# Evaluating Tokenizers on Typologically Distinct Languages: Insights from Turkish and Spanish

**Andreas Flensted**
IT University of Copenhagen
frao@itu.dk

**Eisuke Okuda**
IT University of Copenhagen
eiok@itu.dk

**Anders Hjulmand**
IT University of Copenhagen
ahju@itu.dk

## Abstract

Tokenization is a core processing step that segments text into input for language models. Pretrained language models such as ChatGPT uses more tokens for some languages compared to others. This suggests that the impact of tokenization can be different across languages with varying linguistic typologies. In this study, we analyzed the characteristics and performance of BPE, WordPiece, and Unigram tokenizers on Turkish, an agglutinative language, and Spanish, a fusional language, at vocabulary sizes ranging from 10K to 40K. We found that Turkish used more tokens per word, whereas Spanish used more tokens to convey the same meaning. We also found that the BPE tokenizer with a vocabulary size of 40K aligned best with Turkish morphemes. Our results may have been confounded by the use of a tiny language model and small training data size. The code is available at github.com/ah140797/NLP-project.

## 1 Introduction

Language models have demonstrated remarkable results across a suite of downstream tasks such as common sense reasoning, logic, and translation, both when fine-tuned and in few-shot settings (Brown et al., 2020).

The models are often trained on large corpora from the internet which favors the language models' capabilities towards ressource rich languages. Specifically in machine translation tasks where language models display a consistent gap in performance between high and low resource languages (Ahuja et al., 2023).

This performance gap is already prevalent in the crucial tokenization step of language modeling, where ChatGPT (Brown et al., 2020) uses 3 times more tokens to represent the same text in Arabic as in English (Petrov et al., 2023). The tokenizers serve as the bridge between raw text and the language models, by splitting text into smaller units called tokens. These tokens resemble the input texts from which the language models learn general purpose representations of the tokens. The discrepancy found in tokenizing different languages results in imbalanced learning opportunities for the models.

These differences are not attributed to the tokenizers directly but more to the inherent linguistic features across languages. Specifically languages that vary in morphological typologies such as isolating languages where words generally have one morpheme, and polysynthetic languages where a single words may have several morphemes. Additionally, some languages like Turkish are agglutinative meaning that numerous morphemes within a word are segmentable in contrast to a fusional language like Spanish, where single affixes may conflate multiple morphemes (Daniel Jurafsky, 2024).

These distinctions are critical when tokenizing multilingual text. Using a single tokenizer across diverse languages has significant implications as some languages require up to three times more training data than English (Ali et al., 2024). Similarly, performing a task in one language can cost up to 2.5 times more than the same task in English (Petrov et al., 2023).

To explore the implications of tokenizing typologically distinct languages, we conduct a comparative analysis of the three most adopted tokenizers in language models: BPE, WordPiece, and Unigram. We evaluate their performance by training them on Turkish, an agglutinative language, and Spanish, a fusional language. We examine the impact of varying vocabulary sizes of 10K, 20K, 30K, and 40K on language-model performance. We also compare characteristics of Turkish and Spanish tokenizers using token-level-metrics. Our contributions are as follows:

- **An evaluation** of the Turkish and Spanish tokenizers using the metrics **Parity**, **Fertility**, and **Bits per Character (BPC)** for comparison.

- **Result:** Larger vocabulary size improved training convergence, but failed to perform well on evaluation datasets.

- **Result:** Turkish used more tokens per word, reflecting its agglutinative nature, where multiple affixes are added to form words.

- **Result:** Turkish showed slightly worse performance in BPC compared to Spanish on the evaluation dataset using a vocabulary size 40K, indicating that Turkish might require more training data.

- **Morpheme analysis:** An assessment of the correspondence between tokenizers and Turkish morphemes, showed that BPE with a vocabulary size of 40K achieved the best alignment with morphemes.

## 2 Related work

Toraman et al., 2023 examined the effects of tokenization in Turkish, and found that sub-word tokenizers achieved the best performance at vocabulary sizes of up to 66K. Their results highlight that the effect of tokenization and vocabulary size can vary depending on the typological features of the language. A similar study also finds that increasing vocabulary sizes up to 256K improves performance for Turkish tokenizers (Kaya and Tantuğ, 2024).

Contrarily, Ali et al., 2024 argues that there exists a trade-off between tokenization efficiency and vocabulary size. They demonstrate that a larger vocabulary size leads to less tokens per word, which reduces computational costs with up to 68%, but that larger vocabulary sizes also require more computational resources.

Cotterell et al., 2020 examined whether 21 languages with diverse typological profiles were equally difficult to learn by a language model. They used parallel data for evaluation, and found that agglutinative languages, such as Finnish and Hungarian, were more difficult to model, but did not include Turkish in their study.

In this study, we compare the effects of tokenization and vocabulary sizes on the agglutinative language Turkish and the fusional language Spanish.

