms: A data-driven survey of evolving research on limitations of large language models](#). *CoRR*, abs/2505.19240.
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, and 33 others. 2022. [Quality at a glance: An audit of web-crawled multilingual datasets](#). *Transactions of the Association for Computational Linguistics*, 10:50–72.
- Taku Kudo. 2018. [Subword regularization: Improving neural network translation models with multiple subword candidates](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.
- Jason Lee, Kyunghyun Cho, and Thomas Hofmann. 2017. [Fully character-level neural machine translation without explicit segmentation](#). *Transactions of the Association for Computational Linguistics*, 5:365–378.
- Jindřich Libovický and Jindřich Helcl. 2024. [Lexically grounded subword segmentation](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7403–7420, Miami, Florida, USA. Association for Computational Linguistics.
- Judith-Lisa Lieberum, Markus Toews, Maria-Inti Metzendorf, Felix Heilmeyer, Waldemar Siemens, Christian Haverkamp, Daniel Böhringer, Joerg J Meerpohl, and Angelika Eisele-Metzger. 2025. Large language models for conducting systematic reviews: on the rise, but not yet ready for use—a scoping review. *Journal of Clinical Epidemiology*, 181:111746.
- Alexandre Magueresse, Vincent Carles, and Evan Heetderks. 2020. [Low-resource languages: A review of past work and future challenges](#). *CoRR*, abs/2006.07264.

- Christopher D. Manning. 2011. [Part-of-speech tagging from 97% to 100%: Is it time for some linguistics?](#) In *Computational Linguistics and Intelligent Text Processing - 12th International Conference, CICLing 2011, Tokyo, Japan, February 20-26, 2011. Proceedings, Part I*, volume 6608 of *Lecture Notes in Computer Science*, pages 171–189. Springer.
- Sabrina J. Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y. Lee, Benoît Sagot, and Samson Tan. 2021. [Between words and characters: A brief history of open-vocabulary modeling and tokenization in NLP](#). *CoRR*, abs/2112.10508.
- Arijit Nag, Animesh Mukherjee, Niloy Ganguly, and Soumen Chakrabarti. 2024. [Cost-performance optimization for processing low-resource language tasks using commercial LLMs](#). In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15681–15701, Miami, Florida, USA. Association for Computational Linguistics.
- Aleksandar Petrov, Emanuele La Malfa, Philip H. S. Torr, and Adel Bibi. 2023. [Language model tokenizers introduce unfairness between languages](#). In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.
- Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton-Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, and 6 others. 2023. [Code llama: Open foundation models for code](#). *CoRR*, abs/2308.12950.
- Phillip Rust, Jonas F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. 2023. [Language modelling with pixels](#). In *The Eleventh International Conference on Learning Representations*.
- Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. [How good is your tokenizer? on the monolingual performance of multilingual language models](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.
- Craig W Schmidt, Varshini Reddy, Haoran Zhang, Alec Alameddine, Omri Uzan, Yuval Pinter, and Chris Tanner. 2024a. [Tokenization is more than compression](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 678–702, Miami, Florida, USA. Association for Computational Linguistics.
- Lena Schmidt, Kaitlyn Hair, Sergio Graziozi, Fiona Campbell, Claudia Kapp, Alireza Khantemoori, Dawn Craig, Mark Engelbert, and James Thomas. 2024b. [Exploring the use of a large language model for data extraction in systematic reviews: A rapid feasibility study](#). In *Proceedings of the 3rd Workshop on Augmented Intelligence for Technology-Assisted Reviews Systems (ALTARS 2024) co-located with the 46th European Conference on Information Retrieval (ECIR 2024), Glasgow, Scotland, March 28, 2024 (hybrid event)*, volume 3832 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- Lane Schwartz. 2022. [Primum Non Nocere: Before working with Indigenous data, the ACL must confront ongoing colonialism](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 724–731, Dublin, Ireland. Association for Computational Linguistics.
- Rico Sennrich. 2017. [How grammatical is character-level neural machine translation? assessing MT quality with contrastive translation pairs](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 376–382, Valencia, Spain. Association for Computational Linguistics.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 1715–1725.
- Emma Strubell, Patrick Verga, David Belanger, and Andrew McCallum. 2017. [Fast and accurate entity recognition with iterated dilated convolutions](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2670–2680, Copenhagen, Denmark. Association for Computational Linguistics.
- Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. [Do NLP models know numbers? probing numeracy in embeddings](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5307–5315, Hong Kong, China. Association for Computational Linguistics.
- Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, and 12 others. 2016. [Google’s neural machine translation system: Bridging the gap between human and machine translation](#). *Preprint*, arXiv:1609.08144.
- Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2022. [ByT5: Towards a token-free](#)

future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306.

