ULTRABERT: BEYOND THE LIMIT OF BERT PRE-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent bidirectional encoders have been modernized with innovations from causal large language models (LLMs). However, the pretraining recipe itself remains inherited from small-scale conventions, leaving encoders underoptimized, with no single model performing competitively across both classification and retrieval tasks. In this work, we investigate how far encoder performance can be pushed by revisiting core design choices in training, revealing three key findings: (1) Encoders saturate on downstream tasks beyond a few trillion tokens, indicating that additional compute past this point does not help. (2) Across model sizes, lower masking ratios benefit classification while higher ones benefit retrieval, contradicting prior work suggesting larger ratios for larger models. (3) Distillation with pruning improves classification at low masking ratios but degrades retrieval at high ones. Guided by these findings, we present UltraBERT, a single encoder that assigns separate LM heads to low and high masking ratios within a batch and restricts the distillation objective to the low ratio. To our knowledge, UltraBERT is the first encoder to achieve state of the art on both classification and retrieval tasks, reaching 91.0 on GLUE and 47.7 on BEIR. Together, these results show how far encoder performance can be pushed and provide a principled basis for compute allocation, masking ratio design, and knowledge transfer in bidirectional models.

1 INTRODUCTION

Bidirectional encoders remain essential for non-generative applications, even amid the rapid progress of decoder-only large language models (LLMs). Their modest inference cost allows them to embed or score large volumes of text efficiently, making them the backbone of retrieval-augmented generation (RAG) pipelines (Lewis et al., 2020), quality filtering of pre-training data (Penedo et al., 2024), and reward modeling for reinforcement learning (Gisserot-Boukhlef et al., 2026b). While decoders owe much of their popularity to their generalization to unseen tasks, fine-tuned encoders have often been shown to match much larger decoders on well-defined tasks at a fraction of the cost (Bucher & Martini, 2024; Edwards & Camacho-Collados, 2024; Zhang et al., 2025b).

Recent work has increasingly modernized encoders with innovations from decoder-only LLMs (Portes et al., 2023; Warner et al., 2025; Breton et al., 2025), adopting architectural advances such as rotary position embeddings (Su et al., 2024), FlashAttention-2 (Dao, 2024), along with token scaling (Sardana et al., 2024). However, the pre-training recipe itself remains inherited from small-scale conventions. In particular, the masked language modeling (MLM) objective, which is specific to encoders, is still configured with lessons from prior studies under limited token budgets (Wettig et al., 2023; Gisserot-Boukhlef et al., 2026a), where convergence speed is entangled with final performance. This core component trades off classification against retrieval, leaving encoders trained with a single configuration specialized, with no single model performing competitively on both.

In this work, we begin by revisiting encoder pretraining at scale, and reveal three key findings. (1) Training encoders on up to 8.8T tokens, we find that downstream performance saturates around 1–2T tokens even as training and validation losses keep decreasing, unlike decoders, which continue to improve. This indicates an upper bound that additional tokens cannot raise. (2) At saturation, where convergence speed no longer confounds the comparison, classification consistently favors lower masking ratios across model sizes, contradicting prior work suggesting larger ratios for larger models, whereas retrieval favors higher ones. (3) Unlike in decoders, pruning and distillation from

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

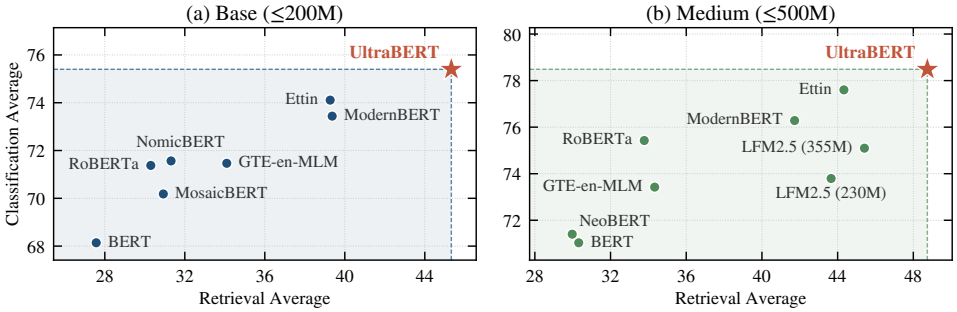


Figure 1: Classification and retrieval performance of encoders in two size groups, (a) base (≤200M) and (b) medium (≤500M). The classification average is the mean of GLUE, SQuAD 2.0, Multi-CoNER v2, Few-NERD, and RewardBench, and the retrieval average is the mean of BEIR, RAR-b (averaged over its Code and Math subsets), and MLDR. All scores are taken from Table 3.

a stronger teacher does not uniformly raise encoder performance: it improves classification at a low masking ratio but degrades retrieval at a high one.

Guided by these findings, we present UltraBERT, which reconciles the two masking regimes within a single encoder rather than settling for a single configuration. Observing that the two regimes shape the LM head into distinct geometries, we propose *LM head decoupling*, which mixes 5% and 50% masked sequences within each batch while assigning a separate LM head to each ratio. Furthermore, we restrict distillation to the low-ratio sequences, while training the high-ratio ones with standard MLM. Starting from a 1.3B teacher, we obtain 380M and 150M models through pruning and distillation, trained on 2T tokens with native 32K context. Across classification and retrieval, where the strongest baseline differs, UltraBERT achieves state of the art on both, reaching 91.0 on GLUE (Wang et al., 2018) and 47.7 on BEIR (Thakur et al., 2021). Our 150M model matches RoBERTa-large (Liu et al., 2019) on classification with 2.3× fewer parameters, and outperforms Ettin-encoder-400M (Weller et al., 2026) on retrieval at 2.6× smaller size. Together, these results redefine what a single encoder can achieve, pushing encoder pretraining beyond its presumed limits.

2 Related Work

Modernized encoders Since BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), the development of encoders has been slower than that of decoder-only LLMs. Recent work has revived encoders by bringing in innovations from LLMs, such as RoPE, GLU, and alternating local and global attention, along with trillions of training tokens (Portes et al., 2023; Nussbaum et al., 2025; Zhang et al., 2024; Warner et al., 2025; Breton et al., 2025; Weller et al., 2026). However, the pre-training recipe itself, in particular the MLM objective and its masking ratio, has been inherited from the conventions of the BERT era. As a result, each of these encoders excels on one side only, with none performing competitively across both classification and retrieval.

Masking ratio and pre-training objectives The masking ratio of MLM has been revisited in several studies, which report that ratios higher than the original 15% are beneficial, especially for larger models (Wettig et al., 2023; Ankner et al., 2024; Yang et al., 2023; Dong et al., 2025; Gisserot-Boukhlef et al., 2026a). However, these findings are based on budgets of 10B–100B tokens, where the effect of the masking ratio is entangled with convergence speed. Another line of work replaces MLM with Replaced Token Detection (RTD) (Clark et al., 2020; He et al., 2023), which is considered superior on classification tasks but is known to degrade retrieval (Rep et al., 2024; Warner et al., 2025). This comparison, however, has always been made at the conventional 15% masking, without varying the corruption level. In contrast, our work revisits both questions after reaching saturation with trillions of tokens, on both classification and retrieval tasks.

Encoders for retrieval Bidirectional encoders are also widely adopted for information retrieval. Bi-encoders are trained to encode a given text into a single vector, which is aligned to be close to those of similar texts (Reimers & Gurevych, 2019). They usually start from MLM pre-training to acquire basic linguistic knowledge, and then go through contrastive fine-tuning (Wang et al., 2022; Li et al., 2023; Nussbaum et al., 2025; Zhang et al., 2024). While the fine-tuning stage has been extensively optimized, the pre-training recipe of the backbone determines the upper bound of the resulting model. Encoders designed for retrieval, however, tend to fall behind on classification.

Knowledge distillation for language models Knowledge distillation for encoders has been proposed to match various targets: masked token reconstruction logits (Sanh et al., 2019; Jiao et al., 2020), hidden state vectors (Sanh et al., 2019; Jiao et al., 2020), self-attention scores (Jiao et al., 2020; Wang et al., 2020), and Gram matrices of attention vectors (Wang et al., 2020; 2021). Meanwhile, recent LLMs usually adopt logit-based distillation to develop small language models (Gemma Team et al., 2024; 2025; Amini et al., 2025; Meta AI, 2024). In particular, Muralidharan et al. (2024) and Liu et al. (2026) propose *Cascade Distillation*, where a small model is initialized from the pruned weights of its parent and then distilled from a stronger teacher. However, whether these approaches transfer to encoders, and how they interact with the masking ratio, remains unexplored.

3 Exploring the Pre-Training Recipe for Stronger Encoders

In this section, we examine the core design choices of MLM pre-training under controlled settings. We train encoders at three scales (150M, 380M, and 1.3B) with RoPE and alternating local and global attention, similar to Warner et al. (2025). All experiments use the same dataset mixture and tokenizer. Following the evaluation protocol of Warner et al. (2025), we fine-tune and evaluate on GLUE (Wang et al., 2018) and SQuAD 2.0 (Rajpurkar et al., 2018) for classification, and fine-tune on MS MARCO (Nguyen et al., 2016) and report the average nDCG@10 on BEIR (Thakur et al., 2021) for retrieval. Further details of the experimental setup are provided in Appendix A and B.

3.1 Training Token Scaling and Downstream Saturation

Prior work has demonstrated that decoder models continue to improve at extreme token scale (Sardana et al., 2024), and that their downstream task performance is also predictable alongside loss (Gadre et al., 2025). This raises a natural question: can encoder models also benefit from more training tokens? To answer this question, we train encoders at two scales (150M and 1.3B) on 8.8 trillion tokens, with a masking ratio of 30%. We then evaluate intermediate checkpoints on GLUE, SQuAD 2.0, and BEIR to observe how downstream performance evolves.

The results are shown in Figure 2. Encoders trained with the MLM objective saturate around 1–2 trillion tokens, while both training and validation losses keep decreasing. In particular, the BEIR scores saturate even earlier, showing no significant improvement beyond the first evaluation point of 275B tokens. These observations indicate that (1) encoders do not continue to improve on downstream tasks with more training tokens, and (2) unlike decoders, their fine-tuned downstream performance is not directly correlated with loss.

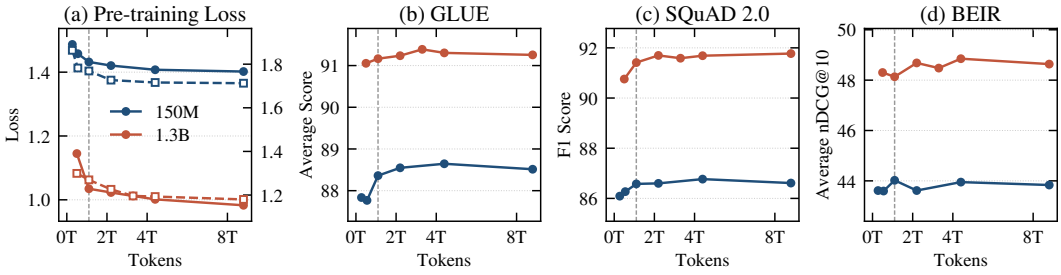


Figure 2: **Scaling the training token budget.** Pre-training loss and downstream performance of 150M and 1.3B encoders. In (a), solid lines are the training loss (left axis) and dashed lines are the validation loss on WikiText-103 (Merity et al., 2017) (right axis). (b) GLUE average score, (c) SQuAD 2.0 F1, and (d) BEIR average nDCG@10 are obtained.

We then ask whether this saturation reflects a fundamental difference between encoders and decoders. To this end, we evaluate the intermediate checkpoints of our 1.3B encoder and the SmolLM2 1.7B decoder (Ben Allal et al., 2025) on 6 benchmarks most widely used for decoders. Figure 3 shows clearly divergent trends. For base models, both the encoder and decoder improve in 5-shot accuracy, and the decoder in particular exhibits a rapid rise known as *emergent abilities* (Wei et al., 2022). However, after instruction tuning on the multiple-choice question answering (MCQA) format, the encoder saturates as on the previous downstream tasks, while the decoder keeps improving.

These results suggest that encoder models have an upper bound on downstream performance that additional pre-training tokens cannot raise. If we have enough compute budget to exceed the saturation point, how can we raise this upper bound itself? This question strongly motivates us to revisit the encoder pre-training methodology.

