New Encoders for German Trained from Scratch: Comparing ModernGBERT with Converted LLM2Vec Models

Julia Wunderle[1,*], **Anton Ehrmanntraut**[2,*], **Jan Pfister**[1]
Fotis Jannidis[2,†], **Andreas Hotho**[1,†]
[1]Data Science [2]Computer Philology and History of Contemporary German Literature
CAIDAS – Center for Artificial Intelligence and Data Science
JMU – Julius-Maximilians-Universität Würzburg
{lastname}@informatik.uni-wuerzburg.de, {firstname.lastname}@uni-wuerzburg.de
[*,†]Equal contribution

Abstract

Encoders remain essential for efficient German NLP and NLU scenarios despite the rise of decoder-only LLMs. This work studies two routes to high-quality German encoders under identical data and training constraints: a) training from scratch and b) converting decoders via LLM2Vec. We introduce two resources: ModernGBERT (134M, 1B), fully transparent German encoders in the ModernBERT style, and LLäMmlein2Vec (120M, 1B, 7B), decoder-to-encoder conversions trained with masked next-token prediction, both undergoing a context extension to 8,192 tokens. Across the specialzed German benchmark SuperGLEBer, ModernGBERT 1B sets a new state of the art (avg 0.808), surpassing $\text{GBERT}_{\text{Large}}$ (+4%) and the seven-times larger converted 7B model (0.787). On German MTEB after supervised fine-tuning, ModernGBERT 1B (0.551) approaches the converted 7B model (0.557). We release all models, checkpoints, datasets, and full training records, and introduce an encoder-adapted QA-NIAH evaluation. All in all, our results provide actionable guidance: when parameter efficiency and latency matter, from-scratch encoders dominate. When a pre-trained decoder exists and compute is limited, conversion offers an effective alternative.

Keywords: Language Resources, Encoder-only Models, German, Model Conversion, Long Context, Benchmarking

1. Introduction

Encoders remain central to efficient (German) NLP and NLU where latency, memory, and cost dominate (Pfister and Hotho, 2024), still accounting for most of the downloads from e.g. Hugging Face (Bourdois, 2025). Especially for tasks like sentence similarity (i.e. important for RAG), mask-filling and text/token classification, encoder models are still the architecture of choice (Bourdois, 2025). Yet, the German NLP landscape lacks high-quality, modern, transparent encoders with reproducible training provenance and long-context support. This paper contributes both resources and a controlled study addressing two practical routes to encoders under identical data and training constraints: training from scratch versus converting decoders.

We introduce to complementary model families: ModernGBERT (134M, 1B), ModernBERT-style German encoders trained from scratch, and LLäMmlein2Vec (120M, 1B, 7B), encoders derived from German decoder-only models via LLM2Vec. ModernBERT (Warner et al., 2024) introduced several architectural improvements for English encoders, including improved relative positional embeddings and efficient attention patterns that allow long context processing. We adapt this design to German, providing strong encoder baselines built entirely from scratch.

In parallel, to assess the practical utility and trade-offs of training encoder models from scratch, we converted the decoder-only model family LLäMmlein into encoders using LLM2Vec (BehnamGhader et al., 2024). Specifically, to align closely with the encoder training objective, we limit the procedure to the first two LLM2Vec steps: 1. enabling a bidirectional attention mask, and 2. masked next-token prediction (MNTP). Since all models share the same training datasets, this setup provides a unique foundation for systematically analyzing the relationship between different architectures and training strategies. We extensively evaluate and compare these models during (post) training via: natural language understanding (SuperGLEBer, Pfister and Hotho, 2024), embedding performance (MTEB; Enevoldsen et al., 2025; Muennighoff et al., 2023; Wehrli et al., 2023), long-context understanding (new Question Answering Needle-in-a-Haystack (QA-NIAH)) and an efficiency suite reflecting variable-length inference. Our key contributions are:

- We introduce a ModernGBERT family, which achieves new state-of-the-art performance on SuperGLEBer and the German MTEB.
- To enable comparisons between training strategies, we also introduce a decoder-turned-encoder LLäMmlein2Vec family, based on the same training dataset.
- We find that dedicated encoders trained from scratch consistently outperform converted decoders of similar size.

Proceedings of the Fifteenth Language Resources and Evaluation Conference (LREC 2026), pages 10424–10446
11–16 May 2026. ©ELRA Language Resources Association (ELRA), 2026