## 3 Methodology

### 3.1 Datasets

**OSCAR** is a multilingual corpus obtained by language-filtering of Common Crawl (Suárez et al., 2019). We used 300K de-duplicated documents of Spanish and Turkish text. We used an automatic language detector (Joulin et al., 2016) to remove documents that were not Spanish or Turkish, resulting in 299K documents. The documents constituted our training dataset used both for training our tokenizers and pre-training our language model (see Table 1).

**FLORES** is a multilingual parallel dataset originally used to benchmark machine translation systems (NLLB Team, 2024). We used all 2K parallel sentences in Spanish and Turkish from the Flores as an evaluation dataset. In Spanish, the average number of words per sentence was approximately 29, while it was 21 for Turkish, indicating that Turkish uses fewer words to convey the same meaning.

**MASSIVE** consists of parallel utterances across multiple languages (FitzGerald et al., 2023). We used 17K utterances from Spanish and Turkish as an additional evaluation dataset. The average word count per utterance was lower for both languages compared to Flores, with approximately 7 words in Spanish and 6 in Turkish.

We used the Flores dataset as the main dataset for evaluation, as the quality was higher compared to Massive, indicated by the number of words per document. Consequently, the Massive dataset was used as a secondary evaluation dataset, and its results can be found in the Appendix F.

**Evaluation on parallel data.** We used parallel data for comparing tokenizers and language models across Turkish and Spanish. In this way they were tasked with processing approximately the same information. Parallel data is required for some of the evaluation metrics used to compare tokenizers across languages.

However, the translated texts can have systematic differences compared to original texts in the given language, losing some of the inherent linguistic properties. The resulting translations often exhibit translationese, where texts are simplified and less ambiguous (Baker, 2019).

### 3.2 Tokenizers

**Byte Pair Encoding (BPE)** is a frequently used tokenizer, that starts from a small vocabulary of single-character-tokens, and recursively merges the

|  | Dataset | Words | Documents | Size (MB) |
|---|---|---|---|---|
| **es** | oscar | 173M | 299K | 887 |
|  | flores | 57K | 2K | 1 |
|  | massive | 120K | 17K | 1 |
| **tr** | oscar | 122M | 299K | 786 |
|  | flores | 42K | 2K | 1 |
|  | massive | 94K | 17K | 1 |

Table 1: All dataset statistics for Spanish (es) and Turkish (tr).

most frequent pair of tokens until the desired vocabulary size is reached (Sennrich et al., 2016).

**WordPiece** is similar to BPE, but uses a different merging-criterion. Instead of selecting the most frequent pair, WordPiece selects the pair with the highest score. Let $f$ be the frequency of a token, then: **score** $= \frac{f(A,B)}{f(A)\cdot f(B)}$. This merging strategy prioritizes pairs where individual tokens $f(A)$, and $f(B)$ are less frequent (Song et al., 2021).

**Unigram** starts from a large vocabulary and recursively removes tokens until it reaches the desired vocabulary size. At each iteration, unigram calculates a loss over the text and removes the tokens that result in the smallest loss increase given the current vocabulary (Kudo, 2018).

**Text normalization:** Text from all three datasets was normalized before applying the tokenizers. This included lowercase conversion, as well as whitespace and NFC Unicode normalization, following a previous study that trained Turkish tokenizers (Toraman et al., 2023). Additionally, newline characters were removed, as required by the automatic language detector (Joulin et al., 2016), and punctuation was split into individual characters.

### 3.3 Model training

We pre-trained a TinyBERT model (Jiao et al., 2020), which is 7.5x smaller and up to 9.4x faster than the original BERT (Devlin et al., 2019). In the Masked Language Modeling (MLM) objective, we opted for the default 15% masking strategy. We trained the model for 1 epoch using a linear learning rate scheduler, with a peak of $5e-4$, a batch size of 64 and a gradient accumulation of 8. We found suboptimal convergence using learning rate peak values of $5e-5$ and $5e-3$. See Appendix A for additional configuration and training details.

We trained the three different tokenizers (BPE, WordPiece, and Unigram) for Spanish and Turkish using different vocabulary sizes of 10K, 20K, 30K, 40K. This resulted in 24 different tokenizers used to pre-train 24 distinct TinyBERT models. The number of model parameters doubled as the vocabulary size increased from 10K to 40K (see Table 2). The training time for all models was two 48 hours using a single NVIDIA L40 GPU.

|  | Vocab size |  |  |  |
|---|---|---|---|---|
|  | **10k** | **20k** | **30k** | **40k** |
| Params. | 8M | 11M | 14M | 17M |

Table 2: Number of parameters in the TinyBERT models per vocabulary size.