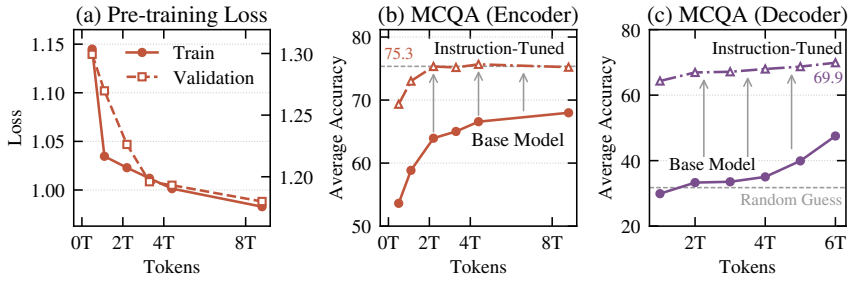


Figure 3: **Accuracy before and after instruction tuning.** (a) Pre-training loss of the 1.3B encoder. (b) Its MCQA average accuracy as a 5-shot base model (solid) and after instruction tuning (dash-dotted). (c) The same comparison for a 1.7B decoder; the horizontal line is the random baseline.

3.2 Masking Ratios and Classification Tasks

The MLM objective differs from the causal LM objective in one respect: the masking ratio. Its impact has been investigated in various contexts (Dong et al., 2025; Yang et al., 2023; Wettig et al., 2023; Gisserot-Boukhlef et al., 2026a; Boizard et al., 2025), yet its effect has been entangled with convergence speed. In this section, we revisit it from the perspective of downstream classification tasks, but after reaching saturation, where this entanglement no longer applies.

In detail, we train encoders at three scales with masking ratios up to 50%. As lower ratios require more tokens to converge (Wettig et al., 2023; Ankner et al., 2024), we allocate tokens accordingly[1] based on the saturation points identified in Section 3.1 and evaluate on GLUE and SQuAD 2.0.

Figure 4 shows that classification performance consistently degrades as the masking ratio increases, across all model sizes. This is contrary to prior work (Wettig et al., 2023; Gisserot-Boukhlef et al., 2026a; Dong et al., 2025), which finds larger masking ratios optimal for larger models. We attribute this discrepancy to eliminating the effect of convergence speed by using trillions of tokens, whereas prior results were confined to 10B–100B budgets. Notably, Wettig et al. (2023) argued that higher ratios are preferable since the gain from faster convergence outweighs the cost of corruption, treating the benefit of less corruption only briefly. At saturation, however, we find the limitation from corruption to be far more pronounced.

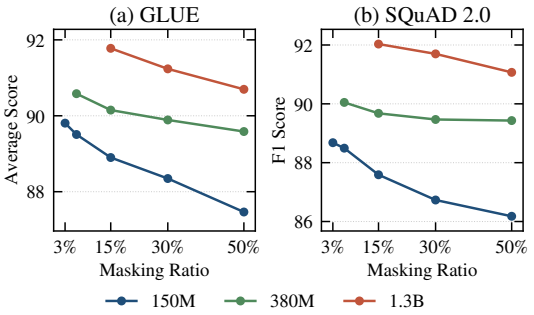


Figure 4: (a) GLUE average score and (b) SQuAD 2.0 F1 against the pretraining masking ratio, for encoders of 150M, 380M and 1.3B scales. Both classification tasks degrade consistently as the masking ratio increases.

Impact of contextual corruption To understand the effect of corruption on encoders, we further compare MLM with RTD. Conventionally, ELECTRA-style models are considered to be superior on classification tasks (Clark et al., 2020; He et al., 2023), and they remain so even against recent modernized encoders trained with MLM (Antoun et al., 2025), though at a critical cost on retrieval (Rep et al., 2024; Warner et al., 2025).

The superiority of RTD is often explained by its higher training efficiency (Clark et al., 2020). Training to saturation factors out this efficiency, allowing us to isolate what makes RTD superior beyond faster convergence. One remaining difference lies in *contextual corruption*, i.e., how contextually plausible the corrupted tokens are. In RTD, a generator fills masked tokens with semantically rel-

[1]We use 1T tokens at 50% and 30%, 2T tokens at 15%, and 5T tokens at 6% and 3%. Empirically, saturation occurred around 1–2T tokens at 15% and 2–3T tokens at 6% and 3%, so these budgets are sufficient.

evant alternatives, which are more informative than the [MASK] token. We hypothesize that this difference mainly explains the superiority.

We train 150M discriminators, with generators of half the depth, following He et al. (2023). Since the actual replacement ratio changes during training as the generator improves, we adjust the generator's masking ratio so that the average replacement ratio matches that of MLM. We report the average score of GLUE and SQuAD 2.0, excluding CoLA.[2]

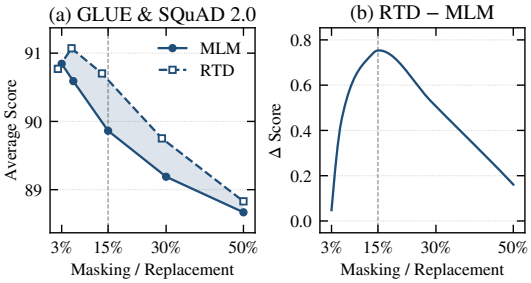


Figure 5: MLM and RTD at matched corruption levels, for 150M encoders. (a) Average of GLUE without CoLA and SQuAD 2.0 F1 score. (b) The gap between interpolated curves. The dashed vertical line marks 15%.

Figure 5 shows the improvement of RTD over MLM. As the corruption ratios of the two objectives are not exactly the same, Figure 5(b) is computed by interpolating both curves. The gap peaks around 15%, the masking ratio adopted by Clark et al. (2020) and He et al. (2023), and diminishes toward both ends for different reasons. Around 50%, the generator has too little context, resulting in trivial replacements, which are known to bring no gain over MLM (Di Liello et al., 2022). Around 3%, the difference between plausible alternatives and [MASK] tokens becomes negligible, as the absolute corruption level is small. In practice, since classification favors low masking ratios in the first place, encoders can retain MLM without sacrificing classification, avoiding the retrieval degradation that RTD entails.

3.3 Masking Ratios and Retrieval Tasks

Lower masking ratios favor classification, but it is unclear whether the same holds for downstream retrieval tasks, the other major application of bidirectional encoders. Beyond the classification tasks, we now examine retrieval, reporting the average nDCG@10 on BEIR as the downstream performance. Figure 6(a) shows the opposite trend to classification. The masking ratio of 50% consistently achieves the highest score across all model scales, which is also in line with a few prior works (Boizard et al., 2025; Gisserot-Boukhlef et al., 2026a). Overall, no single masking ratio is optimal for both classification and retrieval.

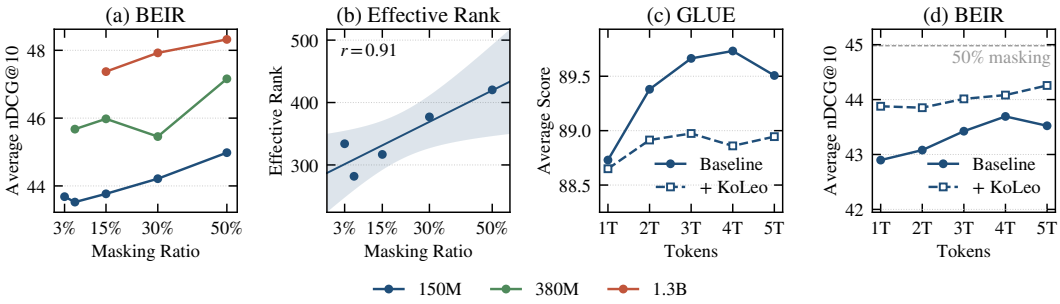


Figure 6: **Masking ratios and retrieval.** (a) BEIR average nDCG@10 against the masking ratio for 150M, 380M, and 1.3B encoders. (b) Effective rank of the 150M model against the masking ratio. (c) GLUE average score and (d) BEIR average nDCG@10 of the 150M encoder pre-trained at 6% masking with (dashed) and without (solid) the KoLeo regularizer.

To understand why higher masking ratios benefit retrieval, we look into what changes in the representation space. We measure the effective rank (Roy & Vetterli, 2007) of the mean-pooled vectors of each trained model. The effective rank reflects how many dimensions in the representation space the model actually uses. As shown in Figure 6(b), the effective rank tends to increase with the masking ratio, indicating that higher masking ratios lead to higher utilization of the representation space.

[2]We discuss CoLA as a confounder in Appendix C.1.

We cannot tell whether the higher effective rank causes the improvement in retrieval, or whether the two are independent phenomena. Since using more dimensions is generally favorable for retrieval (Kusupati et al., 2022; Rottach et al., 2025), we ask whether the effective rank can be raised by regularization instead. As we need to use a lower masking ratio for classification, we train the encoder with 6% masking but try adopting the KoLeo regularizer (Sablayrolles et al., 2019), which is used for representation learning (Oquab et al., 2024; Siméoni et al., 2026). This encourages the vectors to spread isotropically, resulting in a higher effective rank (Zhang et al., 2025a).[3]

Figure 6(c) and (d) compare the performances with and without the KoLeo regularizer on GLUE and BEIR, respectively. Raising the effective rank artificially cannot resolve the trade-off at hand: (1) the regularizer indeed improves BEIR but less than single 50% masking, and (2) it also harms the GLUE score. From this experiment, we confirm the strong contrast between classification and retrieval across masking ratios, and the limitation of using a single masking ratio.

3.4 Mixing Masking Ratios with LM Head Decoupling

From the previous sections, we observed that different masking ratios specialize encoders for different tasks, and that regularization cannot naturally resolve the trade-off. Since the two tasks favor opposite masking ratios, achieving the best of both without compromise can be viewed as multi-task learning (MTL). Under this view, we instead consider mixing multiple masking ratios within a batch, treating each ratio as a task and jointly optimizing their losses.

MTL frequently faces conflicts across tasks due to parameter sharing (Sener & Koltun, 2018; Yu et al., 2020). He et al. (2023) claimed that sharing input embeddings between the generator and discriminator leads to tug-of-war dynamics, where the two disturb each other, and resolved this with gradient-level disentanglement. Similarly, we observed the divergence of the encoders' representation spaces through their effective rank in Section 3.3. We thus visualize the input embeddings and LM heads to look for such a conflict.

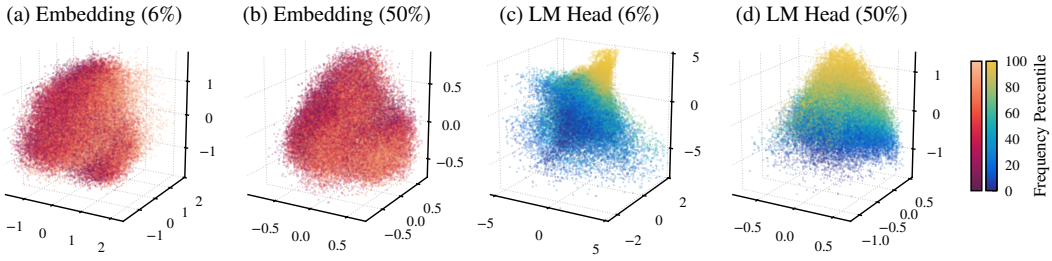


Figure 7: **Geometry of input embeddings and LM heads.** First three principal components of the input embedding matrix (a, b) and the LM head (c, d) of 150M encoders pre-trained at 6% and 50% masking. Each point is one vocabulary entry, colored by its frequency percentile in the corpus.

Figure 7 shows the geometry of both matrices for encoders trained with 6% and 50% masking. As illustrated, the LM heads take clearly different shapes, while the input embeddings are almost identical. The discrepancy is also observed in Figure 8, which plots the singular value spectra of the embeddings and LM heads. The LM head for 6% masking shows more concentrated singular values, indicating that the matrix occupies a lower-dimensional subspace. This is consistent with the model's lower effective rank. The encoder's gradients are computed from the loss through its LM head, so we conjecture that sharing it may cause gradient-level interference, and test decoupling the two heads.

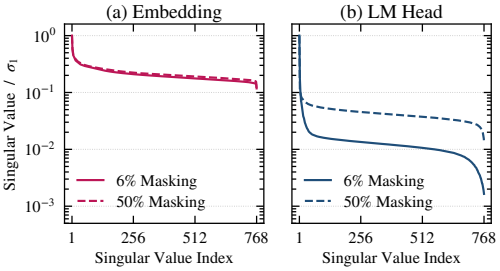


Figure 8: Relative singular values of (a) the input embedding matrix and (b) the LM head, for 150M encoders pre-trained at 6% (solid) and 50% (dashed) masking. Each spectrum is divided by its own leading singular value.

[3]The relation between the KoLeo regularizer and the effective rank is explained in Appendix D.

We train encoders with various masking ratio mixing strategies, which have been explored in preliminary forms with different motivations: (1) decaying the ratio from 50% to 6% and 15% (Ankner et al., 2024; Yang et al., 2023), and (2) sampling the ratio uniformly from 5% to 50% (Dong et al., 2025), denoted as $\mathcal{U}(5\%, 50\%)$. Since performance on each task monotonically favors the opposite extreme, we further consider (3) combining 5% and 50% masked sequences within a batch, denoted as $\{5\%, 50\%\}$, with and without LM head decoupling. Note that 5% masking yields sparser gradient signals and converges more slowly than 50%, so we include four times more sequences masked at 5% in the batch.[4] We evaluate all models on GLUE and BEIR.