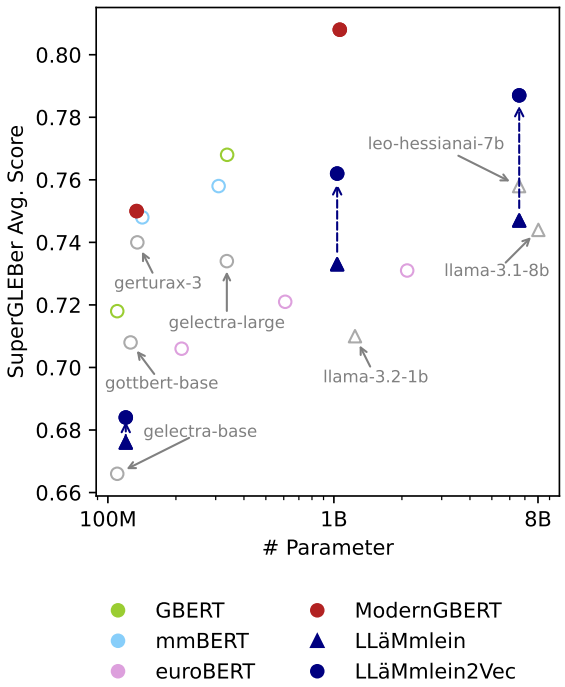

Figure 1: Performance on SuperGLEBer benchmark. • markers: encoders, ▲ markers: decoders. Dashed arrows: LLM2Vec conversion gains. Models of the same family are colored the same.

- Our newly introduced QA-NIAH benchmark confirms strong long-context understanding of our models, validating the effectiveness for extended input sequences.
- To enable further research we release all (intermediary) resources including all model checkpoints, data point tracking, code and data.

Note: Throughout the paper, we highlight interesting findings and insights we gained during the process in little boxes like this one.

2. Datasets

2.1. Pre-Training Dataset

We pre-trained ModernGBERT on the same data as LLäMmlein decoder models (Pfister et al., 2024), using the open-source RedPajamaV2 dataset (Weber et al., 2024). This dataset comprises German CommonCrawl snapshots from 2014 to 2023. As we intend to keep datasets consistent between ModernGBERT and LLäMmlein, we follow LLäMmlein's data pipeline and select the higher quality document-level deduplicated "head" and "middle" partitions, excluding the lower quality "tail" partition. For our 134M model, we only selected the head partition. Our processing pipeline mirrors Pfister et al. (2024): we first perform paragraph-level deduplication using a Bloom filter to remove redundant content such as GDPR notices and web boilerplate, enhancing data diversity. Afterward, we apply a token-to-word ratio filter to further improve text quality. The final dataset is about 6 TB, corresponding to approximately 1.27T tokens using a $\text{GBERT}_{\text{Large}}$ tokenizer.

	Dataset	# Tokens	# Sequences	Median Length
ext1	*LONG-Head*	52B	6,813,019	7,755
	LONG-Head/Middle	90B	11,785,941	8,013
ext2	*High Quality*	14.4B	43,191,271	199
	↳ Fineweb2	7,640M	42,319,173	194
	↳ Fineweb2-long	6,211M	799,296	7,902
	↳ OpenLegalData	407M	53,798	7,583
	↳ Wikipedia	143M	19,004	7,515

Table 1: Composition of post-training datasets for context extension and decoder conversion.

2.2. Context Extension Datasets

The context capacity of ModernBERT is enhanced from 1,024 to 8,192 by fine-tuning in two phases: on an 250B-token subset of 8,192-token sequences from the original pre-training data (*ext1*), followed by 50B-token high-quality dataset with mixed sequence lengths (*ext2*) (Gao et al., 2025).

Following this setup, we construct two German datasets. For *ext1*, we analogously subsample long sequences from our pre-training dataset, resulting in "*LONG-Head*" from the head partition for our 134M model, and "*LONG-Head/Middle*" from the head and middle partition for our 1B model.

For the high-quality dataset used for *ext2*, we selected the German portion of Fineweb2 as basis (Penedo et al., 2024) . To match the original distribution, we take a randomized sample of Fineweb2, and add a separate Fineweb2 sample, selecting long documents with $\geq$8,192 tokens, splitting them into sequences of about 8,192 tokens ("Fineweb2-long"). Additional long documents are drawn from the 2023 German Wikipedia and the 2022 OpenLegalData dump . The resulting dataset contains 14.4B tokens, summarized in Table 1.

3. Methodology

3.1. ModernGBERT

We adapt the ModernBERT architecture and training strategy for German. While ModernGBERT 134M matches the base ModernBERT model size (22 layers, 768 hidden units, but 16M fewer parameters due to a smaller vocabulary size), we create ModernGBERT 1B with 28 layers and a hidden size of 2,048 (See Table 6 for more details).

Pre-/Post-Training Strategy Both models are pre-trained using masked language modeling (MLM) with no next-sentence-prediction, a 30% masking rate, and sequences up to 1,024 tokens (10,000 RoPE theta). ModernGBERT 134M is trained on the head partition (0.47T tokens), as downstream evaluation indicated early saturation, while ModernGBERT 1B is trained on both the head and middle partitions of our pre-training corpus (Section 2.1), totaling 1.27T tokens.

After standard MLM pre-training, we proceed with the two context extension phases: During *ext1*, the RoPE theta is raised to 160,000, and models are trained on the *LONG-Head* (134M) or *LONG-Head/Middle* (1B) datasets (see Section 2.2). In the *ext2* phase, both models are trained on the *High Quality* dataset.