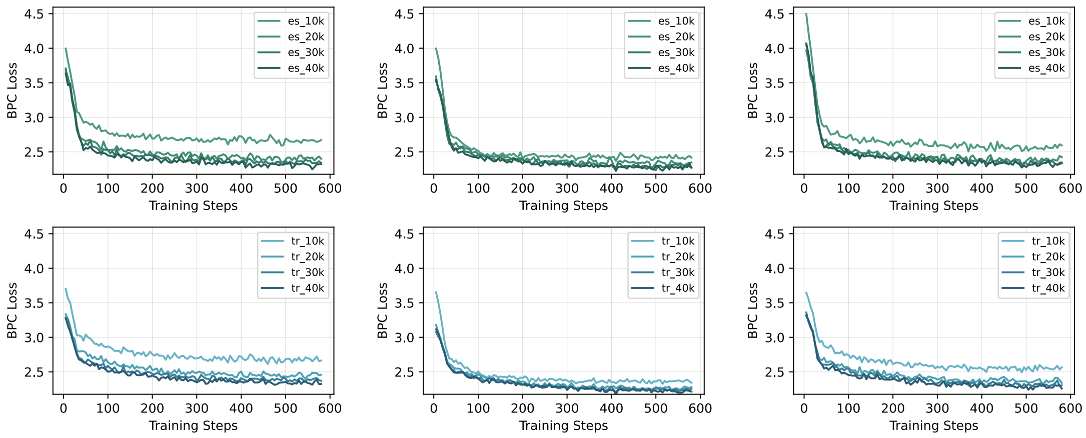
### 3.4 Evaluation Metrics

**Bits-per-character (BPC)** is the average cross-entropy of a text calculated at character level (Cotterell et al., 2020). We used BPC because it enables a comparison of performance between different configurations of the TinyBERT models across languages (Jabbar, 2024). Let a sequence of text be composed of the characters $x_1, \ldots x_T$, and the prediction of a language model be $\hat{P}_t(x_t) = \hat{P}(x_t|x_1, \ldots x_{t-1})$. BPC is then calculated as follows: BPC $= \frac{1}{\ln(2)} \cdot \frac{1}{T} \sum_{t=1}^{T} -\ln \hat{P}_t(x_t)$. Since we do not have a character-level language model, we use perplexity (PPL) to calculate BPC, where $T$ is the number of characters and $N$ is the number of tokens: BPC $= \frac{N}{T} \cdot \frac{\ln(\text{PPL})}{\ln(2)}$.

**Parity:** Given a tokenizer $T$, and two parallel sequences, $s_A, s_B$ from languages $A$, and $B$, parity computes the ratio of the number of tokens in the sequences: $\frac{|T(s_A)|}{|T(s_B)|}$ (Petrov et al., 2023). We adapted parity to compare two monolingual tokenizers: $\frac{|T_A(s_A)|}{|T_B(s_B)|}$. Parity quantifies the ratio between the number of tokens required to convey the same meaning across two languages. In the original paper, a multilingual tokenizer is deemed fair if it achieves a parity $\approx 1$.

**Fertility** quantifies the average number of tokens per word in a sequence of text: $\frac{|\text{Tokens}|}{|\text{Words}|}$ (Rust et al., 2021). A fertility score of 1 indicates that every token corresponds to every word in the given text. We expect that the agglutinative language Turkish, would lead to higher fertility scores than for Spanish.

**Morpheme-Level F1** is used in a further analysis on Turkish, to quantify the tokenizers ability to construct tokens that correspond to mor-



(a) BPE

(b) Wordpiece

(c) Unigram

Figure 1: **BPC training losses** for Spanish (green) and Turkish (blue) with tokenizers BPE (a), Wordpiece (b), Unigram (c) and vocabulary sizes 10k, 20k, 30k, and 40k. Larger vocabulary sizes showed better BPC performance.

phemes. We obtained morpheme-annotated sentences from the **Turkish Treebanks** dataset that consists of $4,851$ sentences from the web and Wikipedia (Kayadelen et al., 2020). Every sentence in the dataset is manually annotated with morpheme boundaries. For each word in a sentence, let $S_m$ and $S_t$ denote the set of morphemes and tokens used to construct the word. The true positives (tp), false positives (fp), and false negatives (fn) used to calculate the F1 are defined as follows: tp $= |S_m \cap S_t|$, fp $= |S_t \setminus S_m|$, and fn $= |S_m \setminus S_t|$. The F1-score of a sentence was calculated as the average of F1-scores for each word. By having morphemes corresponding tokens in the vocabulary, the tokenizers reflect the morphological structures found in natural language.

**Additional evaluation metrics.** We included two supplementary evaluation metrics: **Productivity** and **Normalized Sequence Length**. Descriptions and results can be found in Appendix B and C.

