ModernBERT or DeBERTaV3? Examining Architecture and Data Influence on Transformer Encoder Models Performance

Wissam Antoun, Benoît Sagot & Djamé Seddah
Inria, Paris, France
{wissam.antoun,benoit.sagot,djame.seddah}@inria.fr

Abstract

Pretrained transformer-encoder models like DeBERTaV3 and ModernBERT introduce architectural advancements aimed at improving efficiency and performance. Although the authors of ModernBERT report improved performance over DeBERTaV3 on several benchmarks, the lack of disclosed training data and the absence of comparisons using a shared dataset make it difficult to determine whether these gains are due to architectural improvements or differences in training data. In this work, we conduct a controlled study by pretraining ModernBERT on the same dataset as CamemBERTaV2, a DeBERTaV3 French model, isolating the effect of model design. Our results show that the previous model generation remains superior in sample efficiency and overall benchmark performance, with ModernBERT's primary advantage being its support for long context, faster training, and inference speed. However, the new proposed model still provides meaningful architectural improvements compared to earlier models such as BERT and RoBERTa. Additionally, we observe that high-quality pre-training data accelerates convergence but does not significantly improve final performance, suggesting potential benchmark saturation. These findings show the importance of disentangling pretraining data from architectural innovations when evaluating transformer models.

1 Introduction

Despite the widespread adoption of decoder-only large language models (LLMs) in our post-ChatGPT era, encoder-only transformers such as BERT (Devlin et al., 2019) continue to play a central role in many NLP applications. These models remain the backbone of a wide range of non-generative tasks such as classification, named entity recognition (NER), and retrieval-based systems, especially in high-throughput or latency-sensitive environments. Their relatively low compute requirements and strong performance across standard information benchmarks benchmarks make them a practical choice for large-scale deployment, including in Retrieval-Augmented Generation (Lewis et al., 2020) pipelines or guardrails systems (Neill et al., 2024). Notable examples include Google's EmbeddingGemma (Vera et al., 2025), the BGE family of 33 models (Chen et al., 2024; Xiao et al., 2023), and multilingual encoders such as multi-ligual modernBERT (Marone et al., 2025) and EuroBERT (Boizard et al., 2025), alongside GTE-ModernBERT (Li et al., 2023; Zhang et al., 2024).

Continuous architectural and training objective improvements have led to more performant and efficient encoder-only transformer variants, among which DeBERTaV3 (He et al., 2021a) and the recently proposed ModernBERT (Warner et al., 2024) stand out as major improvements. According to Warner et al. (2024), their model reports superior performance relative to DeBERTaV3, the previous state-of-art model, across several popular NLP benchmarks. However, interpreting these performance improvements is challenging due to the lack of details regarding their training data. Without comparisons conducted on identical datasets, it remains unclear whether the reported gains reflect genuine architectural advances or simply differences arising from the choice of training data.

This uncertainty motivates our study aimed to evaluate the impact of architecture versus training data by conducting a controlled comparison among ModernBERT, DeBERTaV3, and RoBERTa models. We select CamemBERTaV2 (Antoun et al., 2024, 2023), a French DeBERTaV3 model trained from scratch, as our primary reference since both its intermediate checkpoints and training dataset are publicly available.

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, pages 3061–3074

Additionally, we include CamemBERTv2 (Antoun et al., 2023, 2024), a RoBERTa (Liu, 2019) based model pretrained on the same dataset, to comprehensively assess how ModernBERT's architectural advancements compare not only against the latest DeBERTaV3-based models but also against more traditional BERT/RoBERTa architectures. Leveraging these resources, we pretrained a French ModernBERT using the exact same dataset as CamemBERTaV2 and CamemBERTv2, thus ensuring that differences in performance directly reflect architectural variations rather than dataset composition or quality. In addition, we pretrained another ModernBERT variant on a carefully curated, high-quality French corpus to further explore the role of dataset quality in model performance.

The key takeaways from our comprehensive experiments are as follows:

- When controlling for dataset differences, DeBERTaV3 outperforms ModernBERT in terms of overall benchmark performance and training sample efficiency, with the notable exception of text retrieval tasks where DeBERTaV3 fails completely. This indicates that while DeBERTaV3's architecture and training objective optimization provide superior learning capabilities compared to ModernBERT's efficiency-oriented design, they do not generalize to all tasks.

- Nonetheless, ModernBERT presents a clear practical advantage due to its significantly faster training and inference speeds, driven by an orthogonal set of optimizations. Moreover, while ModernBERT may not surpass DeBERTaV3, it offers meaningful improvements over previous transformer-based models such as BERT and RoBERTa.

- We also observe that training models on high-quality, filtered datasets results in faster convergence but does not substantially increase final performance metrics. This finding highlights a potential limitation of current NLP benchmarks, suggesting possible saturation that prevents fine-grained discrimination between models of similar performance.

Our findings show the importance of clearly separating respective effects of architectural changes and training datasets when evaluating NLP models. Our controlled comparison using the same pretraining dataset provides more accurate insights into the strengths and limitations of ModernBERT and DeBERTaV3 architectures.

To promote further research and ensure reproducibility, we publicly release our two pretrained French ModernBERT models, collectively named ModernCamemBERT, including one trained on the CamemBERTaV2 dataset and another on our high-quality filtered corpus. These models, along with intermediate checkpoints, and evaluation results, are available on HuggingFace[1].

2 Related Works

Transformer-based language models have become the cornerstone of modern NLP, starting with BERT (Devlin et al., 2019), which introduced masked language modeling (MLM) and next sentence prediction as self-supervised pretraining tasks used to pretrain encoder-only transformer models. RoBERTa (Liu, 2019) subsequently improved upon BERT by removing the next sentence prediction objective, training on larger corpora, and applying more robust optimization techniques.