Tokenization For tokenization, we use the original BERT-style tokenizer from $\text{GBERT}_{\text{Large}}$, resulting in a 31,168-word embedding layer. While LLäMmlein (Pfister et al., 2024) provides a dedicated German BPE tokenizer, our preliminary ablations using this tokenizer consistently showed degraded downstream performance. Consequently, we retained the $\text{GBERT}_{\text{Large}}$ tokenizer.

Training Progress Tracking Throughout the training we save, evaluate and release all checkpoints to support further research. Inspired by Pythia (Biderman et al., 2023) we provide full training transparency by logging and releasing the order of data points seen during training; thus, all checkpoints can be linked with the exact data points seen up to that checkpoint.

3.2. LLM2Vec: Turning Decoders to Encoders

BehnamGhader et al. (2024) proposes a method to convert decoder-only LLMs into text encoders through the following steps: First, the causal attention mask is replaced with a full attention mask, enabling bidirectional attention across tokens. Second, the model is trained using a masked next token prediction (MNTP) objective. Third, regular LLM2Vec includes a unsupervised contrastive learning (SimCSE) step, improving embedding quality by maximizing agreement between differently dropped-out versions of the same input. However, we intend to remain as close as possible to the training objectives of ModernGBERT to allow fair and direct comparisons. Therefore, we limit the procedure to the first two steps of LLM2Vec, specifically employing the MNTP objective, which is most closely aligned with MLM.

Pre-/Post-Training Strategy We train all three LLäMmlein models (120M, 1B, 7B) using the same two context extension datasets as employed by ModernGBERT's context extensions (*ext1* and *ext2*) (Section 2.2). Specifically, the 120M model mirrors ModernGBERT 134M, being trained on the *LONG-Head* (*ext1*) and *High Quality* (*ext2*) datasets, while the 1B and 7B models follow ModernGBERT 1B, using *LONG-Head/Middle* (*ext1*) and the *High Quality* dataset (*ext2*). Notably, MNTP training is applied separately to each dataset, resulting in two adapter modules for each model. Additionally, all adapters were trained on the full corresponding datasets[1], although models were able to achieve comparable results during earlier training stages. This indicates, that compute time could have been drastically reduced.

Further aligning with ModernGBERT we combined MNTP with context extension — a setup that, to our knowledge, has not yet been widely explored. Therefore, we also extended the model sequence length to 8,192 tokens and increased RoPE theta to 160,000. We evaluate both individual adapters (*ext1* & *ext2*) as well as a merged model (*ext1+2*) that combines both

4. Evaluation Setup

4.1. SuperGLEBer

We assess our final models using the German SuperGLEBer benchmark (Pfister and Hotho, 2024), which includes 29 tasks across text classification, sequence tagging, question answering, and sentence similarity. These tasks cover diverse domains such as news, legal texts, and consumer reviews. For each task, models are fine-tuned with QLoRA (Dettmers et al., 2023) by default, or LoRA as fallback. In addition to evaluating final checkpoints, we follow LLäMmlein (Pfister et al., 2024) and evaluate intermediate checkpoints of ModernGBERT as well as LLäMmlein2Vec on the same representative SuperGLEBer subset consisting of: the classification tasks NLI (Conneau et al., 2018), FactClaiming Comments (Risch et al., 2021), DB Aspect (Wojatzki et al., 2017), and WebCAGe (Henrich et al., 2012), the sequence tagging task EuroParl (Faruqui and Padó, 2010), and the sentence similarity task PAWSX (Liang et al., 2020).

[1] Except for the LLäMmlein2Vec 7B trained on the *LONG-Head/Middle* model, which we trained on 64 nodes with 4 H200 each for 14 hours, before stopping the training due to compute constraints. We expect longer training to mitigate the drop in long-context performance.

4.2. Massive Text Embedding Benchmark

We further evaluate the models on the German subset of the Massive Text Embedding Benchmark *MTEB(deu,v1)* (Enevoldsen et al., 2025), comprising 19 tasks. In addition to text pair classification and semantic textual similarity, which are already covered by the SuperGLEBer benchmark, MTEB includes clustering (Wehrli et al., 2023), as well as reranking and retrieval tasks. These latter tasks provide a more comprehensive assessment of general-purpose sentence embeddings, focusing on the models' ability to produce robust semantic representations.

To adapt the base models for embedding tasks, we fine-tune them using the Sentence-Transformer framework (Reimers and Gurevych, 2019) in a supervised setup. Fine-tuning employs 10,000 samples from the German portion of the machine-translated multilingual mMARCO passage ranking dataset (Bonifacio et al., 2022), maximizing similarity between query and positive passages, while minimizing similarity to negative passages. Sentence embeddings are obtained by mean pooling over the final token representations. We use InfoNCE loss with a batch size of 128 and a learning rate of 5×10^{-5}. We apply QLoRA for efficient training (falling back to LoRA for the GBERT family, where quantization is not supported).