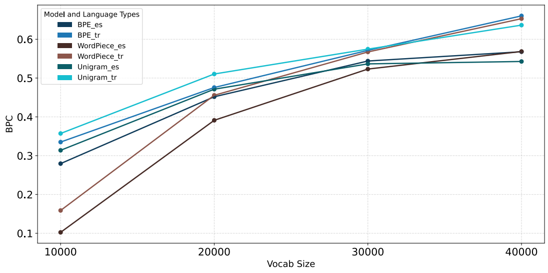


Figure 2: **BPC** results on the evaluation dataset FLORES

## 4 Results

### 4.1 Model Convergence and BPC

**Larger vocabulary size improved training convergence.** Figure 1 shows the BPC loss during pretraining for all 24 TinyBERT models. Models with higher vocabulary size showed better convergence. This improvement was more pronounced for BPE (a) and Unigram (c) compared to WordPiece (b). These trends were similar in both Spanish and Turkish. While Turkish was slightly slower to converge, both languages reached similar BPC. The observed instability in BPC loss is likely due to fluctuations in the lengths of the training documents. Interestingly, the BPC losses contrast with the Negative-Log-Likelihood (NLL) loss, which systematically increases with larger vocabulary sizes across models (see Figure 13 in Appendix F). The NLL losses may indicate that the TinyBERT models require more training examples to perform well at higher vocabulary sizes. A similar trend was observed when evaluating the models on the Flores dataset, where we also found an increase in NLL loss with larger vocabularies (see Figure 2). This likely reflects that more training data is needed to effectively learn the larger distribution of tokens in large vocabularies.

**Turkish performed worse in evaluation, especially at higher vocabulary sizes.** Interestingly, at smaller vocabulary sizes of 10K and 20K, the differences in BPC between tokenizers was more emphasized, while at larger vocabulary sizes of 30K and 40K, differences between tokenizers largely disappeared. The Turkish evaluation BPC loss at

|  |  | Vocab size |  |  |  |
|---|---|---|---|---|---|
|  | **Tokenizer** | **10k** | **20k** | **30k** | **40k** |
| **es** | BPE | 3.3 | 3.9 | 4.1 | 4.2 |
|  | WordPiece | 2.4 | 3.7 | 4.0 | 4.1 |
|  | Unigram | 3.0 | 3.5 | 3.7 | 3.8 |
| **tr** | BPE | 3.6 | 4.2 | 4.5 | 4.7 |
|  | WordPiece | 2.7 | 4.0 | 4.4 | 4.6 |
|  | Unigram | 3.6 | 4.2 | 4.5 | 4.6 |

Table 3: Average character length per token in the FLORES dataset.

30K and 40K was higher than for Spanish BPC loss. We believe that Turkish may require more training examples than Spanish as vocabulary size increases, due to its rich morphology. The same results were observed when evaluating on the Massive dataset (see Figure 10 in Appendix F).

### 4.2 Tokenization Results

**Turkish tokens are longer.** Table 3 illustrates that Turkish tokens are consistently longer across tokenizers and vocabulary sizes in the FLORES dataset. We also note that higher vocabulary sizes correspond to longer tokens. Similar results are found in the MASSIVE dataset (see Table 7 in Appendix F).

**Spanish uses more tokens to convey the same meaning.** Figure 3 illustrates that Spanish used more tokens to convey the same meaning, in more than 85% of the sentences (green) in the FLORES dataset across all tokenizers. This difference was less pronounced in the MASSIVE dataset, where Spanish used more tokens in only 60% of the sentences (see Figure 11 in Appendix F). The gap between sentence lengths in Turkish and Spanish was larger in FLORES than in MASSIVE. This discrepancy persisted throughout the tokenized representations, and likely explains why Spanish contains more tokens per sentence.

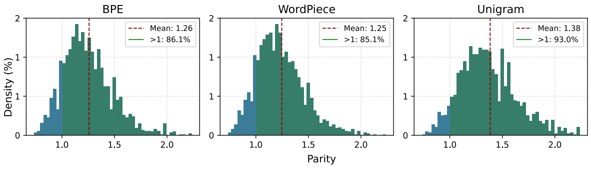


Figure 3: **Parity** results in the FLORES dataset using vocabulary size 40K. A parity higher than 1 (green) indicates that Spanish used more tokens to convey the same meaning.

**Turkish uses more tokens per word.** Figure 4 shows that Turkish had a higher fertility but only when using the BPE and WordPiece tokenizers. This might stem from Turkish being an agglutinative language, that adds multiple prefixes to represent grammatical categories such as tense and gender (see Figure 5 for an example). In contrast, Spanish is a fusional language that merges multiple grammatical categories into a single morpheme, thereby compressing the number of tokens per word.