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

A LLM Hyper-parameters

For filtering, we use the model through HuggingFace’s transformers text-generation pipeline. We needed up to eight MI210 AMD GPUs with 64GB of memory each.

- Model: openai/gpt-oss-120b
- dtype: auto (bfloat16 and uint8)
- do_sample: False
- max_new_tokens: 1024

For extracting information, we use the same model locally hosted, with the following decoding parameters:

- temperature: 0.8
- top-k: 40
- top-p: 0.9

B Prompts

These are the prompts used for filtering the papers and extracting information. The placeholders TITLE, ABSTRACT, and CONTENT refer to the specific paper we are processing.

Filtering

System. You are a helpful assistant. Your task is to decide whether the following paper is about text tokenization. This includes tokenization, subword segmentation, tokenizer free methods, pixel language models, etc. You are given the title and abstract of the paper.

User. The title is: 'TITLE'.
The abstract is: 'ABSTRACT'.
Answer only with 'Yes' or 'No'.

Filtering (Tokenizer-free)

System. You are a helpful assistant. Your task is to decide whether the following paper is about tokenizer-free methods for text tokenization. You are given the title and abstract of the paper.

User. The title is: 'TITLE'.
The abstract is: 'ABSTRACT'.
Answer only with 'Yes' or 'No'.

Extract languages

System. You are a helpful assistant. Your task is to identify the languages that are used in the experiments of a research paper on text tokenization, based on the paper’s title and content. Answer only with the list of languages, each on its own line formatted as '* {language}'. If no language is specified, answer with 'Not specified'. If the languages is not explicitly specified but can be inferred from the content, add '(inferred)' after the language name.

User. The title is: 'TITLE'.
The content is: 'CONTENT'.
List all the main languages that are used in the paper. Do not add any explanation.

Evaluation – Extrinsic (Tasks)

System. You are a helpful assistant. Your task is to determine if the text tokenizer described in the paper is evaluated extrinsically, and if so, identify the downstream tasks used for evaluation, based on the paper’s title and content. Answer only with a list of the downstream tasks, each on its own line formatted as '* {task}'. If the paper does not perform extrinsic evaluation, answer with 'No'. Examples of downstream tasks include: language modeling, machine translation, part-of-speech tagging, etc. While examples of intrinsic metrics include: fertility, morphological accuracy, tokenization parity, etc.

User. The title is: 'TITLE'.
The content is: 'CONTENT'.

Is the text tokenizer described in the paper evaluated extrinsically? If so, what are the downstream tasks used for evaluation? Answer only with a list of the downstream tasks, each on its own line formatted as '* {task}'. If the paper does not perform extrinsic evaluation, answer with 'No'.

Evaluation – Intrinsic

System. You are a helpful assistant. Your task is to determine if the text tokenizer described in the paper is evaluated intrinsically, and if so, identify the evaluation metrics used, based on the paper’s title and content. Answer only with a list of the evaluation metrics, each on its own line formatted as '* {metric}'. If the paper does not perform intrinsic evaluation, answer with 'No'. Examples of intrinsic metrics include: fertility, morphological accuracy, tokenization parity, etc. While examples of extrinsic downstream tasks include: language modeling, machine translation, part-of-speech tagging, etc.

User. The title is: 'TITLE'.
The content is: 'CONTENT'.

Is the text tokenizer described in the paper evaluated intrinsically? If so, what are the evaluation metrics used? Answer only with a list of the evaluation metrics, each on its own line formatted as '* {metric}'. If the paper does not perform intrinsic evaluation, answer with 'No'.

Extract units

System. You are a helpful assistant. Your task is to identify the basic unit of analysis a text tokenizer uses, based on the paper’s title and content. The possible options are: bytes, characters, subwords, words, morphemes, patches, pixels, or other (specify). Answer only with a list of the most appropriate units, each on its own line formatted as '* {unit}'.

User. The title is: 'TITLE'.
The content is: 'CONTENT'.

What is the basic unit of analysis used by the text tokenizer described in the paper? Answer only with a list of the most appropriate units, each on its own line formatted as '* {unit}'.

C List of European Countries

We include the following countries: Albania, Andorra, Austria, Belarus, Belgium, Bosnia and Herzegovina, Bulgaria, Croatia, Cyprus, Czechia,

Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Liechtenstein, Lithuania, Luxembourg, Malta, Moldova, Monaco, Montenegro, Netherlands, North Macedonia, Norway, Poland, Portugal, Romania, San Marino, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Ukraine, United Kingdom, Vatican City.