Despite these advances, both BERT and RoBERTa shared a fundamental architectural limitation: they used absolute positional embeddings and standard attention mechanisms that lacked efficiency and fine-grained contextual representation. To address these limitations, DeBERTa (He et al., 2021b) introduced a disentangled attention mechanism, decoupling content and positional information, thereby improving the model's ability to generalize across contexts. DeBERTaV3 (He et al., 2021a) further extended these innovations by incorporating Replaced Token Detection (RTD) (Clark et al., 2020) for more sample-efficient training, as well as Gradient-Disentangled Embedding Sharing (GDES) to prevent conflicting updates in shared embeddings between the generator and discriminator during training.

In parallel, architectural and efficiency-driven improvements have become an active area of research. ModernBERT (Warner et al., 2024) was recently proposed to modernize the BERT

[1] https://huggingface.co/collections/almanach/moderncamembert

architecture by incorporating a suite of design choices aimed at improving inference speed, training throughput, and context window size. These include FlashAttention (Dao et al., 2022; Dao, 2024; Shah et al., 2024), alternating global and local attention layers (Team et al., 2024), sequence packing (Portes et al., 2023), and rotary positional embeddings (RoPE) (Su et al., 2021). ModernBERT also removes architectural elements such as bias terms and introduces GeGLU (Shazeer, 2020) activations, making it a strong contender for production scenarios requiring high efficiency.

While ModernBERT has gained popularity other models have also made significant contributions to the field. For instance, MosaicBERT (Portes et al., 2023) was the first to focus on enhancing training efficiency and performance by increasing masking rates, sequence packing and FlashAttention. NomicBERT (Nussbaum et al., 2025) introduced architectural improvements like SwiGLU activation functions, RoPE positional encoding and extended context lengths, enhancing its ability to handle longer sequences up to 2048 tokens. NeoBERT (Breton et al., 2025) further advanced these developments by optimizing the depth-to-width ratio and doubling the context length, while also significantly increasing training corpus size.

We chose to compare against ModernBERT because it was the best available model at the moment we started our experimentation and it is the most popular encoder in the field. While the authors of ModernBERT report superior benchmark performance over DeBERTaV3, the absence of transparent training data and lack of head-to-head comparisons on shared datasets introduces ambiguity. It is thus unclear whether the reported improvements are driven by architectural enhancements or the underlying training data.

3 Methodology

We conduct a controlled study focusing on the performance of ModernBERT compared to DeBERTaV3-based and RoBERTa-based models. Our goal is to identify and separate architectural improvements from data-driven performance differences, addressing ambiguities in prior studies that used undisclosed datasets.

3.1 Pre-training Datasets

Our experiments involve two distinct pre-training datasets:

CamemBERTaV2 Original Dataset. We first make use of the publicly available French dataset originally used by the authors of CamemBERTaV2 (Antoun et al., 2024). This dataset has 275 billion tokens, sourced from:

- **CulturaX-FR Corpus** (Nguyen et al., 2023): The French subset of a multilingual corpus containing around 265 billion French tokens, constructed from multiple snapshots of OSCAR (Ortiz Suárez et al., 2019; Abadji et al., 2021, 2022) and mC4 (Xue et al., 2021) corpora.
- **HALvesting Corpus** (Kulumba et al., 2024): Approximately 4.7 billion tokens of academic and scientific content from French theses and research papers.
- **French Wikipedia**: Roughly 0.5 billion tokens from Wikipedia, intentionally upsampled to enhance general knowledge representation.

This dataset serves as a reference point, allowing us to directly compare models under identical training dataset conditions.

High-Quality Filtered Dataset. We also feature a second significantly larger 1T tokens French dataset created by applying heuristic and semantic filters in addition to full deduplication to the French section of the RedPajamaV2 corpus (Weber et al., 2024), combined with the HALvesting corpus and French Wikipedia. Semantic filtering was done following the FineWeb-Edu (Penedo et al., 2024) methodology. This method has been effective in increasing the overall quality of a corpus used for LLM training and has been widely adopted in the literature (Li et al., 2024; Su et al., 2025). First, We annotated 200K samples from the RedPajamaV2 dataset with quality labels (low, medium, high), using the LLama-3 70B model (Grattafiori et al., 2024) and the prompt provided in Appendix A. This annotated subset was then used to fine-tune XLM-V-base (Liang et al., 2023), which we use to annotate the whole RedPajamaV2 corpus. The semantic score was combined with the perplexity score from a language model trained on Wikipedia, as included in the RedPajama dataset. The

RedPajama authors categorize data into head, middle, and tail buckets based on perplexity. We only select data if it is either from the head bucket or has a high score from the quality classifier, while disgrading any tail or low labeled documents.

3.2 Model Training

We pretrained two variants of the ModernBERT model, one on the CamemBERTaV2 dataset, which we call ModernBERT-CV2, and the other on our high-quality filtered dataset, named ModernBERT-HQ, periodically saving intermediate checkpoints.[2] Both variants are base-sized models trained with Masked Language Modeling (MLM) for 1 trillion tokens, with a maximum sequence length of 1024, and use the same tokenizer as CamemBERTaV2. We maintained the rate of dynamic token masking to 30%, while retaining all other hyperparameters consistent with those of ModernBERT-base. Training was done on 48 NVidia H100 80GB GPUs using Pytorch's FSDP full sharding with bfloat16 mixed precision to speed up training[3].

Since our models were trained using a Warmup-Stable-Decay (WSD) learning rate schedule, each intermediate checkpoint underwent additional cooldown training over an extra 50 billion tokens, during which the learning rate decayed fully to zero, ensuring fair comparisons across checkpoints.

Additionally, we leveraged publicly available intermediate checkpoints from the CamemBERTaV2 and CamemBERTv2 models, allowing direct comparisons of learning trajectories and data efficiency across different architectures.