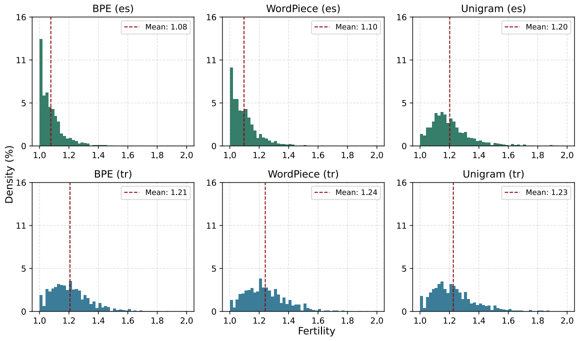


Figure 4: **Fertility** results in the FLORES dataset using vocabulary size 40K. Turkish used more tokens per word on average.

hes|per|on|y|ch|us |elizabet|ha|e|, |dro|ma|e|osa|ur|ida|e |ailesinin
|bir |türüdür |ve |vel|o|cir|ap|tor|’ |un |kuz|eni|dir|.

Figure 5: **High fertility Turkish example** from the FLORES dataset. Red vertical lines indicate the segmentation made by BPE using a vocabulary size of 40K. This example give a fertility of 2.46

## 5 Analysis

**BPE tokens corresponds best to morphemes.** Figure 6 shows the results of the morpheme level F1 for all 12 Turkish tokenizers. There is an increase in performance with higher vocabulary across tokenizers. WordPiece, in particular, shows a substantial improvement when the vocabulary size increases from 10K to 20K. This indicates that with larger vocabulary sizes, tokenizers are capable of making segmentations that more closely align with morpheme boundaries. Out of all tokenizers, BPE exhibits the most consistent performance across different vocabulary sizes. It outperforms other tokenizers when using a vocabulary size of 40K, with $3,962$ sentences ($\approx 82\%$) classified at an F1 score threshold of $0.5$ or higher. Figure 7 shows an example of a sentence with an F1 score above

0.9. In contrast, while Unigram excelled across tokenizers using a vocabulary size of 10K, it is outperformed by both WordPiece and BPE when using a vocabulary size of 40K. A recent study highlights the importance of tokens corresponding to morphemes as they retain a closer resemblance to natural linguistic features, thus potentially resulting in better representations of the original text (Parra, 2024). Based on these results, BPE serves as the best bridge between features of natural language and tokens.

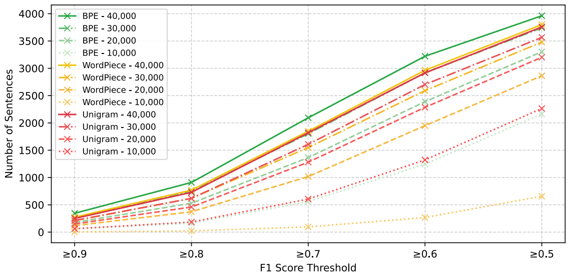


Figure 6: **Sentence level Morpheme F1** results for BPE (green), WordPiece (yellow), and Unigram (red), with vocabulary sizes of 10K, 20K, 30K and 40K. At each threshold, the F1 score is aggregated and displayed cumulatively.

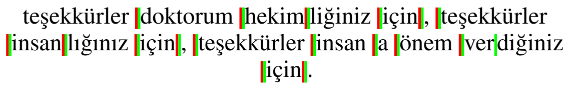

teşekkürler |doktorum |hekim|liğiniz |için|, |teşekkürler
|insan|lığınız |için|, |teşekkürler |insan |a |önem |ver|diğiniz
|için|.

Figure 7: **High F1 morpheme example** from the Turkish Treebanks dataset. Green and red vertical lines respectively denote the morpheme boundaries and tokenizer segmentations from BPE with the vocabulary size of 40K. This example gives a F1 score of 0.92.

## 6 Discussion and Limitations

**Impact of tokenizers on agglutinative and fusional languages.** We found that Turkish had a longer average token length and used more tokens per word than Spanish across tokenizers. These findings indicate that the impact of tokenizers varies between agglutinative and fusional languages. Due to time-constraints, we only included one fusional language (Spanish) and one agglutinative language (Turkish). As a result, the scope of our findings is limited to Turkish and Spanish. To obtain more robust results, future work should include a broader range of languages from these linguistic typologies.

**Turkish was more difficult to model.** When evaluating the TinyBERT models of the two languages, Turkish showed a slightly worse performance than Spanish, at vocabulary sizes 30K and 40K. However, we trained on far fewer documents than what is typically used, which could have disadvantaged larger vocabulary sizes that require more training data to capture the broader distribution of tokens. Our results are similar to Cotterell et al., 2020 which showed that agglutinative languages, such as Finnish and Hungarian, are more difficult to model.

**Spanish is more expensive.** We observed that Spanish requires more tokens than Turkish to convey the same meaning. This may be attributed to the skewed sentence-length distributions in the evaluation datasets, rather than the tokenization itself. The token-discrepancy could result in fairness disparity when using commercial LMs, which often charge users proportional to the number of tokens processed. Similarly, findings from Ahia et al., 2023 highlight that LM tokenizers favor Latin languages over underrepresented ones, mostly due to the differences in the number of tokens.