4.3. Long-Context Understanding

Evaluating long-context capabilities in German is hindered by the scarcity of native high-quality datasets, with translations from English often introducing artifacts. To address this, we construct a new Question-Answering Needle-In-a-Haystack (QA-NIAH) evaluation (Ivgi et al., 2023; Hsieh et al., 2024) based on the human-annotated GermanQuAD dataset (Möller et al., 2021). To our knowledge, this is the first QA-NIAH evaluation specifically adapted for encoder models. The goal in QA-NIAH is to extract an answer span from a long document. We adapt GermanQuAD to a QA-NIAH setup as follows: for each question-paragraph ("needle") pair, we sample up to 3 distractor paragraphs and shuffle them with the needle, forming a "haystack" document of up to 1,024 tokens. The answer always appears only in the needle paragraph. For evaluation, up to 20 distractors are included to test generalization to longer contexts, yielding documents up to 8,192 tokens. This results in 11,518 training and 2,204 test question–haystack pairs. Models were again fine-tuned using QLoRA falling back to LoRA, where quantization is not supported. We will release the QA-NIAH generation code and splits to facilitate reuse and extension.

5. Evaluation Results

The tables in this paper present only a summary of our results. The Appendix contains extended benchmarking results and complete hyperparameter sets for all models.

5.1. Intermediate Model Evaluation

ModernGBERT Training To track the pretraining progress, we evaluated intermediate checkpoints on the SuperGLEBer benchmark (see Section 4.1). Figure 2 exemplary shows the results for the ModernGBERT 1B model in the top row. To quantify trends, we first assessed the full SuperGLEBer suite on five 1B and respective three 134M equally spaced ModernGBERT checkpoints and performed Wilcoxon signed-rank tests. The 134M model plateaued after 72B tokens (15% of data), with no further significant gains. In contrast, the 1B model achieved significant improvements after the same amount of data ($p < 0.0001$) and on the middle partition ($p < 0.00052$), before plateauing after 864B tokens (67% of data), with only minor gains thereafter ($0.777 \rightarrow 0.791$)

To gain more insights, we tracked six representative SuperGLEBer subtasks at each checkpoint. For the 134M model, only PAWSX showed a positive Spearman rank correlation with training duration ($r = 0.655$; $p < 0.003$), whereas for 1B, all tasks except EuroParl improved significantly ($r > 0.57$; $p < 0.00014$), and complex tasks like PAWSX continued to show modest gains even after the aggregate score stabilized (gray in Figure 2).

These saturation patterns, including per-task trends and overall performance plateaus, are consistent with findings by Pfister et al. (2024) for decoder models, and by Antoun et al. (2024) for the French ModernBERT variant ModernCamemBERT (136M parameters). Our results confirm that while small ModernBERT models saturate quickly, larger models continue to benefit from additional data. Extrapolating from this observed scaling behaviour between ModernGBERT 134M and 1B, we hypothesize that a larger 7B encoder could leverage extensive monolingual corpora to achieve performance beyond ModernGBERT 1B.

Confirmation: Our findings corroborate Antoun et al. (2024) and Pfister et al. (2024): small ModernGBERT models reach saturation early, while scaling model and dataset size enables improvements.

LLäMmlein2Vec Conversion To remain as consistent as possible with the ModernGBERT training procedure, we trained the LLäMmlein2Vec conversions on the same post-training datasets. To assess whether training on the full dataset was neces-