D Data Sources

*CL papers are accessed through the ACL Anthology¹⁴. NeurIPS papers are from the NeurIPS proceedings¹⁵. All remaining papers are publicly accessible from OpenReview¹⁶.

E Task Normalization

We convert the task names to lowercase and remove parentheses with explanations and formatting. The normalization and grouping are based on tasks (and datasets) that appear at least 5 times. We obtain the tasks from the model’s answer instead of starting from a predefined list of tasks. Then, we first assign a task if the string contains any of the following terms, in order (the task name is in bold):

- **word segmentation**: word segmentation
- **machine translation**: translation, →
- **part-of-speech tagging**: tagging
- **named entity recognition**: entity recognition
- **dependency parsing**: dependency
- **sentiment analysis**: sentiment
- **language modeling**: modeling
- **natural language inference**: inference
- **summarization**: sum
- **similarity (generic)**: similarity
- **question answering**: question answer
- **text classification**: text classification, topic classification
- **automatic speech recognition**: automatic speech recognition
- **lemmatization**: lemmatization
- **commonsense reasoning**: common sense, commonsense
- **code generation**: code generation
- **paraphrase detection**: paraphrase
- **character tasks**: character
- **math**: math
- **linguistic judgment**: acceptability
- **reading comprehension**: comprehension

¹⁴<https://aclanthology.org/>

¹⁵<https://proceedings.neurips.cc/>

¹⁶<https://openreview.net/>

Then, we check for common abbreviations:

- **machine translation:** mt
- **part-of-speech tagging:** pos
- **named entity recognition:** ner
- **language modeling:** lm
- **question answering:** qa
- **natural language inference:** nli
- **summarization:** sum

Finally, we assign datasets to the tasks if they are not covered by the previous steps.

- **machine translation:** wmt, flores
- **natural language inference:** xnli, mnli, qnli, wnli, rte, nli
- **question answering:** mmlu, boolq, arc, open-bookqa, triviaqa, truthfulqa
- **sentiment analysis:** sst
- **commonsense reasoning:** commonsenseqa, hellaswag, piqa, winograd, copa
- **language modeling:** lambada
- **code generation:** mbpp, humaneval
- **paraphrase detection:** paws, qqp, mrpc
- **character tasks:** cute
- **math:** gsm8k, math
- **linguistic judgment:** cola, blimp
- **similarity:** sts
- **summarization:** cnn
- **reading comprehension:** squad, xquad, race, drop, quad

F Metric Normalization

We convert the metrics names to lowercase and remove parentheses with explanations and formatting. The normalization and grouping are based on the metrics that are mentioned at least twice.

We normalize the text and remove the dash character. We also group some specific metrics with their more generic variants. Therefore, we remove *top*, *@*, numbers, and the following strings: *mean*, *average*, *averaged*, *macro*, *micro*, *normalized*, *overall*, *full*, *avg*, *exact*, *perfect*, *proportion of*, *number of*, *and*, *percentage of*, *pos tagging*, *part of speech tagging*, *ner*, *pos*. *Out-of-vocabulary* is changed to *oov*.

We assign the units (in bold) if the string is any of the following:

- **precision (generic):** precision, p
- **recall (generic):** recall, in vocabulary recall, r
- **f1 score (generic):** f1, f1 score, f score, fscore, f measure, word f1, f, f1 measure, f 1 score
- **accuracy (generic):** accuracy, acc, token level accuracy, word accuracy, sentence ac-

curacy, exponent accuracy, classification accuracy, form accuracy, error rate, a, sequence accuracy