4 Experiments and Results

4.1 Downstream Evaluation Tasks

To evaluate our models, we consider a range of French downstream tasks and datasets, including:

- **Question Answering (QA)**: using FQuAD 1.0 (d'Hoffschmidt et al., 2020)
- **Named Entity Recognition (NER)**: on the 2008 FTB version (Abeillé et al., 2000; Candito and Crabbé, 2009) with NER annotations by Sagot et al. (Sagot et al., 2012)
- **Text Classification** capabilities assessed using the FLUE benchmark (Le et al., 2020) using the CLS amazon reviews classification task, the PAWS-X paraphrase identification task and XNLI task.
- **Text Retrieval**: We used the French subset of the translated Semantic Textual Similarity (STS) benchmark (May, 2021) for training and then evaluated the resulting models using the French Massive Text Embedding Benchmark (MTEB) (Ciancone et al., 2024; Enevoldsen et al., 2025; Muennighoff et al., 2022).

We re-used the same splits from the CamemBERTaV2 authors and performed hyper-parameter tuning on all models and datasets with 5 seed variations.

4.2 Downstream Results Analysis

The downstream evaluation results summarized in Table 1 show the following insights into model architectures and pretaining dataset effects:

Architecture Impact. Comparing the models trained on identical datasets (ModernBERT-CV2 and CamemBERTaV2/CamemBERTv2), we observe that ModernBERT-CV2 consistently outperforms CamemBERTv2 with the exception of text retrieval, thus showing ModernBERT's improvements over BERT/RoBERTa. However, it fails to surpass CamemBERTaV2 on any non-retrieval task, even though the latter being only trained for a single epoch on the dataset compared to three epochs (1T tokens) for ModernBERT-CV2. This clearly demonstrates that while ModernBERT offers valuable throughput-driven architectural enhancements, these improvements do not match the contextual learning capabilities provided by DeBERTaV3's disentangled attention and RTD-based pretraining objective. Our results also confirm the observation that DeBERTaV3 fails on text embedding tasks. Despite its strong performance on NLU tasks, its sentence representations are poorly structured in the embedding space.

Data Quality Impact. Interestingly, switching to our high-quality filtered dataset (ModernBERT-HQ) only marginally improved performance on downstream tasks, despite the dataset containing three times more unique

[2]We use the publicly available ModernBERT codebase https://github.com/AnswerDotAI/ModernBERT

[3]See pretraining hyperparameter details in Appendix B

Model	NER	QA		CLS	PAWS-X	XNLI	MTEB
	F1	F1	EM	Acc	Acc	Acc	Avg
CamemBERTV2	91.99±0.96	80.39±0.36	61.35±0.39	95.07±0.11	92.00±0.24	81.75±0.62	**51.67±0.57**
CamemBERTaV2	**93.40±0.62**	**83.04±0.19**	**64.29±0.31**	**95.63±0.16**	**93.06±0.45**	**84.82±0.54**	31.15±5.26
ModernBERT-CV2	92.03±0.14	81.34±0.35	61.47±0.46	95.18±0.20	92.79±0.22	83.28±0.34	49.44±1.36
ModernBERT-HQ	91.80±0.47	81.11±0.26	62.07±0.44	95.04±0.09	92.55±0.54	83.66±0.67	49.93±0.60

Table 1: Downstream tasks results. **Bold** indicates best score overall while underline indicates best score between the ModernBERT models. **ModernBERT-CV2** is the ModernBERT model trained on the same data as CamemBERTaV2 while **ModernBERT-HQ** is the one trained on the high-quality filtered dataset. *Scores are the 5-seed average of the best performing set of hyperparamters for each model. MTEB scores are the average over all tasks. Full MTEB scores are available in Table 6*

tokens than the original CamemBERTaV2 dataset. ModernBERT-HQ slightly outperformed ModernBERT-CV2 on QA (FQuad), CLS, XNLI and text retrieval tasks, but improvements remained within small margins. This limited gain suggests two potential explanations: either current transformer architectures exhibit diminishing returns when exposed to additional data beyond a certain threshold, or standard French benchmarks are becoming saturated and unfit to measure model quality with further improvements in model performance. The latter possibility stresses the need for more challenging and diverse benchmarks that can effectively capture the improvements brought by higher-quality data.

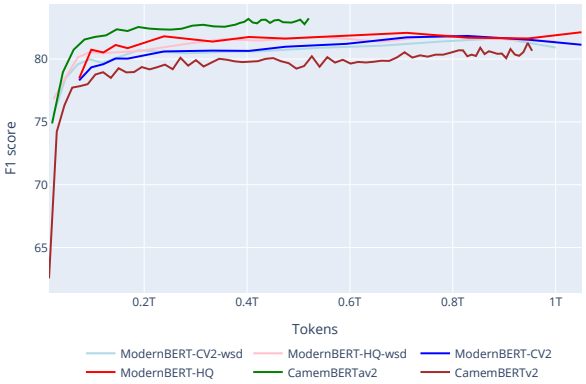

Figure 1: Downstream Performance on QA throughout the pre-training stage. *wsd* are the models tested before the cooldown period.

4.3 Pre-training Dynamics and Sample Efficiency

We further explored the learning trajectories of the various models by evaluating intermediate checkpoints on QA (FQuad) and NER tasks. This analysis offers a more detailed view of the training dynamics and sample efficiency:

Architectural Efficiency. The training curves (shown in Figures 1 and 2) indicate that CamemBERTaV2 reaches higher performance significantly earlier in training compared to ModernBERT-CV2. The DeBERTaV3-based model's faster improvement rate strongly suggests its better sample efficiency is due to optimizations like RTD and gradient-disentangled embedding sharing (GDES). Moreover, in scenarios where pre-training data is limited or scarce, its architectures might be more advantageous.

