

Loss-Guided Vocabulary Discovery for Multilingual Static Encoders

Boris Malashenko Ivan Jarsky Sviatoslav Stumpf Valeria Efimova

ITMO University, Saint Petersburg, Russia

btmalashenko@itmo.ru, ivanjarsky@itmo.ru, sastumpf@itmo.ru, vefimova@itmo.ru

Abstract

Static sentence encoders offer efficient multilingual inference, but their vocabularies are fixed independently of the sentence-level objective. We introduce loss-guided vocabulary discovery, a train-time procedure that uses contrastive-loss utility to identify useful adjacent token pairs, assigns each selected pair its own row, and keeps the selected pair set unchanged during continued training. In our main run, the method selects 56,071 pairs, increasing the number of embedding rows by 21.74% without changing the tokenizer or introducing contextual inference. We train the resulting encoder on 176 languages. Over three training seeds, loss-guided discovery outperforms an independently collected equal-cardinality frequency selector by 1.70 ± 0.07 , 0.12 ± 0.05 , and 0.62 ± 0.02 points on Tatoeba, BUCC, and Flores, and the margin widens with continued training. Under matched update budgets, it also improves all 12 comparisons across four additional base-model families. The selected-pair model improves the no-pair control in all four training-resource quartiles on Tatoeba, including by 0.86 points in the lowest-resource quartile. On MTEB, our encoder outperforms the strongest competing static model by 2.92 points. Loss-guided discovery thus lets the sentence-level training signal allocate multilingual vocabulary capacity within a static encoder.

1 Introduction

Static sentence encoders represent text with a vocabulary-indexed embedding table followed by pooling and normalization, avoiding a Transformer at inference. This family includes subword-compositional encoders such as fastText (Bojanowski et al., 2017) and Sent2Vec (Pagliardini et al., 2018), as well as static tables distilled or extracted from contextual models, including Model2Vec (Tulkens and van Dongen, 2024) and occurrence aggregation (Wada et al., 2025). Their

table-based inference offers predictable latency, memory use, and throughput.

That efficiency leaves little architectural freedom. Once the pooling rule and vector dimension are fixed, the principal discrete design choice is the vocabulary: it determines which text units receive independent parameters and which strings must share compositions of existing rows. This choice is especially consequential in multilingual models, where one finite table must allocate capacity across languages. Yet standard tokenizers are usually selected for corpus compression or language-model likelihood before sentence-level training, rather than for the contrastive objective that the static encoder ultimately optimizes.

We propose loss-guided vocabulary discovery, a train-time procedure for learning an auxiliary vocabulary that complements the fixed base tokenizer and is selected by the sentence-level contrastive objective. During discovery, adjacent base-token pairs receive trainable proposals and removal-utility scores. Pairs that pass the utility threshold receive dedicated rows, and the selected pair set is then fixed for continued training. Base tokenization is never modified.

Our contributions are as follows:

1. We propose loss-guided vocabulary discovery, which uses the sentence-level contrastive objective to select adjacent token pairs for a static encoder and outperforms an independently collected, equal-cardinality frequency selector on three cross-lingual benchmarks across three training seeds, with a margin that grows over 10,000 updates.
2. We introduce a multilingual static encoder trained on 176 languages that achieves the best MTEB score among public multilingual static models, outperforming the strongest prior model by 2.92 points.

3. We provide a detailed analysis of pair selection across languages, resource levels, and scripts.

The model weights¹ and the training and evaluation code² are publicly available under the Apache 2.0 license.

2 Related Work

Fixed and hashed n -gram features add local sequence information to static encoders. fastText (Bojanowski et al., 2017) represents words with hashed character n -grams, while Sent2Vec (Pagliarini et al., 2018) combines word and bigram features. Sentence-BERT (Reimers and Gurevych, 2019) and SimCSE (Gao et al., 2021) provide contextual sentence representations. LaBSE (Feng et al., 2022) aligns languages in a shared space. Model2Vec (Tulkens and van Dongen, 2024) and occurrence aggregation (Wada et al., 2025) instead distill or aggregate Transformers into static token tables. Multilingual variants of these encoders share a single vocabulary across all languages.

In multilingual models, BPE (Sennrich et al., 2016) allocates a finite vocabulary from corpus statistics, potentially yielding uneven segmentation across languages (Maronikolakis et al., 2021). Studies of capacity allocation by Zheng et al. (2021) and Liang et al. (2023), gradient-based tokenization (Godey et al., 2022), token reuse (Hamdan et al., 2025), domain adaptation (Pilana Liyanage and Yvon, 2026), and evolving vocabularies (Owodunni et al., 2026) redistribute or adapt vocabulary during training. These methods change the tokenizer or embedding interface.

First-order Taylor criteria estimate parameter importance through the loss change predicted by removing a component (Molchanov et al., 2017). We use this signal for vocabulary construction rather than for network pruning.

3 Method

Our method has three stages. First, a fixed tokenizer defines the base static encoder. Second, adjacent token pairs are evaluated during training and those above a threshold are selected. Third, each selected pair receives its own row, and training continues without changing the selected pair set.