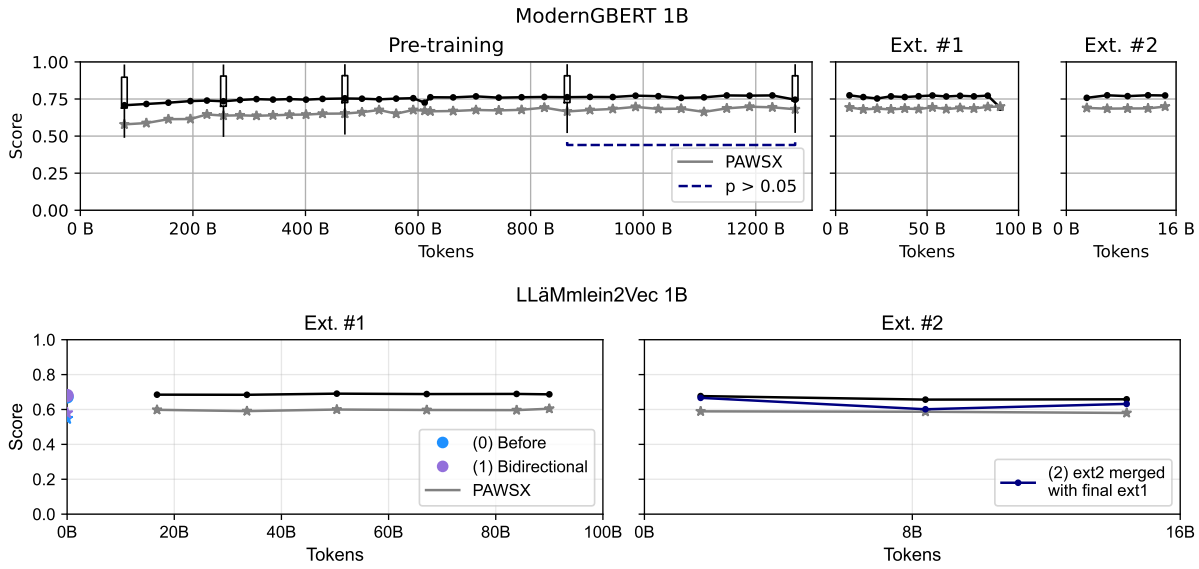

Figure 2: Intermediate checkpoint evaluation. The solid black line shows the mean of six SuperGLEBer tasks (NLI, FactClaiming Comments, DB Aspect, WebCAGe, EuroParl, PAWSX Similarity). The top figure shows ModernGBERT 1B across pre-training and two context extension phases, with box plots representing all 29 SuperGLEBer tasks. For simplicity, no significant improvements between checkpoint pairs are marked with brackets (Wilcoxon signed-rank test). All other pairs of box plots show significant improvements of at least $p < 0.01$. The bottom figure shows LLäMmlein 1B after LLM2Vec conversion, including the starting point in (blue) and the average score after switching the mask (purple). Checkpoints on *ext2* are shown alone and merged with the last *ext1* checkpoint.

sary, we evaluated multiple checkpoints throughout the entire training process on the six SuperGLEBer tasks. In the bottom half of Figure 2 we illustrate the average SuperGLEBer performance alongside the standard decoder models (blue), as well as the performance of decoders with a bidirectional attention mask (purple). Interestingly, for encoder-typical tasks such as sentence similarity (PAWSX), the performance of both the 120M, 1B and 7B LLäMmlein models increased even without explicit bidirectional continued pre-training, showing gains from bidirectional fine-tuning alone (120M: 0.477 $\rightarrow$ 0.516; 1B: 0.548 $\rightarrow$ 0.580; 7B: 0.524 $\rightarrow$ 0.660).

Observation: Switching the decoder attention mask to bidirectional already yields improvements on encoder-specific tasks without continued pre-training but bidirectional fine-tuning.

Regarding the increases in average performance, the scores remained largely consistent over time, without showing significant improvements. This suggests that training could have been stopped earlier, thus reducing the overall training time, emphasizing that converting a decoder is substantially faster than training a model from scratch.

5.2. Final Model Evaluation

Natural Language Understanding We evaluate all final models on the full SuperGLEBer benchmark. Table 2 reports average performances across a subset of our models. More detailed results can be found in Table 10. In particular, we compare our models to established encoders: GBERT (Chan et al., 2020), GeBERTa (Dada et al., 2023), XLM-RoBERTa (Conneau et al., 2020;

Model	Size	Avg.	Class.	NER	PAWSX	QA
$\text{GBERT}_{\text{Base}}$	111M	0.718	0.723	0.786	0.561	0.803
$\text{GBERT}_{\text{Large}}$	337M	0.768	0.785	0.799	0.654	0.832
$\text{GeBERTa}_{\text{Base}}$	139M	0.716	0.715	0.778	0.559	0.813
$\text{GeBERTa}_{\text{Large}}$	406M	0.749	0.743	0.791	0.619	0.844
$\text{GeBERTa}_{\text{XLarge}}$	887M	0.767	0.770	0.807	0.643	0.848
$\text{XLM-RoBERTa}_{\text{Base}}$	279M	0.689	0.693	0.754	0.505	0.802
$\text{XLM-RoBERTa}_{\text{Large}}$	561M	0.730	0.714	0.787	0.583	0.837
$\text{XLM-RoBERTa}_{\text{XLarge}}$	3.48B	0.758	0.750	0.802	0.656	0.822
mmBERT-small	142M	0.748	0.759	0.800	0.606	0.828
mmBERT-base	309M	0.758	0.757	0.804	0.622	0.849
EuroBERT-210m	212M	0.706	0.659	0.698	0.607	0.859
EuroBERT-610m	609M	0.721	0.633	0.750	0.633	0.869
EuroBERT-2b	2.11B	0.731	0.633	0.764	0.655	0.873
LLäMmlein	120M	0.676	0.702	0.712	0.477	0.812
LLäMmlein2Vec (*ext1*)	120M	0.684	0.703	0.741	0.472	0.819
LLäMmlein	1B	0.733	0.781	0.773	0.548	0.828
LLäMmlein2Vec (*ext1*)	1B	0.762	0.776	0.812	0.615	0.843
LLäMmlein	7B	0.747	0.810	0.805	0.524	0.851
LLäMmlein2Vec (*ext1*)	7B	0.787	0.799	0.838	0.670	0.842
ModernGBERT	134M	0.730	0.716	0.782	0.589	0.833
ModernGBERT (*ext1+2*)	134M	0.749	0.735	0.805	0.612	0.836
ModernGBERT	1B	0.800	0.806	0.839	0.681	0.874
ModernGBERT (*ext1+2*)	1B	**0.808**	**0.812**	**0.845**	**0.699**	**0.876**

Table 2: Performance comparison on SuperGLEBer benchmark. For ModernGBERT we provide results before and after the context extension.