Table 1 shows the results. Masking ratio decay brings no improvement over the single masking ratio on GLUE, and even harms it at 6% (88.8 vs 89.5). BEIR improves slightly, as the encoder is exposed to the larger ratio, yet still far from 45.0 of 50%. Either the early high corruption limits the upper bound, or the low-ratio phase needs to be much longer, which defeats the purpose of decay. Meanwhile, $\mathcal{U}(5\%, 50\%)$, whose average ratio is 27.5%, performs almost the same as the single 30% (88.0 vs 88.3 on GLUE, both 44.2 on BEIR). Figure 9(a) and (b) also show that its LM head geometry is close to that of 30%. Mixing ratios over a range thus converges to the effect of the average.

In contrast, $\{5\%, 50\%\}$ with decoupled LM heads achieves 89.4 on GLUE and 45.1 on BEIR, reaching the best scores of 6% (89.5) and 50% (45.0). Without decoupling, GLUE stays at 88.9, on par with the single 15%, while BEIR still improves from 43.8 to 44.7. Although exposure to the large masking ratio alone helps retrieval, these results suggest that sharing the LM head creates a conflict between masking regimes, which head decoupling alleviates. Figure 9(c) and (d) are consistent with this, where the decoupled heads for 5% and 50% take distinct geometries.

Masking Ratio	GLUE	BEIR
Single Ratio		
6%	**89.5**	43.5
15%	88.9	43.8
30%	88.3	44.2
50%	87.5	**45.0**
Decaying Ratio		
$50\% \rightarrow 6\%$	88.8	43.9
$50\% \rightarrow 15\%$	88.6	44.1
Mixing Ratios		
$\mathcal{U}(5\%, 50\%)$	88.0	44.2
$\{5\%, 50\%\}$	88.9	44.7
+ Decoupling	**89.4**	**45.1**

Table 1: Comparison of masking ratio mixing for the 150M encoder. Decoupling denotes separate LM heads for each masking ratio.

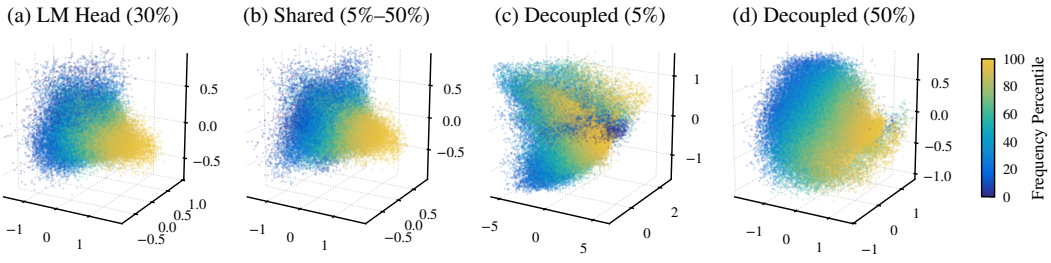


Figure 9: **LM head geometry under masking ratio mixing.** First three principal components of the LM head, with each point a vocabulary entry colored by its frequency percentile. (a) An encoder pre-trained at a single 30% masking ratio, (b) the shared head of $\mathcal{U}(5\%, 50\%)$, and (c, d) the decoupled heads of $\{5\%, 50\%\}$ for 5% and 50% masking, respectively.

3.5 Revisiting Pruning and Distillation

We resolve the trade-off between classification and retrieval with LM head decoupling, but each is still bounded by what the model can reach on its own. To go beyond the limit, we revisit pruning and distillation, as in Sanh et al. (2019), Muralidharan et al. (2024), and Liu et al. (2026), under different masking ratios.

For distillation, we restrict ourselves to logit-based distillation, which is still widely used in recent LLMs. Attention-based distillation, including self-attention scores (Jiao et al., 2020; Wang et al., 2020) and Gram matrices (Wang et al., 2020; 2021), requires $O(L^2)$ memory and computational costs for length L. These are unaffordable for long contexts and incompatible with FlashAttention.

[4] Contrary to common expectation, we found the convergence speed not proportional to the masking ratio, though lower ratios still converge slower.

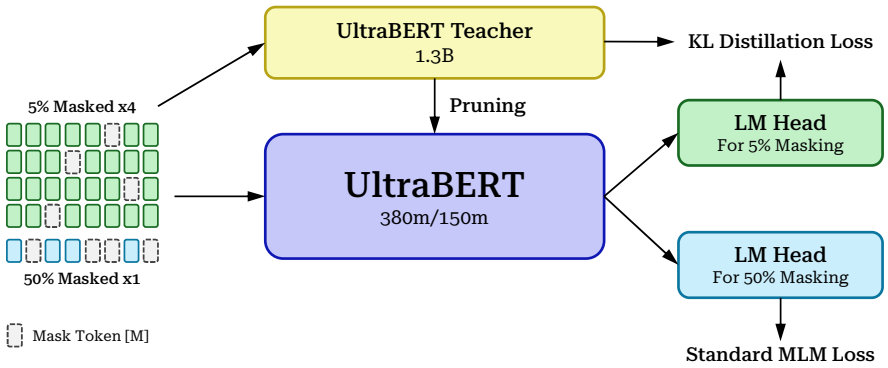


Figure 10: **Overview of UltraBERT.** Starting from the 1.3B encoder as a teacher, we prune it to 380M and then further to 150M. Each batch mixes 5% and 50% masked sequences at a 4:1 ratio, and each ratio has its own LM head. The 5% head is trained with the KL distillation loss from the 1.3B teacher, while the 50% head is trained with the standard MLM loss.

All experiments are conducted at 15% and 50% masking. We first prune the trained 1.3B encoders to 380M following Muralidharan et al. (2024) and Liu et al. (2026), and distill each from its 1.3B parent on 500B tokens.[5] We then evaluate on GLUE and BEIR, comparing with the 380M models trained from scratch. Results are presented in Table 2. For 15% masking, the distilled model indeed improves on GLUE (90.2 to 90.8), consistent with lower ratios benefiting classification, while BEIR remains the same. In contrast, the model distilled at 50% masking performs worse than the from-scratch model on BEIR (47.2 to 46.5). This is somewhat surprising, since we expected distillation to help the corresponding task. This degradation may come from the trivial signal of the teacher at 50% masking, similar to the interpretation of RTD in Section 3.2. Regardless of the underlying cause, pruning and distillation yield no gains over training from scratch at 50% masking, motivating us to apply distillation only at the low ratio.

Model	GLUE	BEIR
15% Masking		
1.3B	91.8	47.4
380M	90.2	46.0
380M (Distilled)	90.8	46.0
50% Masking		
1.3B	90.7	48.3
380M	89.6	47.2
380M (Distilled)	89.6	46.5

Table 2: Logit-based distillation from 1.3B to 380M at 15% and 50% ratios.

4 UltraBERT: Beyond the Limit of BERT Pre-Training

Building on these findings, we introduce **UltraBERT**, a family of encoders in two sizes (150M and 380M) with 32K context that pushes the limit of existing encoders on both classification and retrieval without trade-off. Following Muralidharan et al. (2024) and Liu et al. (2026), we start from the 1.3B encoder as a teacher, prune it to 380M and then further to 150M, and distill both from the 1.3B teacher. Based on Section 3.4, we construct each batch with both 5% and 50% masking at a 4:1 ratio, with decoupled LM heads. Following Section 3.5, we apply KL divergence loss only to the 5% masked sequences, while the 50% ones are trained with the standard MLM loss. Figure 10 illustrates the framework, and the training setup follows Section 3 with the remaining details in Appendix A.

4.1 Evaluation Setup

To evaluate classification from various perspectives, we report GLUE and SQuAD 2.0 as in the ablation study. Furthermore, we include the English subset of MultiCoNER v2 (Fetahu et al., 2023) and Few-NERD (Ding et al., 2021) for the harder NER tasks. We also evaluate on RewardBench (Lambert et al., 2025) as a pairwise preference classification task. Each benchmark is fine-tuned with hyperparameter grid search on its training set (see Appendix B).

For the retrieval tasks, we follow Section 3 to report the BEIR score. Using the best model on BEIR, we report the code and math subsets of RAR-b (Xiao et al., 2024) to estimate generalizability in the code and math domains. To evaluate long context ability, we further fine-tune the best

[5] As in prior work, distillation converges faster; we observe saturation at 250B tokens for 15% masking.

Model	# Params	Classification					Retrieval				
		GLUE	SQuAD 2.0	Multi-CoNER v2	Few-NERD	Reward Bench	BEIR	RAR-b Code	RAR-b Math	MLDR	Long Embed
Base (≤200M)											
BERT	110M	84.7	76.3	53.8	66.6	59.4	36.7	6.6	27.5	29.0	29.3
RoBERTa	125M	86.4	83.7	56.5	67.1	63.2	39.6	11.7	33.4	28.7	33.8
MosaicBERT	137M	85.4	79.4	57.8	65.7	62.5	41.4	12.7	30.9	29.6	40.5
NomicBERT	137M	84.0	82.7	**60.1**	66.9	64.1	42.0	13.8	30.2	29.9	45.0
GTE-en-MLM	137M	85.6	82.7	59.2	66.4	63.4	42.4	12.0	33.6	37.1	58.5
ModernBERT	149M	88.4	86.7	57.4	66.7	68.0	42.6	29.6	46.1	37.7	63.2
Ettin Encoder	149M	88.9	87.4	57.4	66.5	70.3	42.4	31.8	47.1	35.9	60.1
UltraBERT	152M	**89.5**	**89.2**	57.9	**67.4**	**73.0**	**45.2**	**39.1**	**55.4**	**43.5**	**72.0**
Medium (≤500M)											
BERT	334M	85.2	81.8	57.7	67.8	62.6	38.6	12.2	33.0	29.8	30.0
RoBERTa	355M	88.9	89.4	62.3	68.1	68.5	42.8	20.3	37.8	29.5	32.2
GTE-en-MLM	434M	87.6	87.2	63.0	67.1	62.2	39.4	13.5	32.0	40.8	57.0
NeoBERT	222M	89.0	86.6	57.4	66.4	57.6	42.9	17.3	33.4	21.7	32.4
LFM2.5-Encoder	230M	88.2	86.1	59.6	67.6	67.5	46.7	30.9	50.2	43.7	70.0
LFM2.5-Encoder	355M	89.2	87.2	62.7	67.9	68.5	47.3	38.6	50.2	**44.5**	71.7
ModernBERT	395M	90.4	89.8	60.9	67.6	72.8	45.2	34.0	46.1	39.9	63.5
Ettin Encoder	395M	90.8	90.4	62.3	67.8	76.7	45.9	38.5	54.0	40.9	64.3
UltraBERT	380M	**91.0**	**90.9**	**63.5**	**68.3**	**78.8**	**47.7**	**47.0**	**61.7**	44.1	**74.9**

Table 3: Comparison with existing encoders on classification and retrieval benchmarks. Parameter counts exclude the LM head. Best scores in each group are in bold and second best are underlined.

model on the English subset of MLDR (Chen et al., 2024) and report its test set score, alongside its LongEmbed (Zhu et al., 2024) score.

For all tasks, we compare with various model families as baselines: BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), MosaicBERT (Portes et al., 2023), NomicBERT (Nussbaum et al., 2025), GTE-en-MLM (Zhang et al., 2024), ModernBERT (Warner et al., 2025), Ettin Encoder (Weller et al., 2026), NeoBERT (Breton et al., 2025), and LFM2.5-Encoder (Liquid AI, 2026).

4.2 Results

As shown in Table 3, UltraBERT achieves state-of-the-art performance on almost all benchmarks. For the base size, UltraBERT achieves the best score on all benchmarks except MultiCoNER v2, where it still outperforms both ModernBERT and Ettin Encoder. For the medium size, UltraBERT achieves the best on all benchmarks except MLDR, where it is behind LFM2.5-Encoder-350M.

As the UltraBERT series natively supports 32K context length, both scales achieve the best score on LongEmbed, where even the base model outperforms the medium models of other families. Furthermore, the largest improvements are on code and math retrieval, with gains of 7.3 and 8.3 points at base and 8.4 and 7.7 points at medium over the second best, respectively.

Most importantly, previous encoders each excel in different areas. On GLUE, SQuAD 2.0, and RewardBench, Ettin Encoder was the best among prior models. On the NER tasks, NomicBERT, RoBERTa, and GTE-en-MLM led. On BEIR, ModernBERT was the strongest base model and LFM2.5-Encoder-350M the strongest medium model. None of them, however, reaches the top across the full range of benchmarks. UltraBERT is the first to do so at both scales, indicating that our recipe yields a truly general encoder rather than one specialized for a single task family.