Impact of Data Quality on Convergence. When comparing ModernBERT-CV2 and ModernBERT-HQ downstream performance throughout the training on the challenging QA tasks 1, we observed that the model trained on the higher-quality dataset achieved its performance plateau faster, indicating that improved data quality enhances training efficiency and accelerates convergence. Yet, it does not substantially increase the final task-specific performance scores, further confirming the hypothesis of saturation effects on standard NLP benchmarks.

Task-specific Dynamics. The intermediate checkpoints downstream score shows a clear difference in learning dynamics between the QA and NER tasks. While QA scores continued to improve gradually throughout training for all models, NER performance plateaued relatively early, with minimal further improvements, except for the CamemBERTaV2 NER scores which increased steadily. This difference suggests that the disentangled attention mechanism, which separately encodes content and positional

Model	NER	QA		CLS	PAWS-X	XNLI	MTEB
	F1	F1	EM	Acc	Acc	Acc	Avg
ModernBERT-CV2	92.03±0.14	81.34±0.35	61.47±0.46	95.18±0.20	92.79±0.22	83.28±0.34	49.49±1.36
ModernBERT-HQ	91.80±0.47	81.11±0.26	62.07±0.44	95.04±0.09	92.55±0.54	83.66±0.67	49.93±0.60
ModernBERT-CV2-final	92.17±0.48 ↑	81.68±0.46 ↑	62.00±0.53 ↑	94.86±0.16 ↓	92.71±0.39 –	82.85±0.45 ↓	48.79±0.45 ↓
ModernBERT-HQ-final	91.33±0.27 ↓	82.19±0.46 ↑	62.66±0.79 ↑	94.92±0.06 ↑	92.52±0.36 –	83.62±0.67 –	49.29±0.81 ↓

Table 2: Downstream tasks results after context extension and cooldown. ↑, ↓, and – indicate an increase, decrease or no change in scores after continual pretraining. *Scores are the 5-seed average of the best performing set of hyperparamters for each model.*

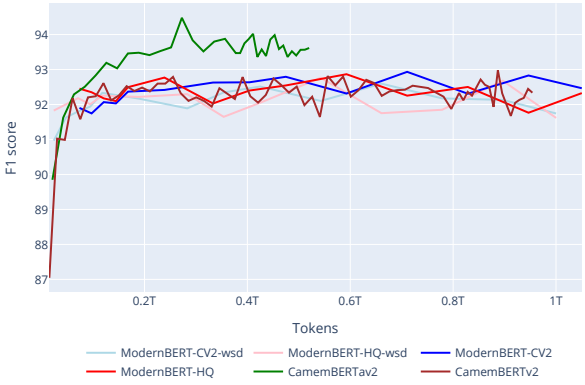

Figure 2: Downstream Performance on NER throughout the pre-training stage. *wsd* are the models tested before the cooldown period.

Model	Context	MLDR (NDCG@10)	
		Max	Avg
CamemBERTV2	1024	32.59	28.37±2.77
CamemBERTaV2	1024	2.44	00.91±1.08
ModernBERT-CV2	1024	21.45	10.39±4.83
ModernBERT-CV2-final	8192	26.93	22.59±2.73
ModernBERT-HQ	1024	31.76	25.80±1.99
ModernBERT-HQ-final	8192	**39.07**	**34.32**±5.44

Table 3: Maximum and highest 5-seed averaged NDCG@10 score for the Multi Long Doc Retrieval task.

embeddings, provides an advantage on token-level tasks such as NER.

4.4 Context Length Extension and Final Model Release

One of ModernBERT's advantages is supporting longer context length due to it's more efficient attention implementation and alternation of local and global attention layers. On the other hand, the older models had limited context length due to the high memory usage of their attention layer implementation. Hence, in order to study the effect of context length extension, we continue our ModernBERT's pretraining, as in the original model's strategy, and increase its maximum input length to 8,192 tokens. This phase also includes a cooldown stage, during which the learning rate is gradually reduced to zero over high-quality, long-context data.

To support this phase, we curated two dataset variants:

- **Long-Context Subset:** We filtered documents longer than 2,048 tokens and retained them fully. Shorter documents were retained with a 10% probability to preserve some distributional diversity.
- **High-Quality Long-Context Subset:** For this version, we upsampled high-quality, long-form sources such as French Wikipedia and academic literature, while only retaining documents rated as "high quality" by our semantic filter within the HQ dataset.

We resumed training for both model variants, the one trained on the CamemBERTaV2 dataset and our High-Quality dataset, using their corresponding long-context subsets. Training was done for an additional 150 billion tokens to extend context capabilities, using a fixed learning rate of 3×10^{-4}. This was followed by a final 100 billion token cooldown phase, during which the learning rate was linearly decayed to zero.

Impact on Downstream Performance. We observe in Table 2, that ModernBERT-CV2 provides modest gains in NER (+0.14 F1) and QA (+0.34 F1 / +0.53 EM), while performance slightly decreases on classification (CLS: –0.32 Acc, XNLI: –0.43 Acc), with PAWS-X remaining stable. Meanwhile, ModernBERT-HQ-final displays clear improvements in QA (+1.08 F1 / +0.59 EM) and CLS (+0.88 Acc), while

maintaining stable results on PAWS-X and XNLI. Although NER and text retrieval performance drops slightly (–0.47 F1 and -0.64), the overall trend indicates that high-quality long-context pretraining primarily benefits tasks requiring deeper semantic understanding or longer-range dependencies.