**Small model and training size.** Tao et al., 2024 highlights that the number of parameters in larger models are more capable of learning the representations of all the tokens in larger vocabulary sizes. We used the TinyBERT model, that may be limited in learning the representations of the tokens in higher vocabulary sizes such as 40K. Our results may thus have been confounded by the use of a small model.

## 7 Conclusion

In this study, we analyzed the characteristics and performance of BPE, WordPiece, and Unigram tokenizers on Turkish, an agglutinative language, and Spanish, a fusional language. We found that Turkish uses more tokens per word, had a longer average token length, and was more difficult to model. In our further analysis, we found that the BPE tokenizer with a vocabulary size of 40K aligned best with Turkish morphemes. Our findings emphasize that the impact of tokenizers differs between agglutinative and fusional languages, but they may have been confounded by the use of a small language model and training dataset.

## References

Orevaoghene Ahia, Sachin Kumar, Hila Gonen, Jungo Kasai, David R. Mortensen, Noah A. Smith, and Yulia Tsvetkov. 2023. Do All Languages Cost the

Same? Tokenization in the Era of Commercial Language Models. *arXiv preprint*. ArXiv:2305.13707 [cs].

Kabir Ahuja, Harshita Diddee, Rishav Hada, Millicent Ochieng, Krithika Ramesh, Prachi Jain, Akshay Nambi, Tanuja Ganu, Sameer Segal, Maxamed Axmed, Kalika Bali, and Sunayana Sitaram. 2023. MEGA: Multilingual Evaluation of Generative AI. *arXiv preprint*. ArXiv:2303.12528.

Mehdi Ali, Michael Fromm, Klaudia Thellmann, Richard Rutmann, Max Lübbering, Johannes Leveling, Katrin Klug, Jan Ebert, Niclas Doll, Jasper Schulze Buschhoff, Charvi Jain, Alexander Arno Weber, Lena Jurkschat, Hammam Abdelwahab, Chelsea John, Pedro Ortiz Suarez, Malte Ostendorff, Samuel Weinbach, Rafet Sifa, Stefan Kesselheim, and Nicolas Flores-Herr. 2024. Tokenizer Choice For LLM Training: Negligible or Crucial? *arXiv preprint*. ArXiv:2310.08754.

Mona Baker. 2019. Corpus Linguistics and Translation Studies*: Implications and applications. In *Researching Translation in the Age of Technology and Global Conflict*. Routledge. Num Pages: 16.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *arXiv preprint*. ArXiv:2005.14165.

Ryan Cotterell, Sabrina J. Mielke, Jason Eisner, and Brian Roark. 2020. Are All Languages Equally Hard to Language-Model? *arXiv preprint*. ArXiv:1806.03743 [cs].

Gautier Dagan, Gabriel Synnaeve, and Baptiste Rozière. 2024. Getting the most out of your tokenizer for pre-training and domain adaptation. *arXiv preprint*. ArXiv:2402.01035.

James H. Martin Daniel Jurafsky. 2024. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd edition.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint*. ArXiv:1810.04805.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2023. MASSIVE: A 1M-Example Multilingual Natural Language Understanding Dataset with 51 Typologically-Diverse Languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada. Association for Computational Linguistics.

Ximena Gutierrez-Vasques, Christian Bentz, and Tanja Samardžić. 2023. Languages Through the Looking Glass of BPE Compression. *Computational Linguistics*, 49(4):943–1001. Place: Cambridge, MA Publisher: MIT Press.

Haris Jabbar. 2024. MorphPiece : A Linguistic Tokenizer for Large Language Models. *arXiv preprint*. ArXiv:2307.07262 [cs].

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for Natural Language Understanding. *arXiv preprint*. ArXiv:1909.10351 [cs].

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2016. Bag of Tricks for Efficient Text Classification. *arXiv preprint*. ArXiv:1607.01759 [cs].

Yiğit Bekir Kaya and A. Cüneyd Tantuğ. 2024. Effect of tokenization granularity for Turkish large language models. *Intelligent Systems with Applications*, 21:200335.

Tolga Kayadelen, Adnan Öztürel, and Bernd Bohnet. 2020. A gold standard dependency treebank for Turkish. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5156–5163.

Taku Kudo. 2018. Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates. *arXiv preprint*. ArXiv:1804.10959.

NLLB Team. 2024. Scaling neural machine translation to 200 languages. *Nature*, 630(8018):841–846.

Iñigo Parra. 2024. Morphological Typology in BPE Subword Productivity and Language Modeling. *arXiv preprint*. ArXiv:2410.23656.

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

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.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural Machine Translation of Rare Words with Subword Units. *arXiv preprint*. ArXiv:1508.07909.

Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou. 2021. Fast WordPiece Tokenization. *arXiv preprint*. ArXiv:2012.15524 [cs].

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures. Leibniz-Institut für Deutsche Sprache.