5 Conclusion

We revisit the pre-training recipe of bidirectional encoders at the scale of trillions of tokens. Encoders saturate on downstream tasks at a few trillion tokens, beyond which the masking ratio determines their specialization: lower ratios favor classification, higher ratios favor retrieval. We reconcile the two by mixing both ratios within a batch with decoupled LM heads, and by applying logit-based distillation only at the low ratio. Built on these findings, UltraBERT achieves state-of-the-art results across classification and retrieval at both 150M and 380M with a 32K context, showing that the specialization of encoders is not inherent but a consequence of the recipe.

AI USE STATEMENT

In this work, we used generative AI tools to assist in writing the proof in Appendix D, where the structure of the argument was designed by the authors and the AI assisted in formalizing the individual steps. We have not used generative AI tools to develop the conceptual framework, propose hypotheses, design experiments, implement methods, or interpret results; all training and evaluation code was written by the authors. The remaining tasks with required disclosure are not applicable to this work. Additionally, we used generative AI tools to edit the paper for readability, including converting result logs into LaTeX tables, to summarize and identify related literature, and to search for information. We have reviewed all AI-assisted work: the proof in Appendix D was checked by the authors, all numbers in the tables were verified against the original logs, and all claims about prior work were checked against the original papers. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work pre-trains and evaluates general-purpose text encoders on publicly available datasets. The pre-training mixture consists of web, code, academic, and instruction data that are publicly released under their respective licenses, and we did not collect data from human subjects. As with any model trained on web-scale data, the encoders may inherit biases present in the training data, and downstream users should evaluate them for their use case. We do not foresee ethical concerns specific to this work beyond those common to encoder pre-training.

REPRODUCIBILITY STATEMENT

We describe the architecture, optimization, pruning, and distillation in Appendix A, the pre-training data mixture with per-dataset token counts in Appendix A.3, and the fine-tuning and evaluation protocols for every benchmark, including hyperparameter grids and data sources, in Appendix B. The proof of the relation between the KoLeo regularizer and effective rank is given in Appendix D. Full per-task results are reported in Appendix E, along with the sources of baseline scores taken from the literature. The evaluation code used for all benchmarks is included in the supplementary material. Upon acceptance, we will release the model weights and the pre-training data mixture.

REFERENCES

Alexander Amini, Anna Banaszak, Harold Benoit, Arthur Böök, Tarek Dakhran, Song Duong, Alfred Eng, Fernando Fernandes, Marc Härkönen, Anne Harrington, et al. Lfm2 technical report. *arXiv preprint arXiv:2511.23404*, 2025.

Zachary Ankner, Naomi Saphra, Davis Blalock, Jonathan Frankle, and Matthew Leavitt. Dynamic masking rate schedules for MLM pretraining. In Yvette Graham and Matthew Purver (eds.), *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 477–487, St. Julian's, Malta, March 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.eacl-short.42. URL https://aclanthology.org/2024.eacl-short.42/.

Wissam Antoun, Benoît Sagot, and Djamé Seddah. ModernBERT or DeBERTaV3? examining architecture and data influence on transformer encoder models performance. In Kentaro Inui, Sakriani Sakti, Haofen Wang, Derek F. Wong, Pushpak Bhattacharyya, Biplab Banerjee, Asif Ekbal, Tanmoy Chakraborty, and Dhirendra Pratap Singh (eds.), *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 3061–3074, Mumbai, India, December 2025. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics. ISBN 979-8-89176-298-5. doi: 10.18653/v1/2025.ijcnlp-long.164. URL https://aclanthology.org/2025.ijcnlp-long.164/.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Jan Beirlant, E. Dudewicz, L. Gyor, and E.C. Meulen. Nonparametric entropy estimation: An overview. *International Journal of Mathematical and Statistical Sciences*, 6, 01 1997.

Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. Smollm-corpus, 2024. URL https://huggingface.co/datasets/HuggingFaceTB/smollm-corpus.

Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martin Blazquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Agustín Piqueres Lajarín, Hynek Kydlíček, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan Son NGUYEN, Ben Burtenshaw, Clémentine Fourrier, Haojun Zhao, Hugo Larcher, Mathieu Morlon, Cyril Zakka, Colin Raffel, Leandro Von Werra, and Thomas Wolf. SmolLM2: When smol goes big — data-centric training of a fully open small language model. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=3JiCl2A14H.

Nicolas Boizard, Hippolyte Gisserot-Boukhlef, Duarte Miguel Alves, Andre Martins, Ayoub Hammal, Caio Corro, CELINE HUDELOT, Emmanuel Malherbe, Etienne Malaboeuf, Fanny Jourdan, Gabriel Hautreux, João Alves, Kevin El Haddad, Manuel Faysse, Maxime Peyrard, Nuno M Guerreiro, Patrick Fernandes, Ricardo Rei, and Pierre Colombo. EuroBERT: Scaling multilingual encoders for european languages. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=jdOC24msVq.

Lola Le Breton, Quentin Fournier, John Xavier Morris, Mariam El Mezouar, and Sarath Chandar. NeoBERT: A next generation BERT. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=TJRyDi7mwH. Reproducibility Certification.

Martin Juan José Bucher and Marco Martini. Fine-tuned 'small' llms (still) significantly outperform zero-shot generative ai models in text classification. *arXiv preprint arXiv:2406.08660*, 2024.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2318–2335, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.137. URL https://aclanthology.org/2024.findings-acl.137/.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300. URL https://aclanthology.org/N19-1300/.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1xMH1BtvB.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Thomas M. Cover and Joy A. Thomas. *Information Theory and Statistics*, chapter 11, pp. 347–408. John Wiley Sons, Ltd, 2005. ISBN 9780471748823. doi: https://doi.org/10.1002/047174882X.ch11. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/047174882X.ch11.

Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. ULTRAFEEDBACK: Boosting language models with scaled AI feedback. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 9722–9744. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/cui24f.html.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=mZn2Xyh9Ec.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

Luca Di Liello, Matteo Gabburo, and Alessandro Moschitti. Efficient pre-training objectives for transformers. *arXiv preprint arXiv:2104.09694*, 2021.

Luca Di Liello, Matteo Gabburo, and Alessandro Moschitti. Effective pretraining objectives for transformer-based autoencoders. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 5533–5547, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.405. URL https://aclanthology.org/2022.findings-emnlp.405/.

Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. Few-NERD: A few-shot named entity recognition dataset. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *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)*, pp. 3198–3213, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.248. URL https://aclanthology.org/2021.acl-long.248/.

Mingze Dong, Leda Wang, and Yuval Kluger. Understanding and enhancing mask-based pretraining towards universal representations. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 142696–142739. Curran Associates, Inc., 2025. doi: 10.52202/085713-4777. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/d202099e9d671859246a0a78b66117d7-Paper-Conference.pdf.

Aleksandra Edwards and Jose Camacho-Collados. Language models for text classification: Is in-context learning enough? In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue (eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 10058–10072, Torino, Italia, May 2024. ELRA and ICCL. URL https://aclanthology.org/2024.lrec-main.879/.

Besnik Fetahu, Zhiyu Chen, Sudipta Kar, Oleg Rokhlenko, and Shervin Malmasi. MultiCoNER v2: a large multilingual dataset for fine-grained and noisy named entity recognition. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 2027–2051, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.134. URL https://aclanthology.org/2023.findings-emnlp.134/.

Samir Yitzhak Gadre, Georgios Smyrnis, Vaishaal Shankar, Suchin Gururangan, Mitchell Wortsman, Rulin Shao, Jean Mercat, Alex Fang, Jeffrey Li, Sedrick Keh, Rui Xin, Marianna Nezhurina, Igor Vasiljevic, Luca Soldaini, Jenia Jitsev, Alex Dimakis, Gabriel Ilharco, Pang Wei Koh,

Shuran Song, Thomas Kollar, Yair Carmon, Achal Dave, Reinhard Heckel, Niklas Muennighoff, and Ludwig Schmidt. Language models scale reliably with over-training and on downstream tasks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=iZeQBqJamf.

Luyu Gao, Yunyi Zhang, Jiawei Han, and Jamie Callan. Scaling deep contrastive learning batch size under memory limited setup. In Anna Rogers, Iacer Calixto, Ivan Vulić, Naomi Saphra, Nora Kassner, Oana-Maria Camburu, Trapit Bansal, and Vered Shwartz (eds.), *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pp. 316–321, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.repl4nlp-1.31. URL https://aclanthology.org/2021.repl4nlp-1.31/.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. How to train long-context language models (effectively). In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7376–7399, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.366. URL https://aclanthology.org/2025.acl-long.366/.

Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann Lecun. RankMe: Assessing the downstream performance of pretrained self-supervised representations by their rank. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10929–10974. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/garrido23a.html.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Hippolyte Gisserot-Boukhlef, Nicolas Boizard, Manuel Faysse, Duarte Miguel Alves, Emmanuel Malherbe, Andre Martins, CELINE HUDELOT, and Pierre Colombo. Should we still pretrain encoders with masked language modeling? In *The Fourteenth International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=jpz7e3jhRq.

Hippolyte Gisserot-Boukhlef, Nicolas Boizard, Emmanuel Malherbe, CELINE HUDELOT, and Pierre Colombo. BERT-as-a-judge: A robust alternative to lexical methods for efficient reference-based LLM evaluation. In *Third Conference on Language Modeling*, 2026b. URL https://openreview.net/forum?id=gKNb0vxOTB.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=sE7-XhLxHA.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=d7KBjmI3GmQ.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021b. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/be83ab3ecd0db773eb2dc1b0a17836a1-Paper-round2.pdf.

Shengding Hu, Yuge Tu, Xu Han, Ganqu Cui, Chaoqun He, Weilin Zhao, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Xinrong Zhang, Zhen Leng Thai, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, dahai li, Zhiyuan Liu, and Maosong Sun. MiniCPM: Unveiling the potential of small language models with scalable training strategies. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=3X2L2TFr0f.

Hamel Husain, Ho-Hsiang Wu, Tiferet Gazit, Miltiadis Allamanis, and Marc Brockschmidt. Codesearchnet challenge: Evaluating the state of semantic code search. *arXiv preprint arXiv:1909.09436*, 2019.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. TinyBERT: Distilling BERT for natural language understanding. In Trevor Cohn, Yulan He, and Yang Liu (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4163–4174, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.372. URL https://aclanthology.org/2020.findings-emnlp.372/.

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, and Ali Farhadi. Matryoshka representation learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 30233–30249. Curran Associates, Inc., 2022. doi: 10.52202/068431-2192. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/c32319f4868da7613d78af9993100e42-Paper-Conference.pdf.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. RewardBench: Evaluating reward models for language modeling. In Luis Chiruzzo, Alan Ritter, and Lu Wang (eds.), *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 1755–1797, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-195-7. doi: 10.18653/v1/2025.findings-naacl.96. URL https://aclanthology.org/2025.findings-naacl.96/.

Guillaume Letellier, Antonin Vacheret, and Frédéric Jurie. Jetparticle-jepa: An efficient self-supervised representation learning method for jet tagging in high-energy physics. *arXiv preprint arXiv:2606.14813*, 2026.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Josh Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Chandu, Thao Nguyen, Igor Vasiljevic, Sham Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alexandros G. Dimakis, Yair Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. Datacomp-lm: In search of the next generation of training sets for language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 14200–14282. Curran Associates, Inc., 2024. doi: 10.52202/079017-0455. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/19e4ea30dded58259665db375885e412-Paper-Datasets_and_Benchmarks_Track.pdf.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

Liquid AI. Lfm2.5-encoders: Fast at long context, even on cpu. *Liquid AI Blog*, 2026. www.liquid.ai/blog/lfm2-5-encoders.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, et al. Ministral 3. *arXiv preprint arXiv:2601.08584*, 2026.

Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. Skywork-reward: Bag of tricks for reward modeling in llms. *arXiv preprint arXiv:2410.18451*, 2024.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, and Adam Roberts. The flan collection: Designing data and methods for effective instruction tuning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 22631–22648. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/longpre23a.html.

Marc Marone, Orion Weller, William Fleshman, Eugene Yang, Dawn Lawrie, and Benjamin Van Durme. mmbert: A modern multilingual encoder with annealed language learning. *arXiv preprint arXiv:2509.06888*, 2025.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Byj72udxe.

Meta AI. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models, 2024. URL https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. MTEB: Massive text embedding benchmark. In Andreas Vlachos and Isabelle Augenstein (eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2014–2037, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.148. URL https://aclanthology.org/2023.eacl-main.148/.

Niklas Muennighoff, Qian Liu, Armel Randy Zebaze, Qinkai Zheng, Binyuan Hui, Terry Yue Zhuo, Swayam Singh, Xiangru Tang, Leandro Von Werra, and Shayne Longpre. Octopack: Instruction tuning code large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=mw1PWNSWZP.