To better understand the models' long context extrapolation, we evaluate all models trained on the French STS dataset from earlier on the French test subset of the Multi Long Doc Retrieval (MLDR) and present the results in Table 3. The results clearly demonstrate the importance of the long-context pretraining stage. As expected from the MTEB evaluation, CamemBERTaV2 performs poorly on this retrieval benchmark, further highlighting the unsuitability of its architecture for sentence embedding tasks. Both ModernBERT variants show remarkable improvements after the final training phase, with context-extended ModernBERT-HQ achieving the highest max and average score.

However, the performance of ModernBERT-CV2 (the ModernBERT variant trained on the original CamemBERTaV2 dataset) is unexpectedly poor, scoring significantly lower than both CamemBERTV2 and its counterpart trained on our high-quality dataset, ModernBERT-HQ. We currently lack a definitive explanation for this gap. One hypothesis relates to the fundamental differences between the two datasets used in our study. The original CamemBERTaV2 dataset (275B tokens) consists primarily of web-crawled content from CulturaX-FR (constructed from OSCAR and mC4 snapshots). In contrast, our high-quality filtered dataset (1T tokens) underwent extensive semantic filtering using FineWeb-Edu methodology, perplexity-based selection, and full deduplication on the RedPajamaV2 corpus, resulting in a more coherent and diverse collection of texts. The original dataset's heavy reliance on statistical filters may lack the semantic coherence necessary for learning robust long-form representations, whereas our filtered dataset's emphasis on high-quality, diverse sources appears better suited for supporting complex semantic understanding required in retrieval tasks.

4.5 Downstream Training Stability

During fine-tuning on downstream tasks, we observed differences in training stability between the newer and older model families. We had several cases where only ModernBERT variants failed to converge on the FQuAD question-answering task, as illustrated in Figure 3.

Furthermore, during hyperparameter tuning of the final checkpoint, we found the newer architecture to be particularly sensitive to learning rate choices. [4] Despite hyperparameter tuning, the instability persisted. Further investigation revealed that NaN values appear in the loss from the very first training batch. This strongly supports our hypothesis of a numerical instability in the underlying implementation (e.g., FlashAttention) rather than a simple hyperparameter mismatch.

4.6 Training Efficiency

In addition to model accuracy, training efficiency is a crucial factor in practice, impacting both resource costs and environmental footprint. For pretraining time, our ModernBERT training required 1300 H100 GPU-hours to complete one trillion tokens. On the other hand, CamemBERTv2 took roughly 2100 GPU-hours to train on the same dataset size while CamemBERTaV2 required around 2700 GPU-hours to complete just a single epoch, despite processing one-third of the tokens (275B). This clearly demonstrates ModernBERT's efficient architecture advantages and its practical edge during training. However, it should be noted a significant portion of the speedup over DeBERTaV3-based models comes from engineering optimizations such as unpadding and FlashAttention, both of which are not implemented in the DeBERTa models at the time of this study.

The key takeaway from these experiments is the trade-off between ModernBERT, which offers significantly faster training and inference speeds, making it more efficient for time-sensitive applications, and DeBERTa, which delivers higher raw performance through its most effective use of training data.

5 Discussion

Confirming recent results (Warner et al., 2024; Breton et al., 2025), our experiments showed a weakness in DeBERTaV3, which failed at the text retrieval task, despite its strong performance across other benchmarks. We hypothesize that its architecture may lack the mechanisms to build a global document representation suitable for retrieval using pooling of token embeddings. In

[4] The search space included multiple learning rates (5e-5 to 5e-4), schedulers, batch sizes, and seeds.

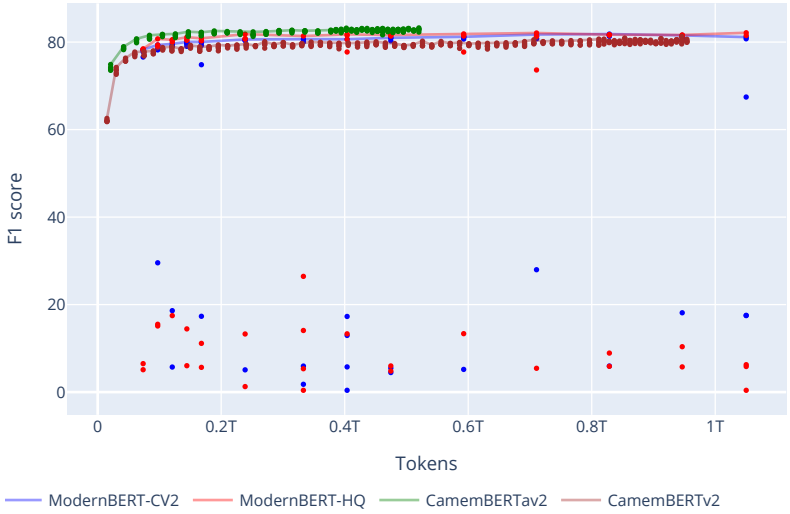

Figure 3: Instances of divergence during QA fine-tuning. *Colored lines illustrate the maximum score at a given step.*

contrast, ModernBERT performed well on this task, suggesting its alternating global and local attention layers provide a suitable architecture for text retrieval.

However, the strengths of ModernBERT in text retrieval did not extend to all tasks. Our experiments showed a deficiency in the ModernBERT question-answering experiments. We suspect that ModernBERT's architecture, may face challenges in learning the long-range dependencies that are needed for QA tasks, particularly when informative signals are sparse. For instance, the model needs to propagate the signal from the correct answer start/end token across long distances to the question context, which resides at the beginning of the input. We hypothesize that while local attention layers are efficient, they might occasionally interrupt the direct propagation path for such sparse signals. The global attention layers are then tasked with bridging these segments, leading to potential difficulties in predicting the answer span based on the distant question. Other tasks such as NER, which rely on token-level prediction, do not require long-range dependencies since the model can infer the correct label based on local context, in addition to the richer learning signal, since named entity tokens are more frequent than a single start/end token per example.