Goyal et al., 2021), mmBERT (Marone et al., 2025) and EuroBERT (Boizard et al., 2025).

Our ModernGBERT consistently outperforms comparable and larger models. The 134M variant achieves an average score of 0.749, surpassing all similar-sized baselines, as well as LLäMmlein 1B (0.733), EuroBERT-2b (0.731) and LLäMmlein 7B (0.747). The ModernGBERT 1B variant achieves a new state-of-the-art average score across the entire SuperGLEBer of 0.808, outperforming $\text{GBERT}_{\text{Large}}$ (0.768) by 4% and beating the seven times larger LLäMmlein2Vec 7B (0.787). It leads in all four evaluation categories, i.e. classification (0.812), NER (0.845), sentence similarity (0.699) and QA (0.876). ModernGBERT scales well, with performance improving for larger model sizes, again suggesting that scaling ModernBERT-style encoders can leverage large monolingual corpora effectively. In the SuperGLEBer setting, adding context extension improved ModernGBERT's average by 1.9% for the 134M model (from 0.730 to 0.749) and by 0.8% for the 1B variant (from 0.800 to 0.808). However, large improvements were not expected, as SuperGLEBer tasks do not make use of long contexts.

Adaptation via LLM2Vec yields consistent gains across models. Our first LLM2Vec tuning (on *ext1*, Section 2.2) showed the most prominent positive effect, while the second fine-tune using the *ext2* datasets showed only marginal increase, or even a decrease in performance. The same holds for a mixture of the two LLM2Vec adapters (*ext1+2*). Therefore, we report only *ext1* results in this and all subsequent tables for simplicity. The LLäMmlein2Vec 7B achieves the strongest results among the LLM2Vec models (0.787). Conversion of LLäMmlein 120M, 1B, 7B improved the average score by 0.8%, 2.9%, and 4.0% respectively. This effect is especially pronounced in PAWSX, with scores increasing by up to 14.6% for LLäMmlein 7B and 6.7% for LLäMmlein 1B.

Observation: LLM2Vec yields the best improvement on similarity-related tasks.

Comparing the LLäMmlein2Vec with the ModernGBERT family, we find that on similarly sized models, ModernGBERT always outperforms the transformed decoders by a large margin. Only the much larger LLäMmlein2Vec 7B approaches the performance of ModernGBERT 1B. This systematic comparison using identical datasets provides the first comprehensive analysis of MLM vs. LLM2Vec for encoder development.

Observation: With similar data and model sizes, training encoders from scratch outperforms our converted models.

Model	Size	Avg.	Clstr	ReRnk	Retr
$\text{GBERT}_{\text{Base}}$	111M	0.360	0.274	0.118	0.226
$\text{GBERT}_{\text{Base}}{}^{f}$	111M	0.500	0.318	0.374	0.461
$\text{GBERT}_{\text{Large}}$	337M	0.412	0.336	0.206	0.297
$\text{GBERT}_{\text{Large}}{}^{f}$	337M	0.521	0.334	0.389	0.493
$\text{GeBERTa}_{\text{Base}}$	139M	0.382	0.312	0.174	0.213
$\text{GeBERTa}_{\text{Base}}{}^{f}$	139M	0.493	0.318	0.374	0.430
$\text{GeBERTa}_{\text{Large}}$	406M	0.397	0.287	0.223	0.274
$\text{GeBERTa}_{\text{Large}}{}^{f}$	406M	0.494	0.311	0.374	0.432
$\text{GeBERTa}_{\text{XLarge}}$	887M	0.325	0.278	0.108	0.058
$\text{GeBERTa}_{\text{XLarge}}{}^{f}$	887M	0.521	0.323	0.414	0.462
$\text{XLM-RoBERTa}_{\text{Base}}$	279M	0.248	0.173	0.024	0.008
$\text{XLM-RoBERTa}_{\text{Base}}{}^{f}$	279M	0.403	0.247	0.247	0.299
$\text{XLM-RoBERTa}_{\text{Large}}$	561M	0.264	0.172	0.048	0.026
$\text{XLM-RoBERTa}_{\text{Large}}{}^{f}$	561M	0.460	0.259	0.343	0.416
$\text{XLM-RoBERTa}_{\text{XLarge}}$	3.48B	0.301	0.225	0.090	0.142
$\text{XLM-RoBERTa}_{\text{XLarge}}{}^{f}$	3.48B	0.479	0.342	0.362	0.407
mmBERT-small	142M	0.289	0.163	0.091	0.075
mmBERT-small ^{f}	142M	0.430	0.167	0.390	0.359
mmBERT-base	309M	0.318	0.237	0.092	0.071
mmBERT-base ^{f}	309M	0.475	0.214	0.428	0.406
EuroBERT-210m	212M	0.293	0.240	0.070	0.118
EuroBERT-210m ^{f}	212M	0.431	0.189	0.404	0.371
EuroBERT-610m	609M	0.299	0.232	0.093	0.134
EuroBERT-610m ^{f}	609M	0.419	0.288	0.263	0.352
EuroBERT-2b	2.11B	0.230	0.186	0.060	0.065
EuroBERT-2b ^{f}	2.11B	0.452	0.191	0.460	0.420
LLäMmlein2Vec (*ext1*)	120M	0.315	0.261	0.139	0.224
LLäMmlein2Vec (*ext1*)^{f}	120M	0.471	0.308	0.325	0.425
LLäMmlein2Vec (*ext1*)	1B	0.399	0.308	0.183	0.276
LLäMmlein2Vec (*ext1*)^{f}	1B	0.540	**0.343**	0.433	0.511
LLäMmlein2Vec (*ext1*)	7B	0.376	0.249	0.169	0.266
LLäMmlein2Vec (*ext1*)^{f}	7B	**0.557**	0.339	**0.477**	**0.522**
ModernGBERT	134M	0.383	0.293	0.139	0.241
ModernGBERT ^{f}	134M	0.485	0.303	0.364	0.432
ModernGBERT (*ext1+2*)	134M	0.376	0.296	0.120	0.213
ModernGBERT (*ext1+2*) ^{f}	134M	0.501	0.312	0.404	0.446
ModernGBERT	1B	0.374	0.318	0.097	0.199
ModernGBERT ^{f}	1B	0.549	0.339	0.463	0.511
ModernGBERT (*ext1+2*)	1B	0.366	0.307	0.088	0.191
ModernGBERT (*ext1+2*) ^{f}	1B	0.551	0.338	0.459	0.512