Saurav Muralidharan, Sharath Turuvekere Sreenivas, Raviraj Joshi, Marcin Chochowski, Mostofa Patwary, Mohammad Shoeybi, Bryan Catanzaro, Jan Kautz, and Pavlo Molchanov. Compact language models via pruning and knowledge distillation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing*

Systems, volume 37, pp. 41076–41102. Curran Associates, Inc., 2024. doi: 10.52202/079017-1299. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/4822991365c962105b1b95b1107d30e5-Paper-Conference.pdf.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016. URL http://arxiv.org/abs/1611.09268.

Zach Nussbaum, John Xavier Morris, Andriy Mulyar, and Brandon Duderstadt. Nomic embed: Training a reproducible long context text embedder. *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=IPmzyQSiQE. Reproducibility Certification.

OLMo Team, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt. Featured Certification.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 30811–30849. Curran Associates, Inc., 2024. doi: 10.52202/079017-0970. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/370df50ccfdf8bde18f8f9c2d9151bda-Paper-Datasets_and_Benchmarks_Track.pdf.

Jacob Portes, Alexander Trott, Sam Havens, DANIEL KING, Abhinav Venigalla, Moin Nadeem, Nikhil Sardana, Daya Khudia, and Jonathan Frankle. Mosaicbert: A bidirectional encoder optimized for fast pretraining. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 3106–3130. Curran Associates, Inc., 2023. doi: 10.52202/075280-0137. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/095a6917768712b7ccc61acbeecad1d8-Paper-Conference.pdf.

Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In Iryna Gurevych and Yusuke Miyao (eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2124. URL https://aclanthology.org/P18-2124/.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

Ivan Rep, David Dukić, and Jan Šnajder. Are ELECTRA's sentence embeddings beyond repair? the case of semantic textual similarity. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 9159–9169, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.535. URL https://aclanthology.org/2024.findings-emnlp.535/.

Florian Rottach, William Rudman, Bastian Rieck, Harrisen Scells, and Carsten Eickhoff. From topology to retrieval: Decoding embedding spaces with unified signatures. *arXiv preprint arXiv:2511.22150*, 2025.

Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. *2007 15th European Signal Processing Conference*, pp. 606–610, 2007. URL https://api.semanticscholar.org/CorpusID:12184201.

Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SkGuG2R5tm.

David Samuel. Berts are generative in-context learners. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 2558–2589. Curran Associates, Inc., 2024. doi: 10.52202/079017-0084. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/04ea184dfb5f1babb78c093e850a83f9-Paper-Conference.pdf.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social IQa: Commonsense reasoning about social interactions. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4463–4473, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1454. URL https://aclanthology.org/D19-1454/.

Nikhil Sardana, Jacob Portes, Sasha Doubov, and Jonathan Frankle. Beyond chinchilla-optimal: Accounting for inference in language model scaling laws. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 43445–43460. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/sardana24a.html.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/432aca3a1e345e339f35a30c8f65edce-Paper.pdf.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seung Eun Yi, Michael Ramamonjisoa, Francisco Massa, Daniel HAZIZA, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Herve Jegou, Patrick Labatut, and Piotr Bojanowski. DINOv3. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=2NlGyqNjns. Featured Certification.

Luca Soldaini and Kyle Lo. peS2o (Pretraining Efficiently on S2ORC) Dataset. Technical report, Allen Institute for AI, 2023. ODC-By, https://github.com/allenai/pes2o.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya Jha, Sachin Kumar, Li Lucy, Xinxi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Abhilasha Ravichander, Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Evan Walsh, Luke Zettlemoyer, Noah Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse Dodge, and Kyle Lo. Dolma: an open corpus of three trillion tokens for language model pretraining

research. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 15725–15788, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.840. URL https://aclanthology.org/2024.acl-long.840/.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom.2023.127063. URL https://www.sciencedirect.com/science/article/pii/S0925231223011864.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *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)*, pp. 4149–4158, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1421. URL https://aclanthology.org/N19-1421/.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/65b9eea6e1cc6bb9f0cd2a47751a186f-Paper-round2.pdf.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Tal Linzen, Grzegorz Chrupała, and Afra Alishahi (eds.), *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5446. URL https://aclanthology.org/W18-5446/.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5776–5788. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. MiniLMv2: Multi-head self-attention relation distillation for compressing pretrained transformers. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (eds.), *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 2140–2151, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.188. URL https://aclanthology.org/2021.findings-acl.188/.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2526–2547, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.127. URL https://aclanthology.org/2025.acl-long.127/.

Maurice Weber, Daniel Y. Fu, Quentin Anthony, Yonatan Oren, Shane Adams, Anton Alexandrov, Xiaozhong Lyu, Huu Nguyen, Xiaozhe Yao, Virginia Adams, Ben Athiwaratkun, Rahul Chalamala, Kezhen Chen, Max Ryabinin, Tri Dao, Percy Liang, Christopher Ré, Irina Rish, and Ce Zhang. Redpajama: an open dataset for training large language models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 116462–116492. Curran Associates, Inc., 2024. doi: 10.52202/079017-3697. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/d34497330b1fd6530f7afd86d0df9f76-Paper-Datasets_and_Benchmarks_Track.pdf.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=yzkSU5zdwD. Survey Certification.

Orion Weller, Kathryn Ricci, Marc Marone, Antoine Chaffin, Dawn Lawrie, and Benjamin Van Durme. Seq vs seq: An open suite of paired encoders and decoders. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=z5Mn8Rxi3l.

Alexander Wettig, Tianyu Gao, Zexuan Zhong, and Danqi Chen. Should you mask 15% in masked language modeling? In Andreas Vlachos and Isabelle Augenstein (eds.), *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 2985–3000, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.217. URL https://aclanthology.org/2023.eacl-main.217/.

Chenghao Xiao, G Thomas Hudson, and Noura Al Moubayed. Rar-b: Reasoning as retrieval benchmark. *arXiv preprint arXiv:2404.06347*, 2024.

Zhenhui Xu, Linyuan Gong, Guolin Ke, Di He, Shuxin Zheng, Liwei Wang, Jiang Bian, and Tie-Yan Liu. Mc-bert: Efficient language pre-training via a meta controller. *arXiv preprint arXiv:2006.05744*, 2020.

Dongjie Yang, Zhuosheng Zhang, and Hai Zhao. Learning better masking for better language model pre-training. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 7255–7267, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.400. URL https://aclanthology.org/2023.acl-long.400/.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng YU, Zhengying Liu, Yu Zhang, James Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=N8N0hgNDRt.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/3fe78a8acf5fda99de95303940a2420c-Paper.pdf.

Weizhe Yuan, Jane Yu, Song Jiang, Karthik Padthe, Yang Li, Dong Wang, Ilia Kulikov, Kyunghyun Cho, Yuandong Tian, Jason Weston, and Xian Li. Naturalreasoning: Reasoning in the wild with 2.8m challenging questions. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference. Curran Associates, Inc., 2025. doi: 10.52202/085713-2246. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/60b22138917015fcba7165670e6f453c-Paper-Datasets_and_Benchmarks_Track.pdf.

Jiayou Zhang, Yifan Shen, Guangyi Chen, Le Song, and Eric Xing. Dimensional collapse in vqvaes: Evidence and remedies. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 170934–170966. Curran Associates, Inc., 2025a. doi: 10.52202/085713-5693. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/f9a50cf037f5ca2f687e3cd70b572c6f-Paper-Conference.pdf.

Junyan Zhang, Yiming Huang, Shuliang Liu, Yubo Gao, and Xuming Hu. Do BERT-like bidirectional models still perform better on text classification in the era of LLMs? In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 18980–18989, Suzhou, China, November 2025b. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.1033. URL https://aclanthology.org/2025.findings-emnlp.1033/.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. mGTE: Generalized long-context text representation and reranking models for multilingual text retrieval. In Franck Dernoncourt, Daniel Preoţiuc-Pietro, and Anastasia Shimorina (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 1393–1412, Miami, Florida, US, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-industry.103. URL https://aclanthology.org/2024.emnlp-industry.103/.

Fan Zhou, Zengzhi Wang, Nikhil Ranjan, Zhoujun Cheng, Liping Tang, Guowei He, Zhengzhong Liu, and Eric P. Xing. Megamath: Pushing the limits of open math corpora. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=SHB0sLrZrh.

Dawei Zhu, Liang Wang, Nan Yang, Yifan Song, Wenhao Wu, Furu Wei, and Sujian Li. LongEmbed: Extending embedding models for long context retrieval. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 802–816, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.47. URL https://aclanthology.org/2024.emnlp-main.47/.

APPENDIX

CONTENTS

A TRAINING SETUP **22**

A.1 ARCHITECTURE AND OPTIMIZATION 22

A.2 PRUNING AND DISTILLATION 22

A.3 DATASET MIXTURE 23

B EVALUATION PROTOCOLS **23**

B.1 CLASSIFICATION TASKS 24

B.2 RETRIEVAL TASKS 25

B.3 ZERO-SHOT AND FEW-SHOT MCQA 25

C ADDITIONAL ANALYSIS ON REPLACED TOKEN DETECTION **26**

C.1 COLA AS A CONFOUNDER 26

C.2 COMPARISON WITH RTD-TRAINED ENCODERS 26

D WHY KOLEO INCREASES EFFECTIVE RANK **26**

E DETAILED EVALUATION RESULTS **28**

E.1 REPORTED BASELINE SCORES 28

E.2 FULL GLUE RESULTS 28

E.3 FULL REWARDBENCH RESULTS 28

E.4 FULL BEIR RESULTS 28

E.5 FULL LONGEMBED RESULTS 28

A Training Setup

A.1 Architecture and Optimization

All encoders share the same architecture design. We use RoPE (Su et al., 2024), SwiGLU (Shazeer, 2020), and alternating local and global attention similar to Warner et al. (2025), where every third layer applies global attention and the remaining layers apply local attention. The RoPE base is 10K for local attention and 640K for global attention, and the latter is increased to 2.56M during the 32K context extension. The LM head is untied from the input embeddings. We use the same tokenizer of SmolLM2 (Ben Allal et al., 2025), with a vocabulary of 49,152 tokens.[6]

We scale the encoders by width while keeping the depth fixed at 24 layers. The hidden size grows from 768 for 150M to 1,024 for 380M and 2,048 for 1.3B, and the intermediate size of SwiGLU is set to 1,024, 3,072, and 5,632, respectively. The 150M model uses 6 attention heads and the two larger models use 8, with a head dimension of 128 for 150M and 380M and 256 for 1.3B.

We use AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 10^{-8}$, a weight decay of 0.1, and gradient clipping at 1.0. The learning rate follows the warmup-stable-decay schedule (Hu et al., 2024), with 10B tokens of warmup and a decay phase over the last 10% of the token budget. Note that the learning rate is decayed to zero for every model evaluated in this paper, including the intermediate checkpoints in Section 3.1, each of which is obtained by branching a decay phase from the stable phase. We use a peak learning rate of 3e-4 for the experiments in Section 3, and 1e-4 for distillation and the final training of UltraBERT. The batch size is 4M tokens, with 2,048 sequences of 2,048 tokens. For UltraBERT, we train on 2T tokens in total, and extend the context length to 32K for the last 100B tokens (Gao et al., 2025).

Each model is trained on a single pod slice of TPU v4-512, v5e-256, or v6e-256, all with 256 chips. We use 16 data-parallel replicas with 16-way fully sharded data parallelism (FSDP), and for the 32K context extension we reduce FSDP to 8-way and add 2-way tensor parallelism.

A.2 Pruning and Distillation

We derive the 380M and 150M encoders by reusing the best 1.3B encoder in Section 3.2 as the teacher, through Cascade Distillation (Liu et al., 2026), an iterative process of pruning and distillation: the 1.3B encoder is pruned to 380M and distilled, and the distilled 380M encoder is pruned again to 150M and distilled. We prune only along the width, keeping all 24 layers, following the observation of Muralidharan et al. (2024) that width pruning outperforms depth pruning at this scale. The 1.3B encoder is pruned to 380M by reducing the hidden size from 2,048 to 1,024, the head dimension from 256 to 128, and the intermediate size from 5,632 to 3,072, and the 380M encoder is pruned to 150M by reducing the hidden size to 768, the number of attention heads from 8 to 6, and the intermediate size to 1,024.

Following Muralidharan et al. (2024), we estimate the importance of hidden channels, attention heads, and FFN neurons from activations with forward passes only. As a calibration set, we take 2,000 documents from the pre-training mixture and pack them into 1,024 sequences of 1,024 tokens. For hidden channels, we use the output of the pre-attention normalization in each layer; for attention heads, the per-head output entering the output projection; and for FFN neurons, the gated activation entering the down projection. In all three cases, the activation magnitude is averaged over the sequence and then aggregated by the L2 norm over the batch. Hidden channel scores are further aggregated over layers, since the residual stream is shared across the network, whereas head and neuron scores are kept per layer. We keep the top-scoring channels, heads, and neurons.