Our work contributes to the renewed debate on encoder architectures versus the prevailing trend of using decoder-only models for all tasks (see (Gisserot-Boukhlef et al., 2025) and references therein). This discussion centers on whether a single, large architecture can be adapted for any task, or if specialized models remain superior for certain domains like NLU. The recent paper by Weller et al. (2025) offers new evidence by training paired encoder and decoder models under the same conditions. Their experiments confirm that encoders are better suited for classification and retrieval, which aligns with our results showing ModernBERT's strength in text retrieval. Furthermore, the paper demonstrates that adapting a model to a task for which it was not designed is an ineffective strategy. This puts our own results into a broader context, suggesting that specialized encoders are not obsolete, and that architectural choice remains a key factor for task performance.

Looking ahead, future work could explore integrating recent efficiency improvements such as *FlashDeBERTa*[5], which applies FlashAttention to the disentangled attention mechanism and greatly reduces DeBERTa's memory and latency costs. Another promising direction is investigating why DeBERTaV3 fails so strongly on embedding-based tasks. Examining its pooling mechanisms and representation geometry may help clarify the limits of disentangled attention for retrieval-oriented objectives.

6 Conclusion

We set out in this work to critically evaluate the claims made in the original ModernBERT paper by reproducing its setup under tightly controlled conditions. We isolated the authors' contributions by retraining their model under the same conditions

[5] https://github.com/Knowledgator/FlashDeBERTa

as the previous state-of-the-art models, to assess their actual impact on training dynamics, efficiency, and downstream performance.

Our findings show that while ModernBERT does offer improvements in training and inference speed compared to older architectures, these do not translate into better sample efficiency or task performance under matched conditions. In fact, under careful evaluation, we found that DeBERTaV3's architecture and training objectives are more advantageous in low-data scenarios or if the goal is to get the absolute best task performance, except for the case of text retrieval where DeBERTaV3 fails completely.

We also observed that increasing the size and quality of pretraining data only yielded marginal gains for the newly proposed architecture, suggesting that current benchmarks may be reaching saturation, or at least they are insufficiently sensitive to capture finer improvements. During fine-tuning, we faced a problem with sensitivity to hyperparameters, which the V2 baselines did not have. These stability concerns present challenges for reproducibility and deployment, and deserve further investigation.

In summary, ModernBERT offers a fast and efficient alternative for scenarios where training and inference speed are critical, but DeBERTaV3 remains the stronger choice when performance and sample efficiency are required. Our study reinforces the importance of evaluating models under shared conditions to truly understand the contributions of architecture, training data, and design choices.

Limitations

Our study has several important limitations. First, we observe that ModernBERT exhibits training instability during fine-tuning, this might be due to numerical instabitlity of the flashattention implementation. Second, our downstream evaluation relies on established NLP benchmarks that may be reaching saturation, potentially masking more nuanced performance differences between architectures. Finally, our analysis focuses on base-sized models, and the relative performance characteristics may differ for larger model variants. Future work should address these limitations through stability analysis, development of more discriminative benchmarks, and evaluation across different model scales.

Ethics Considerations

This work involves training large-scale language models using publicly available data, with special attention given to data quality, filtering, and documentation. We applied both heuristic and semantic filters to reduce harmful, biased, or low-quality content. Nonetheless, we acknowledge that pretrained models may still reflect societal biases present in the underlying data. We encourage responsible use of our models and welcome future research focused on auditing and mitigating bias and potential misuse.

Acknowledgments

This work has received partial funding Benoît Sagot and Djamé Seddah's chairs in the PRAIRIE-PSAI, funded by the French national agency ANR, as part of the "France 2030" strategy under the reference ANR-23-IACL-0008. This project also received funding from the BPI Scribe project. The authors extend their gratitude to the OPAL infrastructure of Université Côte d'Azur for providing essential resources and support. This work was also granted access to the HPC resources of IDRIS by GENCI under the allocation 2024-GC011015610 and AD011013900R2. Special thanks to Nathan Godey and Francis Kulumba for their assistance with training code and for engaging in productive discussions.

All pretraining dataset are sourced from publicly available and wildly used corpora (CultraX from OSCAR [CC0 1.0], and mc4 [odc-by], HAL's open archive [CC], Wikipedia [cc-by-sa-3.0], RedPajamaV2 [Common Crawl Permissive license]). Downstream dataset are all from standard open benchmarks. ModernBERT's codebase is Apache-2.0. CamemBERTaV2, ModernCamembert checkpoints and models are released under the MIT licence.[6]

References

Julien Abadji, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot. 2022. Towards a cleaner document-oriented multilingual crawled corpus. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4344–4355, Marseille, France. European Language Resources Association.

[6] https://huggingface.co/collections/almanach/moderncamembert

Julien Abadji, Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2021. Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-9) 2021. Limerick, 12 July 2021 (Online-Event), pages 1 – 9, Mannheim. Leibniz-Institut für Deutsche Sprache.

Anne Abeillé, Lionel Clément, and Alexandra Kinyon. 2000. Building a treebank for French. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'00)*, Athens, Greece. European Language Resources Association (ELRA).

Wissam Antoun, Francis Kulumba, Rian Touchent, Éric de la Clergerie, Benoît Sagot, and Djamé Seddah. 2024. Camembert 2.0: A smarter french language model aged to perfection. *arXiv preprint arXiv:2411.08868*.

Wissam Antoun, Benoît Sagot, and Djamé Seddah. 2023. Data-efficient french language modeling with camemberta. In *Findings of the Association for Computational Linguistics: ACL 2023*, Toronto, Canada. Association for Computational Linguistics.