¹<https://huggingface.co/BorisTM/loss-guided-static-multi>

²<https://github.com/BlessedTatonka/loss-guide-d-vocab>

3.1 Static encoder

Let V be a tokenizer vocabulary and $E \in \mathbb{R}^{|V| \times d}$ a trainable embedding table. For a non-empty tokenized text $x = (w_1, \dots, w_n)$, the base encoder mean-pools token rows,

$$\begin{aligned} h_0(x) &= \frac{1}{n} \sum_{i=1}^n E_{w_i}, \\ z_0(x) &= \frac{h_0(x)}{\|h_0(x)\|_2}. \end{aligned} \quad (1)$$

The similarity between two texts is $s_0(x, y) = z_0(x)^\top z_0(y)$, i.e., cosine similarity.

3.2 Adjacent-pair discovery

Candidate-pair representation. Let $\mathcal{L}$ be the set of language indices. For a text with language $\ell \in \mathcal{L}$, each ordered adjacent pair is $r_i = (w_i, w_{i+1})$. During discovery, a deterministic mapping $\phi : V^2 \rightarrow \{1, \dots, m\}$ assigns every pair to one of m train-only candidate slots. Slot j contains a vector $P_j \in \mathbb{R}^d$ and a scalar gate logit a_j . Pairs assigned to the same slot share both parameters during discovery. Each language has one learned scalar bias b_ℓ . The gate for occurrence i is $g_i^\ell = \sigma(a_{\phi(r_i)} + b_\ell)$, where σ is the logistic sigmoid. We set the fixed dummy boundary values $g_0^\ell = g_n^\ell = 0$ so that the endpoint tokens use the same notation as interior tokens. The discovery representation is

$$\begin{aligned} h(x, \ell) &= \frac{1}{n} \left[\sum_{i=1}^n \left(1 - \frac{g_{i-1}^\ell + g_i^\ell}{2} \right) E_{w_i} \right. \\ &\quad \left. + \sum_{i=1}^{n-1} g_i^\ell P_{\phi(r_i)} \right]. \end{aligned} \quad (2)$$

At $g_i^\ell = 0$, the pair is inactive: it contributes no proposal vector and does not change the weights of its two token rows. At $g_i^\ell = 1$, it contributes $P_{\phi(r_i)}$ and removes one half of the weight from each token row. Adjacent gates may be nonzero simultaneously. We normalize $h(x, \ell)$ as in Equation 1 to obtain $z(x, \ell)$. Within a fixed language, b_ℓ shifts all pair logits equally and therefore cannot change their ordering.

Contrastive discovery objective. Let $\mathcal{B} = ((p_1, q_1), \dots, (p_k, q_k))$ be a batch of positive text pairs. We optimize bidirectional in-batch InfoNCE (Gao et al., 2021),

$$\mathcal{L}_{\text{NCE}}(\mathcal{B}, s) = -\frac{1}{k} \sum_i \log \frac{e^{s(p_i, q_i)/\tau}}{\sum_j e^{s(p_i, q_j)/\tau}}, \quad (3)$$

$$\mathcal{L}_{\text{pair}}(\mathcal{B}, s) = \frac{\mathcal{L}_{\text{NCE}}(\mathcal{B}, s) + \mathcal{L}_{\text{NCE}}(\mathcal{B}^\dagger, s)}{2}, \quad (4)$$

where τ is the temperature, $\mathcal{B}^\dagger$ swaps the two texts in every pair, and $s((x, \ell_x), (y, \ell_y)) = z(x, \ell_x)^\top z(y, \ell_y)$. The other texts in the batch act as negatives. This objective updates the base rows, proposals, pair logits, and language biases.

First-order utility and admission. For occurrence o in language $\ell(o)$, we estimate the loss increase from removing its gate with a first-order Taylor score (Molchanov et al., 2017),

$$u_o = -g_o^{\ell(o)} \frac{\partial \mathcal{L}_{\text{pair}}}{\partial g_o^{\ell(o)}}. \quad (5)$$

Positive utility therefore predicts that local removal would increase the contrastive loss. For scalable replay, we retain the largest-magnitude pair-language aggregates in a fixed-capacity table. Let $(\hat{U}_{r,\ell}, \hat{C}_{r,\ell})$ denote their signed utility and support. The admitted pair set is

$$\mathcal{A}_\delta = \left\{ r : \sum_{\ell} \hat{U}_{r,\ell} \geq \delta \wedge \sum_{\ell} \hat{C}_{r,\ell} > 0 \right\}. \quad (6)$$

Thus δ selects retained candidates without specifying a target vocabulary size. Appendix A.1 gives the collection procedure.

3.3 Exact-row promotion and continuation

Each $r \in \mathcal{A}_\delta$ receives its own row $E_r \leftarrow P_{\phi(r)}$ and an exact gate logit $a_r \leftarrow a_{\phi(r)}$. Unselected proposals and the temporary pair-to-slot mapping are discarded. Training then continues with the symmetric InfoNCE loss in Equation 4 while the base table, exact pair rows, gates, and language biases remain trainable. At inference, Equation 2 uses E_{r_i} and a_{r_i} for admitted pairs and a zero gate otherwise.

4 Experiments

4.1 Training setup

We train on 22 datasets covering 176 training languages. They include web text, parallel corpora, and supervised or paraphrase pairs. Source and language sampling both use temperature 0.4, and each batch is homogeneous in source and language. Appendix C.1 lists the datasets and language identifiers and gives filtering and separation details.