When the head dimension is halved from 256 to 128, we reduce the query and key projections by keeping every other rotary frequency pair, so that the pairing of RoPE dimensions is preserved, and scale the query by $1/\sqrt{2}$ to keep the attention logits at the same magnitude. For the value and output projections, we apply a shared random orthogonal projection to the output of the value and the input of the output projection, which preserves their product in expectation.

Each pruned encoder is then distilled from the 1.3B teacher. As described in Section 3.5, we apply the KL divergence between the teacher and student logits on the 5% masked sequences, and the

[6] https://huggingface.co/HuggingFaceTB/cosmo2-tokenizer

standard MLM loss on the 50% masked sequences. The 150M encoder is distilled from the same 1.3B teacher rather than from the 380M encoder, following Muralidharan et al. (2024) and Liu et al. (2026), who found that distilling from the original parent gives better results than distilling from the intermediate child.

A.3 Dataset Mixture

Category	Dataset	Unique (B)	Epochs	Trained (B)	Ratio (%)
Web (66.9%)	DCLM (Li et al., 2024)	756.4	1.17	881.7	44.1
	FineWeb-Edu (Penedo et al., 2024)	195.4	2.33	455.5	22.8
Code (14.0%)	Stack-Edu (Ben Allal et al., 2025)	120.3	2.33	280.5	14.0
Academic (7.1%)	peS2o (Soldaini & Lo, 2023)	60.7	2.33	141.6	7.1
Math (5.8%)	MegaMath (Zhou et al., 2025)	20.9	2.80	58.4	2.9
	FineMath (Ben Allal et al., 2025)	10.9	2.80	30.5	1.5
	InfiWebMath (Ben Allal et al., 2025)	9.6	2.80	26.9	1.3
Reference (3.7%)	Cosmopedia v2 (Ben Allal et al., 2024)	28.1	2.33	65.6	3.3
	Wikipedia (Soldaini et al., 2024)	3.9	2.33	9.2	0.5
Instruction (2.5%)	FLAN (Longpre et al., 2023)	19.0	2.33	44.4	2.2
	StackExchange (Weber et al., 2024)	1.5	2.33	3.4	0.2
	Natural Reasoning (Yuan et al., 2025)	1.0	2.33	2.4	0.1
Total		1,227.8	1.63	2,000.0	100.0

Table 4: **Pre-training data mixture.** Unique denotes the number of tokens in one epoch of each dataset, and Trained the number of tokens seen within the 2T-token budget of UltraBERT, both counted with our tokenizer. Ratios in parentheses are per category.

Table 4 lists the pre-training data mixture. Web data takes about two thirds of the budget, followed by code, academic papers, and mathematics. We also include a small portion of instruction data: FLAN and StackExchange, as in Weller et al. (2026), and Natural Reasoning. Although it is only 2.5% of the mixture, the encoder learns to follow instructions to some extent, which explains why the base model already works in the zero-shot and few-shot MCQA settings of Appendix B.3 and Figure 3. The model recognizes the MCQA format without any tuning, and the instruction data works well through the `[MASK]` position. This goes beyond the observation of Samuel (2024) that MLM-trained encoders can act as in-context learners: with instruction data in the mixture, the encoder follows instructions even in the zero-shot setting without demonstrations.

We also explored multi-stage training that adjusts the mixture ratios across stages, as in Ettin (Weller et al., 2026) and EuroBERT (Boizard et al., 2025). This is common practice for decoder LLMs (OLMo Team et al., 2024; Ben Allal et al., 2025), where upsampling high-quality data in the final stage has been shown to be effective. In our preliminary experiments with several mixture schedules, however, the downstream performance after fine-tuning was barely affected. We hypothesize that the exposure ratio of each dataset changes the token prediction distribution of the model, and thus its apparent zero-shot and few-shot performance, but has little effect on the intrinsic linguistic knowledge that fine-tuning builds on. We therefore use a single mixture throughout training.

B Evaluation Protocols

In this section, we detail the fine-tuning and evaluation protocols used throughout the paper. For the baselines in Table 3, we take scores from the literature where available, such as GLUE and SQuAD 2.0, and fine-tune and evaluate the rest ourselves under the same protocol.

B.1 CLASSIFICATION TASKS

For every fine-tuning run, we use AdamW with β_1=0.9, β_2=0.98, ϵ=10^{-6}, weight decay 0.1, gradient clipping at 1.0, a linear learning rate schedule with 5% warm-up, and bfloat16 mixed precision. Following the common practice of prior encoders (Liu et al., 2019; Warner et al., 2025; Weller et al., 2026), we perform a grid search per task and per model. The learning rate is always swept over {5e-6, 8e-6, 1e-5, 3e-5, 5e-5, 8e-5}, jointly with the task-specific maximum lengths, batch sizes, and epochs in Table 5. During each run, the model is evaluated on the development set periodically and the best checkpoint is kept. We then select the best configuration on the development set and report its score; for tasks with a labeled test set (MultiCoNER v2, Few-NERD), the test score of the selected configuration is reported.

GLUE We evaluate on the 8 GLUE tasks (Wang et al., 2018) excluding WNLI, as in Devlin et al. (2019), and report development-set scores: Matthews correlation for CoLA, Pearson correlation for STS-B, accuracy on the matched set for MNLI, and accuracy for the remaining tasks. Inputs are truncated to 128 tokens. Following Liu et al. (2019), RTE, MRPC, and STS-B are initialized from the best MNLI checkpoint of the same model instead of the pre-trained weights. The large tasks (MNLI, QQP, QNLI, SST-2) use a batch size of 64 with $\{1, 2, 3\}$ epochs. The small tasks (CoLA, STS-B, MRPC, RTE) are known to be unstable (Portes et al., 2023; Nussbaum et al., 2025), so we sweep the batch size and $\{2, 5, 10\}$ epochs with 4 seeds, take the best development score over the grid for each seed, and report the median across the seeds.

SQuAD 2.0 We fine-tune on SQuAD 2.0 (Rajpurkar et al., 2018) with the same protocol as GLUE. Question–context pairs are truncated to 384 tokens with a document stride of 128. We sweep the batch size over $\{16, 32, 64\}$ and the number of epochs over $\{1, 2, 3\}$, and report the F1 score on the development set of the best configuration.

Named Entity Recognition We adopt two fine-grained NER benchmarks, which are known to be more challenging and thus discriminative among models. MultiCoNER v2 (Fetahu et al., 2023) contains short, lower-cased sentences with 33 entity types; we use the English subset (16,778 / 871 / 249,980 sentences for training, development, and test). Since the development set is small, every configuration is run with 3 seeds, and the average development score is used for selection. Few-NERD (Ding et al., 2021) is used in its supervised setting with 66 entity types and the official split (131,767 / 18,824 / 37,648). To avoid tokenization mismatches with byte-level BPE tokenizers, we prepend a leading space to each word so that the pre-tokenized encoding is identical to that of the raw string. We compute entity-level F1 and report the test scores of the selected configurations: macro-F1 over entity types for MultiCoNER v2 and micro-F1 for Few-NERD.

RewardBench To examine encoders as reward models, we fine-tune with the Bradley-Terry loss. The training pairs are the union of `HuggingFaceH4/UltraFeedback-binarized` (Cui et al., 2024) and `Skywork/Skywork-Reward-Preference-80K-v0.2` (Liu et al., 2024), restricted to single-turn conversations. This yields 135,981 pairs, of which 5% (6,800 pairs) are held out as a development set. Prompts and responses are concatenated with a plain template (`USER:` and `ASSISTANT:`) and truncated to 1,024 tokens. Within each run, the epoch checkpoint with the lowest loss on the held-out pairs is kept. We compute the accuracy of the 4 sections (Chat, Chat Hard, Safety, and Reasoning) with the official per-subset weighting, and report their average.

Task	Max Length	Batch Size	Epochs	Metric
MNLI, QQP, QNLI, SST-2	128	64	{1, 2, 3}	Accuracy
CoLA, STS-B, MRPC, RTE	128	{16, 32, 64}	{2, 5, 10}	MCC / Pearson / Acc.
SQuAD 2.0	384	{16, 32, 64}	{1, 2, 3}	F1
MultiCoNER v2	128	16	{2, 5, 10}	Macro-F1
Few-NERD	128	64	{1, 2, 3}	Micro-F1
RewardBench	1024	64	{1, 2, 3}	Section average

Table 5: Fine-tuning grids. The learning rate is swept over the 6 values in Section B.1 for every task.

B.2 Retrieval Tasks

For retrieval, we fine-tune each encoder as a single-vector dense retriever using mean pooling and the cached multiple negatives ranking loss (Gao et al., 2021). We use AdamW with β_1=0.9, β_2=0.999, ϵ=10^{-8} and no weight decay, together with gradient clipping at 1.0, a linear learning rate schedule with 5% warm-up, and bfloat16 mixed precision. Every model is trained with a per-device batch size of 512 on 8 GPUs, and evaluated with the `mteb` library (Muennighoff et al., 2023).

BEIR and RAR-b Following the training setup of Warner et al. (2025) and Weller et al. (2026), we fine-tune on MS MARCO (Nguyen et al., 2016) with 1.25M hard-negative triplets from `sentence-transformers/msmarco-co-condenser-margin-mse-sym-mnrl-mean-v1`, using one epoch and a maximum length of 256 tokens. The learning rate is swept over {1e-5, 3e-5, 5e-5, 8e-5, 1e-4, 2e-4, 3e-4, 5e-4, 8e-4, 1e-3}, and the checkpoint with the highest score on BEIR is selected to measure the best attainable retrieval performance of each backbone. We evaluate on the 15 BEIR datasets (Thakur et al., 2021), using the development split for MS MARCO and the mean over the 12 forums for CQADupstack, and report the average nDCG@10.

To estimate generalization to the code and math domains, the same checkpoint is evaluated on the code and math subsets of RAR-b (Xiao et al., 2024), which cast reasoning problems as retrieval. The code subset is curated from HumanEvalPack (Muennighoff et al., 2024), MBPP (Austin et al., 2021), CodeSearchNet (Husain et al., 2019), and TinyCode[7]. The math subset is curated from MATH (Hendrycks et al., 2021b), GSM8K (Cobbe et al., 2021), and MetaMathQA (Yu et al., 2024). For each query, the ground-truth solution is retrieved from 301K and 389K candidates, respectively, and we report nDCG@10.

MLDR and LongEmbed As MS MARCO passages are short, we further fine-tune the selected BEIR checkpoint on the English training triplets of MLDR (Chen et al., 2024), following the in-domain setting of Warner et al. (2025). We use 80K triplets from `sentence-transformers/mldr` and train with the same learning rate as the MS MARCO stage, with the maximum length set to the context length of each model, capped at 16K tokens. The resulting model is evaluated on the English test set of MLDR and on the 6 tasks of LongEmbed (Zhu et al., 2024), with the maximum length again set to the context length of each model, capped at 32K tokens. For LongEmbed, the 4 real tasks (NarrativeQA, QMSum, SummScreenFD, and 2WikiMultihopQA) are scored by nDCG@10, and the 2 synthetic tasks (Passkey and Needle) by recall@1 averaged over the 8 context lengths from 256 to 32K tokens. We report nDCG@10 on MLDR and the average over the 6 tasks on LongEmbed.

B.3 Zero-Shot and Few-Shot MCQA

To compare encoders and decoders on LLM-style benchmarks in Section 3.1, we evaluate 6 multiple-choice question answering (MCQA) benchmarks: ARC (Clark et al., 2018), BoolQ (Clark et al., 2019), CommonsenseQA (Talmor et al., 2019), MMLU (Hendrycks et al., 2021a), OpenBookQA (Mihaylov et al., 2018), and SIQA (Sap et al., 2019). We use the test sets of ARC, MMLU, and OpenBookQA, and the validation sets of BoolQ, CommonsenseQA, and SIQA, whose test labels are not public. We report accuracy, averaged over the 57 subjects for MMLU and over the Easy and Challenge sets for ARC, and the final score is the average over the 6 benchmarks.

Prompting and Scoring Each question is formatted with the prefix (`Question:`) followed by the options, and the suffix (`Answer:`). For BoolQ and SIQA, which come with contexts, the context is additionally prepended with the prefix `Context:`. For the few-shot setting, the first 5 examples of the development set of the same subject for MMLU, the validation set for ARC and OpenBookQA, and the training set for the others are prepended as demonstrations in the same format. Encoders and decoders share the prompt and differ only in how the answer is read out. For encoders, a single `[MASK]` token is appended after `Answer:` and the MLM logits at that position are compared over the option letters. For decoders, the next-token logits at the last position are used instead. For BoolQ, we compare the summed probabilities of the yes and no variants (`Yes`, `yes`, `True`, `true` vs. `No`, `no`, `False`, `false`). Inputs are truncated from the left to 2,048 tokens.