Table 3: Performance comparison on MTEB. "Avg." refers to the average over all six task groups, not only the ones shown here. f marks the variants with additional training.

Text Embedding We evaluate models on the MTEB benchmark, which covers six task categories: classification, pair classification, clustering, reranking, retrieval, and short text similarity (STS) tasks. A summary of results is visualized in Table 3, while all results are provided in Table 13. In general, supervised fine-tuning on mMARCO yields consistent improvements across all model types. While classification performance sometimes declines, substantial gains can be observed in other areas: 25% on average for reranking, 26% for retrieval and 25% for STS.

Observation: Fine-tuning yields the largest gains in reranking, retrieval, and STS tasks, rather than for classification and clustering.

The best overall average performance is achieved by the fine-tuned LLäMmlein2Vec 7B (0.557), closely followed by the fine-tuned ModernGBERT 1B (0.551), despite the latter being significantly smaller. LLäMmlein2Vec models generally show strong performance after fine-tuning, particularly when trained with the extension dataset of the first phase (*ext1*). Using the second extension phase (*ext2*) or combining both adapters into the base model (*ext1+2*) again harms the performance. Interestingly, the latter shows the largest fine-tuning gains among the three variants. The ModernGBERT models perform competitively to similarly sized models. Before fine-tuning, ModernGBERT 1B (avg. 0.366) already outperforms most encoder-only models, such as $\text{GeBERTa}_{\text{XLarge}}$ (0.325), $\text{XLM-RoBERTa}_{\text{XLarge}}$ (0.301), and EuroBERT-2b (0.230) but not $\text{GBERT}_{\text{Large}}$ (0.412). However, after fine-tuning, it demonstrates clear superiority among native encoder-only models by at least 3% on the average score. As with our observations on the SuperGLEBer benchmark, ModernGBERT's context extension did not show significant improvements here. Comparing ModernGBERT and LLäMmlein2Vec, we find that before fine-tuning, the LLäMmlein2Vec 1B and 7B models produce better representations than ModernGBERT 1B. However, after fine-tuning, ModernGBERT 1B surpasses the 1B variant of LLäMmlein2Vec on average and closely aligns with the larger 7B model.

Long-Context Understanding Table 4 reports results on our new German QA-NIAH benchmark (see Table 14 for more evaluation). We evaluate subsets of short (<1,024), medium (1,024 to 4,095), and long (4,096 to 8,192) sequences, focusing on LLMs supporting up to 8,192 tokens: ModernGBERT, the encoder-converted LLäMmlein2Vec, as well as their original decoder counterparts. Notably, LLäMmlein models were pretrained with a maximum context of 2,048 tokens and only LLäMmlein2Vec was post-trained with a maximum context length of 8,192. ModernGBERT 1B demonstrates strong long-context performance across all lengths, outperforming all encoders. The first extension phase during training approximately tripled accuracy, while the final *High Quality* extension slightly reduced performance, especially for the 134M variant. Regarding LLM2Vec, a sufficiently long conversion improved long-context understanding. Conversion of LLäMmlein 120M and 1B decoders (with native context length of 2,048) improved accuracy by factor 1.3 resp. 2, both not as pronounced in comparison to the ModernGBERT encoders. For LLäMmlein2Vec 7B however (with LLM2Vec training on approximately half of our *ext1* dataset), it decreased by 51%, with no correct answers on haystacks of >4,096 tokens. Given the intensive compute requirements, we did not explore further optimizations regarding context extension of the LLäMmlein2Vec 7B model.