The base table is initialized from mmBERT input embeddings (Marone et al., 2025). Appendix A

gives the initialization and optimization settings. After promotion, the selected model continues to 20,000 updates. Within each model family, the matched no-pair and selected-pair controls keep the prefill, base table, tokenizer, data order, optimizer, and update budget fixed. Table 1 reports 106 Tatoeba (Artetxe and Schwenk, 2019), 4 BUCC (Zweigenbaum et al., 2017), and 193 Flores (Goyal et al., 2022) subsets. For each benchmark, we macro-average F_1 over the evaluated language-pair subsets and report percentages. The mmBERT comparison is repeated with three training seeds, which change the data order; we report the mean and sample standard deviation over seeds. We call the run with the first seed the main run; it is the model whose selected pairs and MTEB scores are reported elsewhere in the paper. For single-seed contrasts, we report paired bootstrap 95% confidence intervals of the mean difference over language-pair subsets (Appendix B).

4.2 Method effectiveness

We compare loss-guided pair selection with no-pair and independently collected frequency controls for mmBERT over three seeds, and evaluate the full method across four additional base-model families under matched update budgets.

Averaged over three seeds and computed from unrounded scores, loss-guided selection improves mmBERT over the no-pair condition by 1.91 ± 0.01 , 0.17 ± 0.05 , and 1.23 ± 0.02 points on Tatoeba, BUCC, and Flores, and over the independently collected equal-cardinality frequency selector by 1.70 ± 0.07 , 0.12 ± 0.05 , and 0.62 ± 0.02 points (mean $\pm$ standard deviation of the per-seed difference). Every difference has the same sign in all three seeds, and the seed-to-seed standard deviation of each condition (0.02–0.15 points) is an order of magnitude smaller than the differences. Within each seed, the paired bootstrap interval over subsets excludes zero on Tatoeba and Flores (Table 5); BUCC has only four subsets, so its interval is not informative. Thus, the loss signal provides information beyond occurrence frequency.

Table 1 also reports four additional base-model families, XLM-R (Conneau et al., 2020), multilingual-E5 small and base (Wang et al., 2024), and LaBSE (Feng et al., 2022), whose input embeddings initialize the base table. The no-pair control and the method receive the same 3,000 updates in the same order, so the update budgets are matched. All 12 family-task differences favor loss-guided

Model	Setting	Base rows	Added pairs	Tatoeba	BUCC	Flores
mmBERT	No pairs	257,914	0	50.33 $\pm$ 0.14	97.56 $\pm$ 0.02	56.59 $\pm$ 0.09
mmBERT	Frequency	257,914	56,071	50.54 $\pm$ 0.08	97.61 $\pm$ 0.03	57.19 $\pm$ 0.10
mmBERT	Loss-guided	257,914	56,044	52.24 $\pm$ 0.15	97.73 $\pm$ 0.04	57.81 $\pm$ 0.09
XLM-R	No pairs	251,916	0	39.94	84.65	31.71
XLM-R	Loss-guided	251,916	31,701	41.41	86.86	33.62
mE5-small	No pairs	251,916	0	50.81	92.69	57.03
mE5-small	Loss-guided	251,916	49,805	52.57	93.39	57.85
mE5-base	No pairs	251,916	0	41.49	86.43	33.64
mE5-base	Loss-guided	251,916	32,957	43.14	88.49	35.56
LaBSE	No pairs	503,067	0	61.55	95.26	67.79
LaBSE	Loss-guided	503,067	45,897	62.47	95.54	68.37

Table 1: Cross-lingual macro- F_1 scores (%) over 106 Tatoeba, 4 BUCC, and 193 Flores subsets at checkpoint 2,000. The mmBERT rows are the mean $\pm$ sample standard deviation over three training seeds; Frequency selects the 56,071 most frequent pairs, the number selected by the main loss-guided run, and Loss-guided selects 55,773–56,239 pairs depending on the seed (mean shown). The four other families are single-seed runs with matched update budgets (Appendix B). Base rows is the initial table size. Best values within each comparison are bold.

discovery, ranging from 0.28 (LaBSE, BUCC) to 2.21 (XLM-R, BUCC) points, and every Tatoeba and Flores interval excludes zero (Table 5).

4.3 Training horizon and admission threshold

Table 2 follows the three mmBERT conditions of the main run from checkpoint 2,000 to 10,000. All conditions improve with training, but the margin of loss-guided selection over the frequency selector grows from +1.64 to +2.39 points on Tatoeba and from +0.63 to +0.95 on Flores (BUCC: +0.06 to +0.18), and the margin over no pairs grows from +1.90 to +2.76 and from +1.23 to +2.06 points.

Updates	Tatoeba			Flores		
	None	Freq.	Loss	None	Freq.	Loss
2,000	50.20	50.46	52.10	56.48	57.08	57.71
4,000	51.90	52.12	54.25	58.57	59.59	60.42
6,000	52.64	53.06	55.25	59.50	60.60	61.32
8,000	53.21	53.50	55.85	59.83	60.86	61.70
10,000	53.49	53.86	56.25	60.04	61.15	62.10

Table 2: mmBERT macro- F_1 (%) at successive continuation checkpoints for no pairs (None), the frequency selector (Freq.), and loss-guided selection (Loss), seed of the main run. BUCC ranges from 97.59 to 98.03 and is omitted.

Table 3 varies the admission threshold δ around the value used in the main run. Lower thresholds admit more pairs and improve all three benchmarks monotonically; even 3,411 pairs improve Tatoeba by 0.62 points over no pairs, and 99,672 pairs add a further 0.42 points over our setting. The threshold was fixed before these scores were inspected and is therefore not tuned to the benchmarks.