[7] `https://huggingface.co/datasets/nampdn-ai/tiny-codes`

Instruction Tuning Base models are evaluated in the 5-shot setting, and instruction-tuned models in the zero-shot setting. For instruction tuning, we use a subset of the FLAN collection (Longpre et al., 2023) with zero-shot templates that present answer options, sampling 100 to 1,000 examples from each of 718 tasks in Flan2021, P3, and NIv2, for a total of 545K examples. We keep the examples whose answers are at most 5 tokens long (480K), and hold out 1% for validation. Encoders are trained by appending as many `[MASK]` tokens as the answer length and predicting the answer tokens with the MLM head, and decoders with the standard causal LM loss on the answer tokens. Both are trained for 3 epochs with a batch size of 128, a learning rate of 5e-5, weight decay 0.01, a linear schedule with 10% warm-up, a maximum length of 256 tokens with left truncation, and bfloat16 mixed precision.

C Additional Analysis on Replaced Token Detection

C.1 CoLA as a Confounder

It has long been observed that RTD-trained encoders are particularly strong on CoLA. Di Liello et al. (2021) found that ELECTRA outperforms MLM and random-substitution variants by a wide margin only on CoLA, RTE, and STS-B, while the gap is not significant on the more stable tasks, and Xu et al. (2020) attributed the gain to the pre-training task of ELECTRA being more aligned with CoLA.

We observe the same in a controlled setting. Starting from the 150M encoder pre-trained with 50% masking in Section 3.2, we continue training with the RTD objective for only 100B tokens. As shown in Table 6, the average of the other GLUE tasks moves by 0.5 points, whereas CoLA jumps by 9.9 points, from 61.6 to 71.5. A short exposure to RTD is enough to change the CoLA score by an amount that no other task shows.

Model	GLUE w/o CoLA	CoLA
50% MLM	91.2	61.6
+ RTD 100B	91.7	71.5

Table 6: Effect of continued RTD training. Starting from the 150M pre-trained with 50% masking, we continue training with RTD for 100B tokens.

CoLA asks whether a sentence is linguistically acceptable, and RTD trains the encoder to judge whether each token is plausible in its context. The two are the same kind of judgment, which explains why the CoLA score reflects the form of the pre-training objective rather than the general quality of the encoder. For this reason, we exclude CoLA when comparing MLM and RTD in Section 3.2.

C.2 Comparison with RTD-Trained Encoders

Following Marone et al. (2025), we do not include RTD-trained encoders such as DeBERTaV3 (He et al., 2023) in Table 3. Prior work has consistently found that they fall far behind on retrieval despite their strength on classification (Warner et al., 2025; Breton et al., 2025; Antoun et al., 2025; Boizard et al., 2025; Marone et al., 2025).

For brevity, we report only the classification and retrieval averages of DeBERTaV3 here, computed as in Figure 1. DeBERTaV3-base achieves 74.2 on classification and 20.7 on retrieval, whereas UltraBERT-150M achieves 75.4 and 45.3, respectively. DeBERTaV3-large achieves 78.1 and 25.8, whereas UltraBERT-380M achieves 78.5 and 48.7. Consistent with prior work, we observe substantially lower retrieval performance, with gaps of 24.6 and 22.9 points at the two sizes.

D Why KoLeo Increases Effective Rank

In this section, we show that the KoLeo regularizer used in Section 3.3 increases the effective rank of representations. The argument proceeds in three steps: KoLeo maximizes the entropy of the representation distribution, near-maximal entropy forces the second moment to be near-isotropic, and near-isotropy flattens the singular value spectrum, which maximizes the effective rank.

Setup Let $Z \in \mathbb{R}^{n \times d}$ stack the ℓ_2-normalized representations $z_1, \ldots, z_n \in \mathbb{S}^{d-1}$ of a batch, drawn from the representation distribution P, and let $\Sigma = \mathbb{E}_P[zz^\top]$ with eigenvalues λ. Since $\|z\| = 1$,

$\operatorname{tr}\Sigma = 1$, so λ lies on the probability simplex. Let U be the uniform distribution on $\mathbb{S}^{d-1}$, whose second moment is $\frac{1}{d}I_d$ by symmetry.

Fact 1 (KoLeo estimates entropy)**.** *Up to a positive scaling and an additive constant, $\mathcal{L}_{\mathrm{KoLeo}}$ is the negative Kozachenko–Leonenko estimate of the entropy of P (Beirlant et al., 1997; Sablayrolles et al., 2019). Minimizing it therefore maximizes the (estimated) entropy of P, that is, it minimizes $\mathrm{KL}(P \,\|\, U)$, its divergence from the entropy-maximizing uniform distribution.*

Proposition 1 (Near-maximal entropy forces near-isotropy)**.**

$$\left\|\Sigma - \tfrac{1}{d}I_d\right\|_2 \;\le\; \sqrt{\mathrm{KL}(P \,\|\, U)/2}.$$

Proof. Fix a unit vector $v \in \mathbb{R}^d$ and consider the test function $g_v(z) = (v^\top z)^2$. By Cauchy–Schwarz, $0 \le (v^\top z)^2 \le \|v\|^2\|z\|^2 = 1$ for $z \in \mathbb{S}^{d-1}$, so g_v takes values in $[0,1]$. Its means under the two distributions are

$$\mathbb{E}_P[g_v] = \mathbb{E}_P\big[v^\top z z^\top v\big] = v^\top \Sigma v, \qquad \mathbb{E}_U[g_v] = v^\top\big(\tfrac{1}{d}I_d\big)v = \tfrac{1}{d},$$

using the second moment of U from the setup and $\|v\| = 1$. Subtracting the two identities gives

$$\big|v^\top\big(\Sigma - \tfrac{1}{d}I_d\big)v\big| = \big|\mathbb{E}_P[g_v] - \mathbb{E}_U[g_v]\big|.$$

We now bound the right-hand side. Since P and U are probability measures, $\int c\,\mathrm{d}(P-U) = 0$ for any constant c, so replacing g_v by the centered function $\tilde{g}_v = g_v - \frac{1}{2}$ leaves the difference of means unchanged while ensuring $|\tilde{g}_v| \le \frac{1}{2}$. Hence

$$\big|\mathbb{E}_P[g_v] - \mathbb{E}_U[g_v]\big| = \left|\int \tilde{g}_v\,\mathrm{d}(P-U)\right| \le \tfrac{1}{2}\,\|P-U\|_1 = \mathrm{TV}(P,U),$$

where the last identity is the ℓ_1 characterization of the total variation distance. Pinsker's inequality (Cover & Thomas, 2005) bounds this in turn by the KL divergence, $\mathrm{TV}(P,U) \le \sqrt{\mathrm{KL}(P \,\|\, U)/2}$. Combining the displays, every unit vector v satisfies $|v^\top(\Sigma - \frac{1}{d}I_d)v| \le \sqrt{\mathrm{KL}(P \,\|\, U)/2}$. Since $\Sigma - \frac{1}{d}I_d$ is symmetric, the spectral theorem gives $\|\Sigma - \frac{1}{d}I_d\|_2 = \sup_{\|v\|=1}|v^\top(\Sigma - \frac{1}{d}I_d)v|$, and taking the supremum over v in the bound above proves the claim. □

Proposition 2 (Isotropy deviation equals spectral deviation)**.**

$$\left\|\Sigma - \tfrac{1}{d}I_d\right\|_2 \;=\; \max_k\big|\lambda_k - \tfrac{1}{d}\big|.$$

Proof. Since Σ is symmetric, the spectral theorem provides an orthonormal eigenbasis $u_1, \dots, u_d$ with $\Sigma u_k = \lambda_k u_k$. Because every vector is an eigenvector of the identity, the subtraction of $\frac{1}{d}I_d$ preserves this eigenbasis and merely shifts the spectrum,

$$\big(\Sigma - \tfrac{1}{d}I_d\big)u_k = \lambda_k u_k - \tfrac{1}{d}u_k = \big(\lambda_k - \tfrac{1}{d}\big)u_k,$$

so $\Sigma - \frac{1}{d}I_d$ is diagonalized by the same basis with eigenvalues $\lambda_k - \frac{1}{d}$. Expanding any unit vector as $v = \sum_k c_k u_k$ with $\sum_k c_k^2 = 1$ then gives

$$\left\|\big(\Sigma - \tfrac{1}{d}I_d\big)v\right\|^2 = \left\|\sum_k c_k\big(\lambda_k - \tfrac{1}{d}\big)u_k\right\|^2 = \sum_k c_k^2\big(\lambda_k - \tfrac{1}{d}\big)^2 \le \max_k\big(\lambda_k - \tfrac{1}{d}\big)^2,$$

where the middle identity uses orthonormality and the inequality holds because the sum is a convex combination of the squared shifted eigenvalues. Choosing $v = u_{k^*}$ for an index k^* attaining the maximum turns the inequality into an equality. Therefore

$$\left\|\Sigma - \tfrac{1}{d}I_d\right\|_2 = \max_{\|v\|=1}\left\|\big(\Sigma - \tfrac{1}{d}I_d\big)v\right\| = \max_k\big|\lambda_k - \tfrac{1}{d}\big|.$$

□

Corollary 1 (KoLeo flattens the spectrum)**.** *Combining Fact 1, Propositions 1 and 2,*

$$\max_k\big|\lambda_k - \tfrac{1}{d}\big| \;\le\; \sqrt{\mathrm{KL}(P \,\|\, U)/2}.$$

Hence, as minimizing $\mathcal{L}_{\mathrm{KoLeo}}$ drives $\mathrm{KL}(P \,\|\, U)$ toward zero, every eigenvalue of Σ is confined to a shrinking interval around $1/d$ and the spectrum flattens.

The batch second moment $\frac{1}{n}Z^\top Z$ concentrates around Σ, and its eigenvalues are the squared singular values of $Z/\sqrt{N}$, so the singular values σ_k flatten toward a common value as well. Since the effective rank (Roy & Vetterli, 2007; Garrido et al., 2023) is maximized precisely when the singular values are equal, KoLeo exerts an explicit pressure toward maximal effective rank, consistent with the empirical increases observed when it is added (Zhang et al., 2025a; Letellier et al., 2026).

E Detailed Evaluation Results

E.1 Reported Baseline Scores

For the baselines in Table 3, we take scores from the literature where available and evaluate the rest with the protocols in Appendix B. GLUE scores are taken from the original papers of RoBERTa (Liu et al., 2019), MosaicBERT (Portes et al., 2023), NomicBERT (Nussbaum et al., 2025), GTE-en-MLM (Zhang et al., 2024), ModernBERT (Warner et al., 2025), Ettin Encoder (Weller et al., 2026), and NeoBERT (Breton et al., 2025), and the GLUE scores of BERT are taken from Portes et al. (2023). SQuAD 2.0 scores of BERT and RoBERTa-large are taken from Liu et al. (2019). All other scores, including GLUE of LFM2.5-Encoder, SQuAD 2.0 of the remaining models, and the NER, RewardBench, and retrieval benchmarks (BEIR, RAR-b, MLDR, and LongEmbed) of every baseline, are obtained by us. Table 7 verifies that our GLUE protocol reproduces the reported scores of Ettin Encoder within 0.2 points at both sizes.

Model	**# Params**	**MNLI**	**QNLI**	**QQP**	**RTE**	**SST-2**	**MRPC**	**CoLA**	**STS-B**	**Avg.**
Ettin Encoder	149M	89.2	94.0	92.4	87.7	95.8	92.6	66.9	92.2	88.9
(Our evaluation)		88.9	93.8	92.1	88.4	95.4	92.2	67.0	92.5	88.8
Ettin Encoder	395M	91.3	95.2	93.0	92.8	96.7	93.6	71.3	92.7	90.8
(Our evaluation)		90.9	95.3	93.0	91.7	96.7	93.1	71.1	92.9	90.6

Table 7: GLUE scores of Ettin Encoder as reported by Weller et al. (2026) and as re-evaluated with our fine-tuning protocol.

E.2 Full GLUE Results

Table 8 reports the per-task GLUE scores of Table 3. UltraBERT is the best on most tasks at both sizes, with the exceptions of QQP and CoLA at the base size and QQP, SST-2, MRPC, and CoLA at the medium size, where it stays within 1 point of the best model except for CoLA.

E.3 Full RewardBench Results

Table 9 reports the 4 section scores of RewardBench. UltraBERT is the best on Chat, Chat Hard, and Reasoning at the base size, and on Chat Hard and Safety at the medium size, where Ettin Encoder is slightly ahead on Chat and Reasoning. The largest margin is on Chat Hard, where UltraBERT-380M leads the second best by 4.9 points.

E.4 Full BEIR Results

Table 10 reports the nDCG@10 on each of the 15 BEIR datasets, along with the selected learning rate for MS MARCO fine-tuning. UltraBERT is the best on 10 datasets at the base size and 8 at the medium size, where LFM2.5-Encoder-350M leads on NFCorpus, SciFact, TREC-COVID, and CQADupstack.