Chaofan Tao, Qian Liu, Longxu Dou, Niklas Muennighoff, Zhongwei Wan, Ping Luo, Min Lin, and Ngai Wong. 2024. Scaling Laws with Vocabulary: Larger Models Deserve Larger Vocabularies. *arXiv preprint*. ArXiv:2407.13623 [cs] version: 1.

Cagri Toraman, Eyup Halit Yilmaz, Furkan Şahinuç, and Oguzhan Ozcelik. 2023. Impact of Tokenization on Language Models: An Analysis for Turkish. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22(4):116:1–116:21.

# A Model implementation details

| **Parameter** | **Configuration** |
|---|---|
| Hidden layers | 4 |
| Attention heads | 12 |
| Hidden dimension size | 312 |
| Activation function | GELU |
| Attention dropout | 0.1 |
| Hidden dropout | 0.1 |
| Initializer std. dev. | 0.02 |
| Layer-norm $\epsilon$ | $1e-12$ |

Table 4: TinyBert configuration details.

| **Parameter** | **Configuration** |
|---|---|
| Batch size | 64 |
| Gradient accumulation steps | 8 |
| Learning rate | $5e-4$ |
| Epochs | 1 |
| Learning rate scheduler | Linear |
| Warmup ratio | $5e-2$ |
| Save strategy | Epoch |
| Adam: $\beta_1$, $\beta_1$, $\epsilon$ | 0.9, 0.999, $1e-8$ |

Table 5: TinyBert training details.

## B Additional evaluation metric: Productivity

**Productivity** measures how frequent a token occurs in the set of unique words in the dataset, thus indicating how "productive" the language is (Gutierrez-Vasques et al., 2023). Let $\mathbf{W}$ denote the set of unique words, and let $\mathbf{Wt} \subseteq \mathbf{W}$ denote the unique words that contains a token $t$. Productivity is defined as $|\mathbf{Wt}|$ averaged over the vocabulary: $\frac{1}{T}\sum_{t=1}^{T}|\mathbf{Wt}|$.

**Productivity gap only exists at low vocabulary size.** Figure 8 illustrates that WordPiece had a much higher productivity when using a vocabulary size of 10K, which may be due to its merging-rule of most frequent pairs. When increasing the vocabulary size, the productivity gap becomes negligible. The difference between languages was less pronounced compared to tokenizers. These patterns was also observed in the Massive dataset (see Figure 9 in Appendix F).

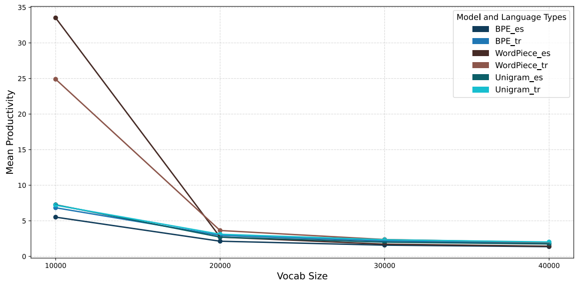


Figure 8: **Productivity** results in the FLORES Dataset.

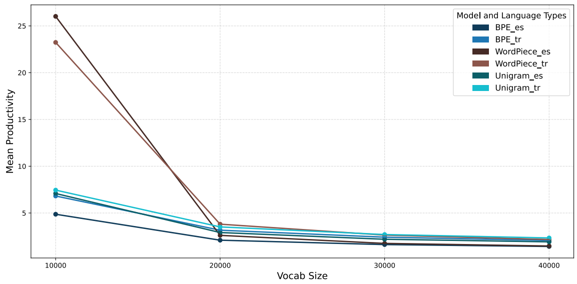


Figure 9: **Productivity** results in the Massive Dataset.

| | | NSL | |
|---|---|---|---|
| | **Dataset** | **BPE** | **Wordpiece** |
| **es** | flores | 0.92 | 0.90 |
| | massive | 0.93 | 0.91 |
| **tr** | flores | 1.01 | 0.99 |
| | massive | 0.98 | 0.96 |

Table 6: **Normalized Sequence Length** for BPE and Wordpiece relative to Unigram, using vocabulary size 40K.

## C Additional evaluation metric: Normalized Sequence Length (NSL)

**Normalized Sequence Length (NSL):** Given two tokenizers, $T_A, T_B$ and a sequence $s$, NSL computes the ratio of the number of tokens: $\frac{|T_A(s)|}{|T_B(s)|}$ (Dagan et al., 2024). We used Unigram as the relative tokenizer $T_B$, meaning that an NSL $< 1$ indicates that a tokenizer $T_A$ is more efficient at compressing the same sequence compared to Unigram.

**Tokenizer compression discrepancies are larger for Spanish.** Table 6 shows that BPE and WordPiece had lower normalized sequence length values, thereby using fewer tokens than Unigram to encode the same sentence in Spanish. These differences were less apparent for Turkish.

# D Group Contributions

Group members Andreas Flensted, Eisuke Okuda, and Anders Hjulmand contributed equally in all parts of the project: ideation phase, code implemention, conducting experiments, making figures, and writing the report.