δ	Pairs	Tatoeba	BUCC	Flores
No pairs	0	50.20	97.59	56.48
2×10^{-3}	3,411	50.82	97.67	57.36
1×10^{-3}	8,694	51.18	97.70	57.42
5×10^{-4}	21,288	51.46	97.69	57.53
2×10^{-4} (ours)	56,239	52.10	97.69	57.71
1×10^{-4}	99,672	52.52	97.77	57.87

Table 3: Admission-threshold sensitivity for mmBERT at checkpoint 2,000, seed of the main run. The 2×10^{-4} row repeats the main run’s discovery; because utility accumulation on the GPU is not bit-reproducible, it selects 56,239 rather than 56,071 pairs, and it matches the main run’s scores within 0.05 points.

4.4 Baselines

We compare against two public multilingual static embedding models: Static-Sim³ and Potion⁴. We choose them because both provide static inference with broad multilingual coverage while following different training recipes. Together, they cover contrastive training and Transformer-to-static distillation. Static-Sim is a 1,024-dimensional model trained contrastively on 18 datasets covering 51 languages, using InfoNCE. Potion is a 256-dimensional Model2Vec model (Tulkens and van Dongen, 2024) distilled from BAAI/bge-m3 and then trained with Tokenlearn on 2 million C4 sentences sampled from 101 languages.

³<https://huggingface.co/sentence-transformers/static-similarity-mrl-multilingual-v1>. Official MTEB result tree.

⁴<https://huggingface.co/minishlab/potion-multilingual-128M>. Official MTEB result tree.

4.5 Evaluation on MTEB

The Massive Text Embedding Benchmark (MTEB) evaluates one encoder across heterogeneous tasks rather than a single retrieval dataset (Muenighoff et al., 2023). We use MTEB (Multilingual, v2) (Enevoldsen et al., 2025), which contains 131 tasks across 9 task categories.

Task category	n	Static-Sim	Potion	Ours
Bitext mining	13	50.62	40.72	53.90
Classification	43	48.60	52.36	52.87
Clustering	16	30.67	37.63	38.91
Instruction reranking	3	-1.24	-2.08	-1.04
Multilabel classification	5	13.83	15.11	17.64
Pair classification	11	74.34	71.39	72.69
Reranking	6	49.45	47.39	49.42
Retrieval	18	41.21	37.86	41.64
STS	16	64.02	61.23	64.99
Mean (task type)		41.28	40.18	43.45
Mean (task)		47.21	47.20	50.13

Table 4: MTEB scores by task category, scaled by 100. Here n is the task count. Static-Sim and Potion are the baselines in Section 4.4. The aggregate rows average the 9 categories and all 131 tasks. Best values are bold.

Table 4 reports results for all 3 models. Our model reaches 50.13 Mean (task), compared with 47.21 for Static-Sim and 47.20 for Potion. Its Mean (task type) is 43.45, compared with 41.28 and 40.18, respectively.

Our model ranks first in 7 of the 9 categories. Using unrounded category means, its largest margins over the stronger baseline in each category are 3.29 points on bitext mining and 2.53 points on multilabel classification. It is also ahead on classification, clustering, instruction reranking, retrieval, and STS.

5 Analysis

In the main run, discovery selects 56,071 adjacent pairs and increases the number of table rows by 21.74%. The resulting pair set is a data-dependent extension of the base vocabulary: pairs receive their own rows according to their estimated contribution to the sentence objective, while the tokenizer remains unchanged. Decoded examples include within-word units and cross-word combinations from all four language-resource groups (Appendix D.2).

We group the 176 training languages into four equal-sized quartiles by corpus size. In the matched checkpoint-2,000 comparison of the main run, the method improves the group means over the no-pair

control in every Tatoeba quartile. In Q1–Q4 order, the gains are $[+0.86, +0.93, +2.53, +2.76]$ points. Thus, the selected pairs benefit the full range of training languages, including the lowest-resource quartile. Appendix D.1 reports the corresponding subset counts and win rates.

Pair allocation itself follows exposure rather than need: in Q1–Q4 order, the four quartiles receive $[4.35, 6.18, 17.18, 72.28]\%$ of fractional row credit, row credit correlates with a language’s share of the training data ($\rho = 0.828$), and the per-language selection rate does not ($\rho = -0.132$). Loss-guided discovery therefore reproduces the training distribution instead of rebalancing it. Q1–Q2 languages receive few dedicated rows yet still improve, consistent with selected rows being shared globally; an explicit allocation quota remains future work (see Limitations).

The evidence used to select pairs spans all 27 scripts represented in the training data, and 22.05% of selected pairs are supported by more than one language. Latin, Cyrillic, and Arabic account for 66.16%, 11.42%, and 8.17% of fractional row credit, while the other 24 scripts account for 14.25%. Once selected, every pair row is shared globally, allowing the same local unit to be used wherever it occurs.

6 Conclusion

We introduced loss-guided adjacent-pair discovery, a method that makes vocabulary construction part of sentence-level training. Starting from a fixed multilingual tokenizer, the method identifies useful local units, assigns each selected pair its own embedding row, and retains the simple static-encoder architecture at inference time.

Trained on 176 languages, the resulting encoder reaches 50.13 Mean (task) on MTEB, outperforming the strongest previous static model by 2.92 points and ranking first in 7 of 9 task categories. Matched comparisons show gains on Tatoeba, BUCC, and Flores over both no pairs and an independently collected equal-cardinality frequency selector across three training seeds, a margin that widens over 10,000 updates, consistent gains in four further base-model families under matched update budgets, and positive Tatoeba group-mean improvements across all four training-data quartiles. A static encoder can therefore refine its vocabulary with the sentence objective without contextual inference.