E.5 Full LongEmbed Results

Table 11 reports the 6 LongEmbed tasks with the maximum context length of each model. UltraBERT is the best on all tasks at the base size, and at the medium size it leads on all tasks except Needle and 2WikiMultihopQA, where LFM2.5-Encoder-230M is ahead. The largest gains are on the two synthetic tasks, Passkey and Needle, where the 32K context is fully used.

Figure 11 breaks down the Passkey accuracy by document length. Every baseline degrades once the document exceeds its context length: the 512-context models fail beyond 1K tokens, and the 8K-context models, including ModernBERT and Ettin Encoder, drop sharply at 16K and 32K. LFM2.5-Encoder degrades more gracefully but still falls below 90% at 32K. UltraBERT is the only model that retains 99% accuracy at every length up to 32K, at both sizes.

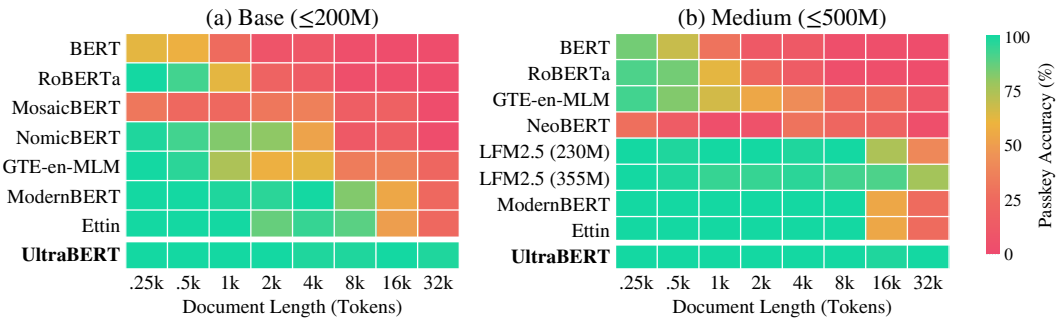


Figure 11: Passkey retrieval accuracy on LongEmbed as a function of document length, for encoders in two size groups, (a) base (≤200M parameters) and (b) medium (≤500M parameters). Each cell is the recall@1 over the 50 queries at that length, from 256 to 32,768 tokens.

Model	# Params	MNLI	QNLI	QQP	RTE	SST-2	MRPC	CoLA	STS-B	Avg.
					Base (≤200M)					
BERT	110M	85.4	91.6	91.4	78.2	93.1	89.5	59.0	89.4	84.7
RoBERTa	125M	87.6	92.8	91.9	78.7	94.8	90.2	63.6	91.2	86.4
MosaicBERT	137M	85.6	91.4	92.0	83.0	93.5	89.0	58.2	90.3	85.4
NomicBERT	137M	86.0	92.0	92.0	82.0	93.0	88.0	50.0	90.0	84.0
GTE-en-MLM	137M	86.7	91.9	88.8	84.8	93.4	92.1	57.0	90.2	85.6
ModernBERT	149M	89.1	93.9	92.1	87.4	<u>96.0</u>	92.2	65.1	91.8	88.4
Ettin Encoder	149M	<u>89.2</u>	<u>94.0</u>	**92.4**	<u>87.7</u>	95.8	<u>92.6</u>	**66.9**	<u>92.2</u>	<u>88.9</u>
UltraBERT	152M	**90.0**	**94.5**	<u>92.2</u>	**91.0**	**96.3**	**93.6**	<u>65.2</u>	**93.1**	**89.5**
					Medium (≤500M)					
BERT	334M	86.3	92.8	90.9	83.8	93.3	87.8	56.2	90.6	85.2
RoBERTa	355M	90.2	94.7	92.2	86.6	96.4	90.9	68.0	92.4	88.9
GTE-en-MLM	434M	89.2	93.9	89.2	88.1	95.1	<u>93.5</u>	60.4	91.4	87.6
NeoBERT	222M	89.0	93.7	90.7	91.3	95.6	93.4	66.2	91.8	89.0
LFM2.5-Encoder	230M	88.4	93.7	92.4	86.6	94.6	91.9	65.5	92.4	88.2
LFM2.5-Encoder	355M	89.5	94.4	92.6	89.2	96.3	91.8	66.8	<u>92.8</u>	89.2
ModernBERT	395M	<u>90.8</u>	<u>95.2</u>	<u>92.7</u>	92.1	**97.1**	91.7	**71.4**	<u>92.8</u>	90.4
Ettin Encoder	395M	**91.3**	<u>95.2</u>	**93.0**	<u>92.8</u>	<u>96.7</u>	**93.6**	<u>71.3</u>	92.7	<u>90.8</u>
UltraBERT	380M	**91.3**	**95.4**	92.6	**93.9**	96.4	93.4	71.2	**93.4**	**91.0**

Table 8: Detailed GLUE results on the development set. Best scores in each group are in bold and second best are underlined.

Model	# Params	Chat	Chat Hard	Safety	Reasoning	Avg.
			Base (≤200M)			
BERT	110M	74.0	52.9	58.6	51.9	59.4
RoBERTa	125M	77.7	55.7	58.6	60.8	63.2
MosaicBERT	137M	75.7	51.1	58.2	65.0	62.5
NomicBERT	137M	74.9	53.9	<u>66.4</u>	61.4	64.1
GTE-en-MLM	137M	77.7	55.7	**66.5**	53.7	63.4
ModernBERT	149M	77.7	59.2	60.7	74.4	68.0
Ettin Encoder	149M	<u>80.7</u>	<u>61.2</u>	63.1	<u>76.2</u>	<u>70.3</u>
UltraBERT	152M	**85.8**	**63.4**	<u>66.4</u>	**76.5**	**73.0**
			Medium (≤500M)			
BERT	334M	72.1	52.6	60.9	64.9	62.6
RoBERTa	355M	80.4	62.3	65.3	66.0	68.5
GTE-en-MLM	434M	78.2	49.8	61.4	59.6	62.2
NeoBERT	222M	61.5	53.7	54.5	60.9	57.6
LFM2.5-Encoder	230M	82.1	59.2	61.9	66.7	67.5
LFM2.5-Encoder	355M	82.4	57.9	63.5	70.4	68.5
ModernBERT	395M	84.6	64.3	65.1	77.0	72.8
Ettin Encoder	395M	**88.3**	<u>66.4</u>	<u>70.7</u>	**81.6**	<u>76.7</u>
UltraBERT	380M	<u>88.0</u>	**71.3**	**74.9**	<u>81.2</u>	**78.8**

Table 9: Detailed RewardBench results. Best scores in each group are in bold and second best are underlined.

Model	# Params	LR	NF Corpus	SciFact	TREC-COVID	FiQA	ArguAna	Climate FEVER	DBPedia	FEVER	Hotpot QA	MS MARCO	NQ	Quora	SciDocs	Touche 2020	CQA Dupstack	Avg.
Base (≤200M)																		
BERT	110M	3e-4	22.7	49.3	54.7	21.7	40.9	18.9	24.0	62.6	42.0	34.1	37.0	82.9	9.7	22.9	27.1	36.7
RoBERTa	125M	3e-4	24.3	53.1	59.1	26.8	46.5	20.6	26.1	64.3	44.5	37.9	41.5	85.3	12.2	22.4	29.9	39.6
MosaicBERT	137M	3e-4	27.5	53.7	58.9	28.9	45.0	21.5	29.6	67.1	48.9	**40.4**	44.1	86.0	13.6	**23.6**	32.3	41.4
NomicBERT	137M	3e-4	26.7	55.5	67.8	26.1	44.6	22.3	**31.7**	67.1	50.4	39.8	46.2	85.1	13.6	21.9	30.6	42.0
GTE-en-MLM	137M	1e-4	**29.3**	56.2	56.2	31.1	50.0	**25.5**	30.5	68.4	51.0	38.7	43.1	86.1	15.0	20.5	34.0	42.4
ModernBERT	149M	5e-4	26.5	58.3	73.7	29.8	46.3	23.1	28.1	67.0	48.2	38.6	44.3	86.6	13.3	21.9	33.2	42.6
Ettin Encoder	149M	5e-4	25.4	58.8	73.9	30.0	48.3	22.7	26.7	64.4	47.5	38.1	43.5	86.7	13.3	23.1	33.8	42.4
UltraBERT	152M	5e-5	28.4	**64.3**	**75.8**	**33.5**	**50.2**	23.9	30.6	**69.6**	**52.5**	40.1	**47.8**	**87.2**	**15.2**	23.5	**36.0**	**45.2**
Medium (≤500M)																		
BERT	334M	1e-4	24.6	52.5	55.2	25.1	46.3	21.4	23.1	65.7	44.5	32.8	39.0	84.3	13.0	20.7	30.5	38.6
RoBERTa	355M	1e-4	27.9	55.4	60.3	33.0	51.3	24.9	28.6	70.0	49.8	38.6	46.6	86.9	14.4	20.1	34.2	42.8
GTE-en-MLM	434M	1e-4	30.3	58.5	53.4	33.2	48.9	19.6	25.1	49.8	42.0	35.4	42.3	83.7	15.4	19.3	34.8	39.4
NeoBERT	222M	3e-4	30.0	56.3	71.3	28.7	47.8	19.9	31.6	67.5	50.7	40.1	44.3	86.2	15.0	22.4	31.9	42.9
LFM2.5-Encoder	230M	5e-4	30.6	66.1	78.8	33.0	52.3	**26.1**	32.2	69.2	55.0	40.8	50.0	87.3	16.2	**25.9**	37.4	46.7
LFM2.5-Encoder	355M	3e-4	**32.1**	**68.5**	**80.3**	34.9	**53.9**	25.7	32.8	69.8	**56.5**	40.3	50.4	87.6	16.8	21.9	**38.4**	47.3
ModernBERT	395M	3e-4	29.2	62.4	77.7	34.4	51.1	24.1	28.9	68.7	51.2	40.7	49.2	87.2	15.5	22.1	36.0	45.2
Ettin Encoder	395M	5e-4	29.6	65.2	79.4	33.3	50.8	24.2	31.7	67.1	51.2	**41.6**	49.5	87.3	15.6	25.1	36.5	45.9
UltraBERT	380M	3e-5	31.7	66.4	77.0	**38.1**	**53.9**	25.7	**33.9**	**73.2**	**56.5**	41.1	**51.6**	**87.7**	**18.0**	23.3	38.0	**47.7**

Table 10: Detailed BEIR results (nDCG@10). LR denotes the selected learning rate for MS MARCO fine-tuning. Best scores in each group are in bold and second best are underlined.

Model	# Params	Length	Passkey	Needle	Narrative QA	QMSum	Summ ScreenFD	2Wiki MultihopQA	Avg.
Base (≤200M)									
BERT	110M	512	21.5	19.2	18.8	18.1	54.2	43.8	29.3
RoBERTa	125M	512	37.3	21.0	20.5	19.9	58.6	45.6	33.8
MosaicBERT	137M	512	23.0	38.2	23.2	24.2	77.4	57.0	40.5
NomicBERT	137M	2048	54.2	30.8	22.4	25.4	77.5	59.5	45.0
GTE-en-MLM	137M	8192	60.2	50.2	43.1	33.6	91.8	71.7	58.5
ModernBERT	149M	8192	81.7	59.0	35.3	38.5	89.6	75.3	63.2
Ettin Encoder	149M	8192	80.2	50.8	38.3	33.7	90.1	67.6	60.1
UltraBERT	152M	32768	**99.2**	**59.8**	**56.0**	**45.4**	**95.2**	**76.6**	**72.0**
Medium (≤500M)									
BERT	334M	512	27.0	17.8	20.2	17.4	55.3	42.6	30.0
RoBERTa	355M	512	35.2	18.8	20.7	18.9	57.2	42.6	32.2
GTE-en-MLM	434M	8192	49.8	42.8	43.6	37.2	92.2	76.3	57.0
NeoBERT	222M	4096	16.0	14.3	19.5	23.2	70.7	50.5	32.4
LFM2.5-Encoder	230M	8192	88.8	**78.2**	47.3	37.5	89.2	**79.0**	70.0
LFM2.5-Encoder	355M	8192	92.5	72.8	52.4	41.2	93.3	78.0	71.7
ModernBERT	395M	8192	85.0	51.5	40.2	38.0	91.8	74.7	63.5
Ettin Encoder	395M	8192	85.0	56.5	42.3	37.7	91.3	72.8	64.3
UltraBERT	380M	32768	**99.8**	68.2	**58.2**	**48.0**	**96.6**	78.7	**74.9**

Table 11: Detailed LongEmbed results. Length denotes the maximum context length of each model. Passkey and Needle are scored by recall@1 averaged over context lengths, and the remaining tasks by nDCG@10. Best scores in each group are in bold and second best are underlined.