# E Usage of Chatbots

We have used AI writing assistance as described below:

- **Assistance purely with the language of the paper:** Used to a moderate extent.
- **Short-form input assistance:** Not used.
- **Low-novelty text:** Not used.
- **New ideas:** Not used.
- **New ideas + new text:** Not used.

# F Additional results on the MASSIVE dataset

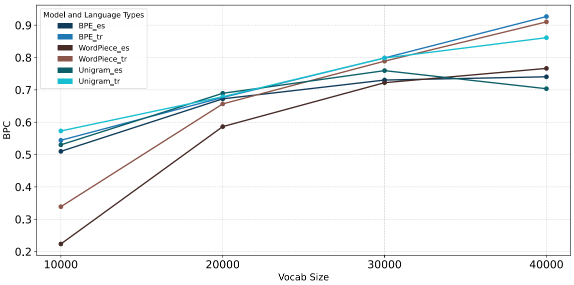


Figure 10: **BPC** results on the evaluation dataset MASSIVE

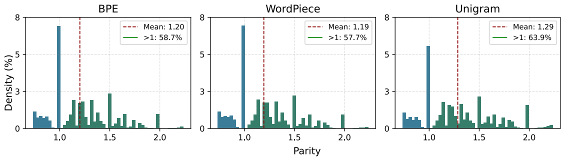


Figure 11: **Parity** results in the MASSIVE dataset using vocabulary size 40K. A parity higher than 1 (green) indicates that Spanish used more tokens to convey the same meaning across tokenizers.

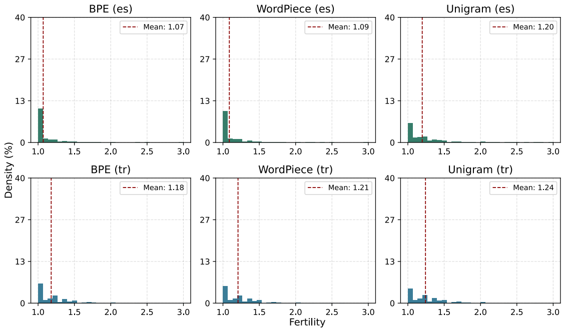
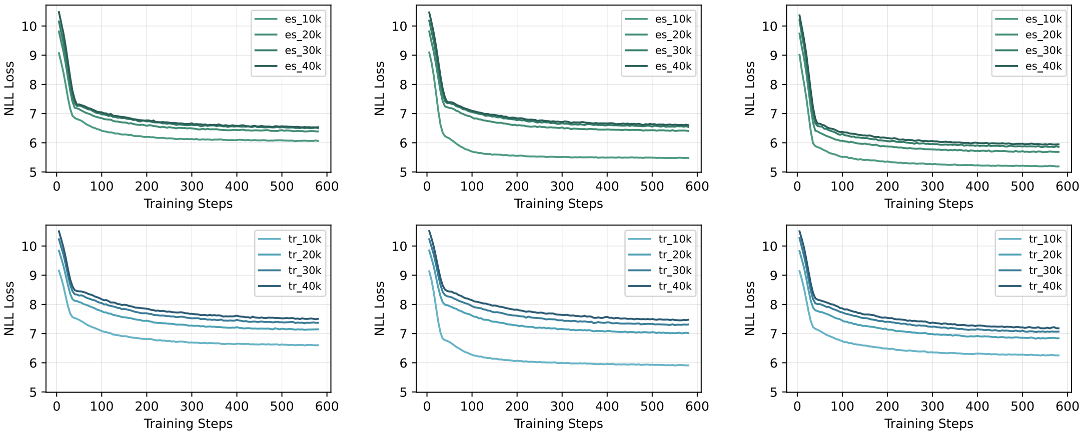


Figure 12: **Fertility** results in the MASSIVE dataset using vocabulary size 40K. Turkish used more tokens per word on average.



(a) BPE

(b) Wordpiece

(c) Unigram

Figure 13: **NLL training losses** for Spanish (green) and Turkish (blue) with tokenizers BPE (a), Wordpiece (b), Unigram (c) and vocabulary sizes 10k, 20k, 30k, and 40k.

| | | **Vocab size** | | | |
|---|---|---|---|---|---|
| | **Tokenizer** | **10k** | **20k** | **30k** | **40k** |
| **es** | BPE | 3.6 | 4.2 | 4.4 | 4.5 |
| | Wordpiece | 2.4 | 4.0 | 4.3 | 4.4 |
| | Unigram | 3.3 | 3.9 | 4.0 | 3.1 |
| **tr** | BPE | 3.7 | 4.3 | 4.6 | 4.7 |
| | Wordpiece | 2.7 | 4.1 | 4.4 | 4.6 |
| | Unigram | 3.7 | 4.2 | 4.4 | 4.5 |

Table 7: Average character length per token in the MASSIVE dataset.