Limitations

Compared with pure mean pooling, pair matching and the additional embedding rows increase inference latency, model size, and runtime memory. Across batch sizes 64–512, the measured mean CPU latency ratio is $1.67\times$.

The method considers only adjacent pairs from the initial tokenizer and assumes a known language identifier. We do not evaluate longer or non-contiguous units, code-switched input, or automatic language identification.

Pair allocation follows the training mixture without an explicit language-balancing quota. The learned vocabulary can therefore reproduce resource imbalances and under-allocate capacity to less represented languages.

The improvement over independent frequency selection varies by benchmark. It is 1.70 points on Tatoeba and 0.62 on Flores, but only 0.12 on BUCC, whose four subsets are close to saturation. Seed replication covers the mmBERT comparison at checkpoint 2,000; the horizon, threshold, and family comparisons use a single seed, and their uncertainty is characterized only over language-pair subsets. Discovery always runs for 1,000 updates with the same threshold; neither was tuned per family.

Ethical Considerations

Our training mixture includes web text, parallel corpora, and supervised or paraphrase pairs. These sources may contain personal information, offensive content, and social or cultural biases. Filtering reduces but cannot eliminate these risks, and the resulting embeddings may reproduce associations present in the training data. We do not redistribute the raw corpora. Their use remains governed by the original providers' licenses and terms.

Training on 176 languages does not imply equal quality across languages, dialects, scripts, or communities. Aggregate benchmark improvements should not be interpreted as evidence of equitable performance. Downstream users should evaluate the model for their target population and application. The model is intended for general representation and retrieval tasks, not for profiling, sensitive-attribute inference, or fully automated high-stakes decisions.

Acknowledgments

This work supported by the Ministry of Economic Development of the Russian Federation (IGK 000000C313925P4C0002), agreement No139-15-2025-010.

References

- Mikel Artetxe and Holger Schwenk. 2019. [Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond](#). *Transactions of the Association for Computational Linguistics*, 7:597–610.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. [Enriching word vectors with subword information](#). *Transactions of the Association for Computational Linguistics*, 5:135–146.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Kenneth Enevoldsen et al. 2025. [MMTEB: Massive multilingual text embedding benchmark](#). In *International Conference on Learning Representations*, pages 101715–101771.
- Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. [Language-agnostic BERT sentence embedding](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. [SimCSE: Simple contrastive learning of sentence embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Nathan Godey, Roman Castagné, Éric de la Clergerie, and Benoît Sagot. 2022. [MANTA: Efficient gradient-based tokenization for end-to-end robust language modeling](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2859–2870, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc’Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. [The FLORES-101 evaluation benchmark for low-resource and multilingual](#)

- machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.
- Nancy Hamdan, Osama Rakan Al Mraikhat, and Fadi A. Zaraket. 2025. **R-BPE: Improving BPE-tokenizers with token reuse**. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22940–22948, Suzhou, China. Association for Computational Linguistics.
- Philipp Koehn. 2005. **Europarl: A parallel corpus for statistical machine translation**. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.
- Davis Liang, Hila Gonen, Yuning Mao, Rui Hou, Naman Goyal, Marjan Ghazvininejad, Luke Zettlemoyer, and Madian Khabsa. 2023. **XLM-V: Overcoming the vocabulary bottleneck in multilingual masked language models**. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13142–13152, Singapore. Association for Computational Linguistics.
- Pierre Lison and Jörg Tiedemann. 2016. **OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles**. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).
- Marc Marone, Orion Weller, William Fleshman, Eugene Yang, Dawn Lawrie, and Benjamin Van Durme. 2025. **mmBERT: A modern multilingual encoder with annealed language learning**. *Preprint*, arXiv:2509.06888.
- Antonis Maronikolakis, Philipp Dufter, and Hinrich Schütze. 2021. **Wine is not v i n. on the compatibility of tokenizations across languages**. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2382–2399, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. 2017. **Pruning convolutional neural networks for resource efficient inference**. In *5th International Conference on Learning Representations*, Toulon, France.
- Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. **MTEB: Massive text embedding benchmark**. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.
- Abraham Toluwase Owodunni, Orevaoghene Ahia, and Sachin Kumar. 2026. **FLEXITOKENS: Flexible tokenization for evolving language models**. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 17170–17190, San Diego, California, United States. Association for Computational Linguistics.
- Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. **Unsupervised learning of sentence embeddings using compositional n-gram features**. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 528–540, New Orleans, Louisiana. Association for Computational Linguistics.
- Guilherme Penedo, Hynek Kydlíček, Loubna Ben Al-lal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro von Werra, and Thomas Wolf. 2024. **The FineWeb datasets: Decanting the web for the finest text data at scale**. *Preprint*, arXiv:2406.17557.
- Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro von Werra, and Thomas Wolf. 2025. **FineWeb2: One pipeline to scale them all—adapting pre-training data processing to every language**. *Preprint*, arXiv:2506.20920.
- Vijini Pilana Liyanage and François Yvon. 2026. **AdaptBPE: From general purpose to specialized tokenizers**. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2607–2620, Rabat, Morocco. Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2019. **Sentence-BERT: Sentence embeddings using Siamese BERT-networks**. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021a. **Wiki-Matrix: Mining 135M parallel sentences in 1620 language pairs from Wikipedia**. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.
- Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021b. **CCMatrix: Mining billions of high-quality parallel sentences on the web**. 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 6490–6500, Online. 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,

Berlin, Germany. Association for Computational Linguistics.