Nicolas Boizard, Hippolyte Gisserot-Boukhlef, Duarte M Alves, André Martins, Ayoub Hammal, Caio Corro, Céline Hudelot, Emmanuel Malherbe, Etienne Malaboeuf, Fanny Jourdan, and 1 others. 2025. Eurobert: scaling multilingual encoders for european languages. *arXiv preprint arXiv:2503.05500*.

Lola Le Breton, Quentin Fournier, Mariam El Mezouar, John X. Morris, and Sarath Chandar. 2025. Neobert: A next-generation bert. *Preprint*, arXiv:2502.19587.

Marie Candito and Benoît Crabbé. 2009. Improving generative statistical parsing with semi-supervised word clustering. In *Proceedings of the 11th International Conference on Parsing Technologies (IWPT'09)*, pages 138–141, Paris, France. Association for Computational Linguistics.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Mathieu Ciancone, Imene Kerboua, Marion Schaeffer, and Wissam Siblini. 2024. Mteb-french: Resources for french sentence embedding evaluation and analysis. *Preprint*, arXiv:2405.20468.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Tri Dao. 2024. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations*.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35:16344–16359.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Martin d'Hoffschmidt, Maxime Vidal, Wacim Belblidia, and Tom Brendlé. 2020. FQuAD: French Question Answering Dataset. *arXiv e-prints*, arXiv:2002.06071.

Kenneth Enevoldsen, Isaac Chung, Imene Kerboua, Márton Kardos, Ashwin Mathur, David Stap, Jay Gala, Wissam Siblini, Dominik Krzemiński, Genta Indra Winata, Saba Sturua, Saiteja Utpala, Mathieu Ciancone, Marion Schaeffer, Gabriel Sequeira, Diganta Misra, Shreeya Dhakal, Jonathan Rystrøm, Roman Solomatin, and 67 others. 2025. Mmteb: Massive multilingual text embedding benchmark. *arXiv preprint arXiv:2502.13595*.

Hippolyte Gisserot-Boukhlef, Nicolas Boizard, Manuel Faysse, Duarte M. Alves, Emmanuel Malherbe, André F. T. Martins, Céline Hudelot, and Pierre Colombo. 2025. Should we still pretrain encoders with masked language modeling?

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021a. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021b. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

Francis Kulumba, Wissam Antoun, Guillaume Vimont, and Laurent Romary. 2024. Harvesting textual and structured data from the hal publication repository. *Preprint*, arXiv:2407.20595.

Hang Le, Loïc Vial, Jibril Frej, Vincent Segonne, Maximin Coavoux, Benjamin Lecouteux, Alexandre Allauzen, Benoit Crabbé, Laurent Besacier, and Didier Schwab. 2020. FlauBERT: Unsupervised language model pre-training for French. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France. European Language Resources Association.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Guha, Sedrick Scott Keh, Kushal Arora, and 1 others. 2024. Datacomp-lm: In search of the next generation of training sets for language models. *Advances in Neural Information Processing Systems*, 37:14200–14282.

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

Davis Liang, Hila Gonen, Yuning Mao, Rui Hou, Naman Goyal, Marjan Ghazvininejad, Luke Zettlemoyer, and Madian Khabsa. 2023. Xlm-v: Overcoming the vocabulary bottleneck in multilingual masked language models. *Preprint*, arXiv:2301.10472.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

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

Philip May. 2021. Machine translated multilingual sts benchmark dataset.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2022. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*.

James O' Neill, Santhosh Subramanian, Eric Lin, Abishek Satish, and Vaikkunth Mugunthan. 2024. Guardformer: Guardrail instruction pretraining for efficient safeguarding. In *Neurips Safe Generative AI Workshop 2024*.

Thuat Nguyen, Chien Van Nguyen, Viet Dac Lai, Hieu Man, Nghia Trung Ngo, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Culturax: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. *Preprint*, arXiv:2309.09400.

Zach Nussbaum, John X. Morris, Brandon Duderstadt, and Andriy Mulyar. 2025. Nomic embed: Training a reproducible long context text embedder. *Preprint*, arXiv:2402.01613.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Proceedings of the Workshop on Challenges in the Management of Large Corpora (CMLC-7) 2019. Cardiff, 22nd July 2019, pages 9 – 16, Mannheim. Leibniz-Institut für Deutsche Sprache.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Preprint*, arXiv:2406.17557.

Jacob Portes, Alexander Trott, Sam Havens, Daniel King, Abhinav Venigalla, Moin Nadeem, Nikhil Sardana, Daya Khudia, and Jonathan Frankle. 2023. Mosaicbert: A bidirectional encoder optimized for fast pretraining. *Advances in Neural Information Processing Systems*, 36:3106–3130.

Benoît Sagot, Marion Richard, and Rosa Stern. 2012. Annotation référentielle du corpus arboré de Paris 7 en entités nommées (referential named entity annotation of the Paris 7 French TreeBank) [in French]. In *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2: TALN*, pages 535–542, Grenoble, France. ATALA/AFCP.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. 2024. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37:68658–68685.

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

Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2025. Nemotron-cc: Transforming common crawl into a refined long-horizon pretraining dataset. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2459–2475.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. 2021. Roformer: Enhanced transformer with rotary position embedding. *Preprint*, arXiv:2104.09864.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, and 1 others. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, and 1 others. 2025. Embeddinggemma: Powerful and lightweight text representations. *arXiv preprint arXiv:2509.20354*.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini,

Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, and 1 others. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*.

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. 2024. Redpajama: an open dataset for training large language models. *NeurIPS Datasets and Benchmarks Track*.

Orion Weller, Kathryn Ricci, Marc Marone, Antoine Chaffin, Dawn Lawrie, and Benjamin Van Durme. 2025. Seq vs seq: An open suite of paired encoders and decoders. *Preprint*, arXiv:2507.11412.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, and 1 others. 2024. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412.