- **oov:** oov recall, oov rate, out of vocabulary recall, out of vocabulary rate
- **exact match:** match, sentence level match, match accuracy
- **rényi/shannon:** rényi entropy, renyi entropy, rényi efficiency, renyi efficiency, shannon efficiency, description length
- **correlation:** spearman's ρ , spearman's rank correlation, spearman correlation
- **fertility:** fertility, subword fertility, tokens per word, subwords per word
- **distance:** edit distance, levenshtein distance
- **morphology:** boundary precision, boundary recall, boundary f1, boundary accuracy, boundary precision recall, morphscore, morpheme precision, morpheme recall, morpheme f1, morpheme accuracy, morphological coverage, emma, emma 2
- **segmentation:** segmentation accuracy, segmentation precision, segmentation recall, word segmentation f1, tokenization f1, segmentation f1, split prediction accuracy, sentence segmentation f1
- **perplexity:** perplexity, ppl, language modeling loss, pseudo perplexity
- **tokenization parity:** tokenization parity, parity
- **overlap/similarity:** overlap, jaccard similarity, vocab overlap, frequency weighted overlap, cosine similarity, dice coefficient
- **token length:** token length distribution, token length, subword length
- **token count:** tokens, corpus token count, sequence length
- **compression:** compression rate, compression ratio, bits per byte, bytes per token
- **coverage:** coverage, vocabulary coverage, vocabulary usage
- **vocabulary size:** vocabulary size
- **type-token ratio:** type token ratio
- **cognitive plausibility:** cognitive plausibility
- **bleu:** bleu, bleu score
- **auc:** auc
- **visual task:** lpips
- **rank:** rank
- **continued words:** continued words

G Unit Normalization

First, we convert to lowercase. Then we split on newlines or commas and remove potential bullet points and lists. We a limited number of units, so we manually group them. We assign the units (in bold) if the string is any of the following:

- **morpheme**: affix, inflectional group, morpheme, morphemes, morpheme-like units, root token, suffix token, template stem token, prefix token, morpheme-like units, affix (clitic), base (stem), inflectional group (ig), paradigm components (roots suffixes), subword (morpheme), sub-segment (morpheme), subwords (morphemes)
- **syllable**: orthographic syllable, syllable, syllable units, syllables
- **bytes**: byte, bytes, byte-level units, utf-8 byte, byte (utf-8), byte-level units (fallback)
- **character**: character, characters, chinese character, japanese character, grapheme, individual character, individual characters, unicode character, unicode code point, character n-grams, character sequence, character trigrams, character (as the smallest element that can be split), character (grapheme), character (hangul syllable), character (including inline flags), character (symbol), character (unicode code point), character n-grams (subword sequences), character trigrams (character n-grams), characters (bytes), characters (graphemes), chinese character (han), grapheme (grapheme cluster), individual characters (fallback), symbol, symbol (i.e., a character from the source alphabet), symbols, unicode character (code point)
- **sub-character**: jamo, sub-characters, sub-character, sub-character, sub-characters, cv (jamo: consonant-vowel), jamo (individual hangul sub-character components), sub-character (ideograph), sub-character (stroke), subwords (jamo)
- **word**: gloss, lemma, whitespace delimiter, word, word token, words, word-form, space character, 词, bigram (two-word collocation), lemma (base form of a word), words (space-delimited tokens), word-form (candidate segment), trigram (three-word collocation), token (word), unigram (single word), word (eojeol)
- **none**: none
- **subword**: subword, subword token, sub-

word tokens, subword unit, subword units, subwords, word-piece, wordpiece, word piece, subword (bpe token), subword (bpe) token, subword (byte-pair) token, subword (byte-pair-encoded) token, subword (sentence-piece), subword (subword token), subword (token), subword (word piece), subword (wordpiece token), subword (word-piece), subword (wordpiece) token, subword (wordpiece) tokens, subword (word-piece), subword (word-piece) token, subword token (bpe), subword token (bpe-derived), subword token (byte-pair encoding), subword token (e.g., bpe/wordpiece unit), subword (character-level) units, subword (character n-gram), subword (e.g., bpe/wordpiece), subword token (word-piece), subword tokens (bpe), subword tokens (subtokens), subwords (bpe), subwords (byte-pair encoding), subwords (subtokens), subwords (token-piece tokens), token (subword token), tokens (subwords), wordpiece (subword)

- **token (generic)**: token, token chunk, token pairs, tokens, special token (e.g., <cite>)
- **phonetics**: phoneme, phonemes, acoustic tokens, short-time audio frames, acoustic tokens (vector-quantized codes), intonation units (prosodic speech segments), short-time audio frames (e.g., 10 ms windows)
- **biological**: amino-acid substitution token, nucleotide, protein family token, single nucleotide, standard amino-acid token, rna family token, intonation units, amino-acid substitution token (e.g., a→v), k-mer, k-mer (contiguous subsequence of nucleotides, variable length 1–10), nucleotide (single base), protein family (pfam) token, rna family (rfam) token, standard amino-acid token (canonical residue), subword token (variable-length nucleotide fragment)
- **sentence**: sentence, phrase, phrase tokens, phrase (multi-word token), phrase (n-gram) tokens, text chunks (e.g., sentences or fixed-length spans of tokens)
- **image**: image patch
- **patch**: patches, other (continuous values)
- **pixels**: pixels