Stéphan Tulkens and Thomas van Dongen. 2024. [Model2Vec: Fast state-of-the-art static embeddings](#).

Takashi Wada, Yuki Hirakawa, Ryotaro Shimizu, Takahiro Kawashima, and Yuki Saito. 2025. [Static word embeddings for sentence semantic representation](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6195–6211, Suzhou, China. Association for Computational Linguistics.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. [Multilingual E5 text embeddings: A technical report](#). *Preprint*, arXiv:2402.05672.

Bo Zheng, Li Dong, Shaohan Huang, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. 2021. [Allocating large vocabulary capacity for cross-lingual language model pre-training](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3203–3215, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2017. [Overview of the second BUCC shared task: Spotting parallel sentences in comparable corpora](#). In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, Vancouver, Canada. Association for Computational Linguistics.

A Implementation and training details

The 256,000-row content table concatenates the input embeddings of jhu-clsp/mmBERT-base and jhu-clsp/mmBERT-small, projects the result from 1,152 to 1,024 dimensions with PCA, and applies Zipf-SIF rank weights. We append 1,914 zero-initialized language-marker rows, producing a 257,914-row base table. Discovery starts with 262,144 zero proposal vectors and logits and language biases initialized to -2 .

A.1 Adjacent-pair discovery and promotion

During discovery, each forward pass aggregates utility by pair and language, retains up to 4,096 largest-magnitude records, and uses a signed CountSketch to populate a 524,288-record exact collector. $\hat{U}_{r,\ell}$ and $\hat{C}_{r,\ell}$ denote retained utility and support.

We aggregate retained records over language and select every pair satisfying

$$\sum_{\ell} \hat{U}_{r,\ell} \geq 2 \times 10^{-4}, \quad \sum_{\ell} \hat{C}_{r,\ell} \geq 1. \quad (7)$$

The proposal table bounds discovery but does not prescribe the number of selected pairs.

Selected pairs inherit trained proposal vectors and logits. Each selected pair is copied from the shared proposal table into its own row. After promotion, the selected pair set remains unchanged and all retained parameters continue training.

A.2 Optimization

Both stages use symmetric cached InfoNCE, BF16, AdamW, and an inverse-square-root learning-rate schedule. They use peak learning rate 10^{-3} , weight decay 0.05, and a maximum sequence length of 128 tokens. The temperature is $\tau = 1/17 \approx 0.059$ during discovery and $\tau = 0.05$ during continuation. An effective pair batch of 16,384 is formed from cached mini-batches of 2,048. Discovery uses 250 warmup updates. Continuation starts a fresh optimizer after promotion, uses 1,000 warmup updates, and the selected model continues to 20,000 updates.

B Additional experiments and diagnostics

Within a seed, all conditions use the same 176 language-script identifiers, filtered data, batch order, and optimizer. The selected-pair model then continues to 20,000 updates for the final model.

At checkpoint 20,000, replacing all learned language biases by their global mean changes Tatoeba and BUCC by $[-0.47, -0.06]$ points, respectively. This evaluation-time diagnostic shows that the learned language calibration contributes to quality.

Seed replication and uncertainty. The three mmBERT conditions in Table 1 were trained with three seeds under the configuration of the main run; the seed controls the data order. Each seed yields one no-pair, one frequency, and one loss-guided model evaluated at checkpoint 2,000. Repeating the main run’s seed selects 56,239 pairs rather than the main run’s 56,071, because utility accumulation on the GPU is not bit-reproducible; its scores match the main run within 0.05 points. The other two seeds select 55,773 and 56,119 pairs, while the frequency selector always uses 56,071. Per-seed differences of loss-guided minus frequency are $[+1.64, +1.78, +1.67]$ on Tatoeba, $[+0.06, +0.16, +0.14]$ on BUCC, and $[+0.63, +0.60, +0.63]$ on Flores. Table 5 reports, for the main contrasts, the mean difference over language-pair subsets with a paired bootstrap 95%

interval (10,000 resamples) and the share of subsets in which loss-guided selection is ahead. The horizon runs in Table 2 continue the main-seed models to 10,000 updates and are evaluated every 2,000 updates. The threshold sweep in Table 3 repeats discovery with $\delta \in \{1, 2, 5, 10, 20\} \times 10^{-4}$ and continues each model for 2,000 updates.

Matched-budget family comparison. For XLM-R, multilingual-E5 small and base, and LaBSE, the base table is the input-embedding matrix of the base model restricted to tokenizer-reachable rows, plus the same 1,914 zero-initialized language-marker rows as the mmBERT table. Both arms use batch 2,048, temperature $\tau = 0.05$, peak learning rate 10^{-3} , weight decay 0.05, an inverse-square-root schedule, and the main run’s seed. The method runs 1,000 discovery updates ($\delta = 2 \times 10^{-4}$, promotion at update 1,000) followed by 2,000 continuation updates; the control runs a 1,000-update no-pair prefill with the same warmup and data order, followed by the same 2,000-update continuation. Both arms therefore see the same 3,000 batches in the same order, and both are evaluated at continuation checkpoint 2,000.