A Quality Labeling Prompt

Prompt used to annotate text quality

```
Below is an extract from a web page. Evaluate the quality of the content based
on the following factors:

    1. Content Accuracy: Assess the correctness and reliability of the
    information presented. Consider the factual accuracy, use of credible
    sources (if mentioned), and absence of misinformation.
    2. Clarity: Evaluate how well the information is communicated. Look for
    clear explanations, well-defined terms, and logical flow of ideas.
    3. Coherence: Analyze the overall structure and organization of the
    content. Consider how well ideas are connected and if the content follows
    a logical progression.
    4. Grammar and Language: Assess the quality of writing, including correct
    grammar, spelling, and punctuation. Consider the appropriateness of
    language for the intended audience.
    5. Depth of Information: Evaluate the level of detail and thoroughness of
    the content. Consider whether it provides surface-level information or
    delves into more comprehensive explanations.
    6. Overall Usefulness: Assess the practical value and relevance of the
    information for a general audience. Consider how applicable or helpful the
    content would be for someone seeking information on the topic.

Based on these factors, give an overall quality score of low, medium, or high.

The extract:
    {input}

After examining the extract:
- Briefly justify your quality classification, up to 100 words on one line
using the format: "Explanation: <justification>"
- Conclude with the quality classification using the format: "Quality score:
<classification>" (on a separate line)

Remember to assess from the AI Assistant perspective, utilizing web search
knowledge as necessary. Evaluate the content based on the quality factors
outlined above.
```

B Pretraining Details and Hyperparamters

We closely follow the original ModernBERT recipe. We present the model parameters in Table 4 and pretraining hyperparamters in Table 5

C Detailed MTEB Scores

Parameter	Base
Vocabulary	32,768
Unused Tokens	418
Layers	22
Hidden Size	768
Transformer Block	Pre-Norm
Activation Function	GeLU
Linear Bias	False
Attention	Multi-head
Attention Heads	12
Global Attention	Every three layers
Local Attention Window	128
Intermediate Size	1,152
GLU Expansion	2,304
Normalization	LayerNorm
Norm Epsilon	1e-5
Norm Bias	False
RoPE theta	160,000
Local Attn RoPE theta	10,000

Table 4: ModernBERT model parameters

	Pretraining	Context Extension	Context Ext & High Quality
Training Tokens	1 trillion	150 billion	100 billion
Max Sequence Length	1,024	8,192	8,192
Batch Size	4,608	768	768
Warmup (tokens)	None		
Microbatch Size	96	8	8
Learning Rate	8e-4	3e-4	3e-4
Schedule	Trapezoidal	Trapezoidal	1-sqrt (50B tokens delayed)
Warmup (tokens)	3 billion	None	None
Weight Decay	1e-5		
Training Time (hours)	22.7	4.5	4
Model Initialization	Megatron	-	-
Dropout (attn out)	0.1		
Dropout (all other layers)	0.0		
Optimizer	DecoupledAdamW		
Betas	(0.90, 0.98)		
Epsilon	1e-06		
Training Hardware	48 GPUs - 12x(4xH100)		
Training Strategy	FSDP - Full Sharding		
Software Libraries	PyTorch 2.5.1, Cuda 12.4, Composer 0.28, Flash Attention 2.6.3-Hopper		

Table 5: Pre-training hyperparameters

	CLUSTERING	CLASSIFICATION	PAIR CLASSIFICATION	RETRIEVAL	RERANKING	STS	SUMMARIZATION	OVERALL
CamemBERTV2	**39.40**$_{\pm 1.67}$	62.40$_{\pm 0.55}$	56.60$_{\pm 0.55}$	**39.40**$_{\pm 0.55}$	**65.20**$_{\pm 1.92}$	**75.60**$_{\pm 0.55}$	31.00$_{\pm 0.71}$	**51.67**$_{\pm 0.57}$
CamemBERTaV2	26.80$_{\pm 1.92}$	41.00$_{\pm 6.04}$	55.60$_{\pm 3.21}$	6.80$_{\pm 2.68}$	35.20$_{\pm 3.83}$	52.00$_{\pm 20.65}$	28.20$_{\pm 1.92}$	31.15$_{\pm 5.26}$
ModernBERT-CV2	39.20$_{\pm 1.79}$	**62.60**$_{\pm 0.55}$	**59.60**$_{\pm 1.14}$	32.20$_{\pm 3.70}$	56.80$_{\pm 2.95}$	74.20$_{\pm 1.30}$	29.80$_{\pm 0.84}$	49.45$_{\pm 1.36}$
ModernBERT-CV2-final	39.20$_{\pm 1.30}$	61.40$_{\pm 0.55}$	59.40$_{\pm 2.07}$	31.20$_{\pm 0.84}$	55.20$_{\pm 0.45}$	73.40$_{\pm 0.55}$	**31.20**$_{\pm 0.84}$	48.79$_{\pm 0.45}$
ModernBERT-HQ	39.20$_{\pm 0.84}$	62.00$_{\pm 0.71}$	57.40$_{\pm 0.89}$	34.20$_{\pm 1.64}$	58.60$_{\pm 1.52}$	75.00$_{\pm 1.22}$	31.00$_{\pm 0.71}$	49.93$_{\pm 0.60}$
ModernBERT-HQ-final	38.40$_{\pm 0.55}$	60.00$_{\pm 0.71}$	57.60$_{\pm 0.89}$	33.80$_{\pm 2.39}$	60.80$_{\pm 2.39}$	74.80$_{\pm 0.84}$	30.00$_{\pm 1.22}$	49.29$_{\pm 0.81}$

Table 6: MTEB task type results. **Bold** indicates best score overall. Scores represent the average performance across all tasks within each task type category with standard deviations.