C Data and sampling protocol

C.1 Training data and sampling

Training draws on 22 data sources. Web text comes from FineWeb and FineWeb2⁵ (Penedo et al., 2024, 2025). Parallel corpora include CCMatrix (Schwenk et al., 2021b), Europarl (Koehn, 2005), Global Voices, MUSE, OpenSubtitles (Lison and Tiedemann, 2016), Talks, WikiMatrix (Schwenk et al., 2021a), and WikiTitles.⁶ Supervised and paraphrase data come from AllNLI, AltLex, COCO Captions, Flickr30k, NLI for SimCSE, Quora Duplicates, Simple Wikipedia, Stack Exchange Duplicates, WikiAnswers, LTG Wiki Paraphrases, OpenGloss, and REDIS LLM Paraphrases.⁷ After filtering, the mixture contains 2,304 Parquet shards and 176 training language identifiers.

Training languages. The complete training list is:

⁵Hugging Face datasets: [FineWeb](#), [FineWeb2](#).

⁶Hugging Face datasets: [CCMatrix](#), [Europarl](#), [Global Voices](#), [MUSE](#), [OpenSubtitles](#), [Talks](#), [WikiMatrix](#), [WikiTitles](#).

⁷Hugging Face datasets: [AllNLI](#), [AltLex](#), [COCO Captions](#), [Flickr30k](#), [NLI for SimCSE](#), [Quora Duplicates](#), [Simple Wikipedia](#), [Stack Exchange Duplicates](#), [WikiAnswers](#), [LTG Wiki Paraphrases](#), [OpenGloss](#), [REDIS LLM Paraphrases](#).

aeb_Arab, afr_Latn, als_Latn, amh_Ethi, apc_Arab, arb_Arab, ars_Arab, ary_Arab, arz_Arab, asm_Beng, ast_Latn, azb_Arab, azj_Latn, bak_Cyrl, ban_Latn, bbc_Latn, bel_Cyrl, ben_Beng, ben_Latn, bew_Latn, bho_Deva, bod_Tibt, bos_Latn, bre_Latn, bul_Cyrl, cat_Latn, ceb_Latn, ces_Latn, chv_Cyrl, ckb_Arab, cmn_Hani, crh_Latn, csb_Latn, cym_Latn, dan_Latn, deu_Latn, div_Thaa, ell_Grek, eng_Latn, epo_Latn, eus_Latn, ewe_Latn, fao_Latn, fas_Arab, fij_Latn, fil_Latn, fin_Latn, fra_Latn, fry_Latn, fur_Latn, gaz_Latn, gla_Latn, gle_Latn, glg_Latn, glk_Arab, grc_Grek, gsw_Latn, guj_Gujr, hat_Latn, hau_Latn, haw_Latn, hbo_Hebr, heb_Hebr, hin_Deva, hin_Latn, hrv_Latn, hsb_Latn, hun_Latn, hye_Armn, ibo_Latn, ilo_Latn, ind_Latn, isl_Latn, ita_Latn, jav_Latn, jpn_Jpan, kab_Latn, kan_Knda, kat_Geor, kaz_Cyrl, khk_Cyrl, khm_Khmr, kin_Latn, kir_Cyrl, kmr_Latn, kor_Hang, krc_Cyrl, lao_Lao, lat_Latn, lij_Latn, lim_Latn, lin_Latn, lit_Latn, lmo_Latn, ltg_Latn, ltz_Latn, lug_Latn, lus_Latn, lvs_Latn, mai_Deva, mal_Mlym, mar_Deva, mhr_Cyrl, mkd_Cyrl, mlt_Latn, mri_Latn, mya_Mymr, nde_Latn, nds_Latn, nld_Latn, nno_Latn, nob_Latn, npi_Deva, nya_Latn, oci_Latn, ory_Orya, pan_Guru, pap_Latn, pbt_Arab, plt_Latn, pms_Latn, pol_Latn, por_Latn, rmy_Latn, ron_Latn, run_Latn, rus_Cyrl, sah_Cyrl, san_Deva, san_Latn, scn_Latn, sco_Latn, shn_Mymr, sin_Sinh, slk_Latn, slv_Latn, smo_Latn, sna_Latn, srp_Cyrl, snd_Arab, som_Latn, sot_Latn, spa_Latn, srd_Latn, srp_Cyrl, srp_Latn, sun_Latn, swe_Latn, swh_Latn, tam_Latn, tam_Taml, tat_Cyrl, tat_Latn, tel_Latn, tel_Telu, tgk_Cyrl, tha_Thai, tir_Ethi, tuk_Latn, tur_Latn, tyv_Cyrl, uig_Arab, uig_Cyrl, uig_Latn, ukr_Cyrl, urd_Arab, urd_Latn, uzb_Cyrl, uzb_Latn, vie_Latn, wln_Latn, xho_Latn, ydd_Hebr, yor_Latn, yue_Hani, zsm_Latn, zul_Latn.

Source and language sampling temperatures are both 0.4, batches are source- and positive-language-homogeneous, and canonicalized texts do not repeat within a batch. Dataset identity is separated from evaluation.

Machine-readable provenance records include configuration and data revisions, splits, row counts, language coverage, shard hashes, and license fields.

Each update consumes a 16,384-pair batch. Source s receives weight $N_s^{0.4}$ and language ℓ within that source receives $N_{s,\ell}^{0.4}$. Deterministic smooth weighted round robin realizes these weights. Exact allocations are included in the scheduler record.

Within (s, ℓ) , ℓ is the positive-side language. No canonicalized text may appear twice in a batch. Markers are prepended separately, so a parallel batch can be English-to- ℓ while a same-language batch is ℓ -to- ℓ . Appendix A.2 records the remaining settings.

The matched comparison reports Tatoeba, BUCC, and Flores separately and as an unweighted mean. Adjacent-sentence retrieval uses at most one pair per document. Monolingual subsets receive their declared marker, cross-lingual retrieval uses the query and document language markers, and mixed-language subsets use one task-level marker to avoid label leakage. All scores use normalized embeddings. The utility threshold was fixed before these scores were inspected.

Contamination audit. We canonicalize every training and evaluation text with Unicode NFKC, case-folding, and whitespace normalization. Exact

Model	Contrast	Tatoeba (106 subsets)		Flores (193 subsets)	
		Δ [95% CI]	Win	Δ [95% CI]	Win
mmBERT, seed 1	Loss – Freq.	+1.64 [+1.22, +2.06]	81	+0.63 [+0.49, +0.79]	74
mmBERT, seed 2	Loss – Freq.	+1.78 [+1.40, +2.16]	82	+0.60 [+0.46, +0.75]	68
mmBERT, seed 3	Loss – Freq.	+1.67 [+1.32, +2.05]	87	+0.63 [+0.48, +0.80]	73
mmBERT, seed 1	Loss – None	+1.90 [+1.46, +2.34]	84	+1.23 [+1.05, +1.42]	86
mmBERT, 10k	Loss – Freq.	+2.39 [+1.84, +2.97]	86	+0.95 [+0.68, +1.27]	72
XLM-R	Loss – None	+1.47 [+1.16, +1.78]	87	+1.90 [+1.65, +2.17]	91
mE5-small	Loss – None	+1.77 [+1.50, +2.04]	86	+0.82 [+0.68, +0.97]	83
mE5-base	Loss – None	+1.65 [+1.36, +1.95]	88	+1.91 [+1.67, +2.17]	92
LaBSE	Loss – None	+0.92 [+0.68, +1.16]	83	+0.57 [+0.45, +0.72]	79

Table 5: Paired differences over language-pair subsets at checkpoint 2,000 (mmBERT, 10k: checkpoint 10,000) in percentage points, with paired bootstrap 95% intervals and the share of subsets (%) in which loss-guided selection (Loss) is ahead of the frequency selector (Freq.) or no pairs (None). Seed 1 is the main run’s seed.

overlap is detected from SHA-256 hashes of canonical strings. For strings of at least 50 characters, we additionally flag near duplicates whose Unicode character 5-gram Jaccard similarity is at least 0.8. The audit covers every text field loaded from the Tatoeba, BUCC, and MTEB evaluation splits and records the training source, evaluation task, split, and example identifiers for every match. Candidate matches are manually inspected and exact and near-duplicate counts are reported separately by source and task.

D Multilingual vocabulary analysis

D.1 Allocation and quality

We rank the 176 languages by usable scheduler rows (language-ID tie-breaker) and form four equal 44-language groups. Q1 is lowest resource and Q4 highest. The evidence record covers all 56,071 selected pairs and their language counts.

Task	Group	n	Δ no pair	Win (%)
Tatoeba	Q1	5	+0.86	60.0
	Q2	12	+0.93	50.0
	Q3	26	+2.53	92.3
	Q4	43	+2.76	97.7
	unseen	20	−0.03	60.0

Table 6: Matched Tatoeba results by resource group. Δ is the percentage-point change from no pairs, and Win is the share of positive subsets. “Unseen” languages are outside the training population.

In Q1–Q4 order, row-credit, discovery-batch, and utility shares are [4.35, 6.18, 17.18, 72.28]%, [3.1, 6.4, 16.1, 74.4]%, and [3.08, 4.46, 12.22, 80.24]%, respectively. Row credit correlates with exposure ($\rho = 0.828$), while selection rate has $\rho = -0.132$ with usable rows. Pairs supported by one, two,

three–four, and at least five languages account for [77.95, 13.57, 5.73, 2.76]%. Selected rows remain globally available.

Evidence spans 27 scripts. Latin, Cyrillic, Arabic, and the other 24 scripts receive [66.16, 11.42, 8.17, 14.25]% of row credit, reflecting exposure rather than a script quota.

At checkpoint 20,000, replacing the learned gate biases with their global mean lowers Tatoeba and BUCC by [0.47, 0.06] points (Appendix B).

D.2 Pair features

Table 7 gives 5 varied English examples and one non-English example from each of Q1–Q4. We report $10^3 \hat{U}_r$, where $\hat{U}_r = \sum_{\ell} \hat{U}_{r,\ell}$.

Type/lang.	Pieces $\rightarrow$ pair	U	Support
<i>English</i>			
Clitic	' + s $\rightarrow$'s	606	785,110
Punctuation	However + , $\rightarrow$ However,	17.7	70,141
Prefix	non + - $\rightarrow$ non-	12.4	27,395
Phrase	each + other $\rightarrow$ each other	22.9	17,026
Named entity	United + States $\rightarrow$ United States	5.55	36,130
<i>Other languages</i>			
Ilocano (Q1)	dag + iti $\rightarrow$ dagiti (<i>article</i>)	12.9	13,577
Yoruba (Q2)	aw + on $\rightarrow$ awon (<i>plural marker</i>)	8.56	25,741
Esperanto (Q3)	ĉ + i $\rightarrow$ ĉi (<i>proximal</i>)	10.2	7,436
Albanian (Q4)	q + ë $\rightarrow$ që (<i>that/which</i>)	26.2	48,459

Table 7: Selected pair examples. Support is the retained occurrence count. Boundary markers are omitted.