Model	Size	<1,024 tok.	1,024 to 4,095 tok.	4,096 to 8,192 tok.	Avg.
LLäMmlein	120M	0.286	0.124	0.049	0.091
LLäMmlein	1B	0.517	0.230	0.088	0.165
LLäMmlein	7B	0.529	0.310	0.122	0.216
LLäMmlein2Vec (*ext1*)	120M	0.315	0.206	0.044	0.120
LLäMmlein2Vec (*ext1*)	1B	0.588	0.448	0.232	0.333
LLäMmlein2Vec (*ext1*)	7B	0.597	0.207	0.000	0.111
ModernGBERT	134M	0.552	0.168	0.013	0.105
ModernGBERT (*ext1*)	134M	0.536	0.410	0.238	0.323
ModernGBERT (*ext1+2*)	134M	0.540	0.393	0.201	0.296
ModernGBERT	1B	0.556	0.233	0.023	0.136
ModernGBERT (*ext1*)	1B	**0.617**	0.506	**0.406**	**0.457**
ModernGBERT (*ext1+2*)	1B	0.601	**0.526**	0.383	0.451

Table 4: QA-NIAH results, with Exact Match metric. All tokens are counted per the model's respective tokenizer.

		Long	
Model	**Size**	Fixed Length	Variable Length
LLäMmlein2Vec	120M	6.69 ± 0.14	8.39 ± 0.35
LLäMmlein2Vec	1B	42.70 ± 0.12	59.70 ± 0.30
LLäMmlein2Vec	7B	180.00 ± 0.19	304.00 ± 0.41
ModernGBERT	134M	5.42 ± 0.33	4.71 ± 0.75
ModernGBERT	1B	28.70 ± 0.31	26.20 ± 0.36

Table 5: Model throughput in seconds per million tokens. All models run on RTX A6000 with Bfloat16 and Flash Attention 2. Reported uncertainty is the empirical standard deviation on 10 repetitions.

Observation: On small training datasets, LLM2Vec tuning limits the understanding of long-context samples.

5.3. Inference Efficiency

We evaluate inference efficiency across varying sequence lengths using four synthetic datasets, each containing 8,192 documents of random tokens. Following Warner et al. (2024), two datasets use fixed-length sequences (512 and 8,192 tokens), while the other two sample sequence lengths from normal distributions (either mean 256, variance 64; or mean 4,096, variance 1,024) to better simulate real-world conditions. Our ModernGBERT models adopt ModernBERT's unpadding approach: padding tokens are removed and sequences in a batch are concatenated, allowing Flash Attention to handle variable-length attention masks. The computational equivalence is facilitated by carefully crafting an appropriate attention mask. All other models rely on conventional padding. Among smaller models (134M

resp. 120M), ModernGBERT and LLäMmlein2Vec achieve comparable efficiency on fixed-length data, both only surpassed by $\text{GBERT}_{\text{Base}}$ and $\text{XLM-RoBERTa}_{\text{Base}}$ in terms of efficiency on short sequences. For 1B variants, ModernGBERT consistently outperforms LLäMmlein2Vec 1B and 7B variations in inference speed, likely due to its architectural decisions optimized for efficiency, such as ensuring that weight matrices have dimension of multiples of 64, and are divisible into 128×256 block for efficient tiling on the GPU. Gains are most pronounced for variable-length datasets, where ModernGBERT's unpadding yields clear benefits (see Table 5, and Table 15): the 134M ModernGBERT is the most efficient model on variable length, and the 1B variant substantially outpaces its LLäMmlein2Vec counterpart. Notably, ModernGBERT 1B matches the MTEB performance of LLäMmlein2Vec 7B (see Table 3) while being ten times faster on long-context documents.

Observation: When considering the trade-off between computational efficiency and downstream performance metrics, ModernGBERT consistently emerges as the optimal solution -frequently outperforming LLäMmlein2Vec on both dimensions.

6. Related Work

Next-Generation Encoders Several recent efforts have extended ModernBERT to other languages and domains, such as French (Antoun et al., 2024) and Japanese (Sugiura et al., 2025). Recently, mmBERT was created - a ModernBERT model trained on 3T tokens covering over 1,800 languages (Marone et al., 2025).

Concurrent to our work, several alternative encoder architectures have been proposed. Breton et al. (2025) introduced NeoBERT, a 250M parameter English encoder incorporating similar architectural innovations like ModernBERT, but scaling up layers rather than hidden dimension, switching from GeLU to SwiGLU activation, and using a modified training scheme (Cosine scheduler, reduced masking). Their model surpasses ModernBERT-large on GLUE and MTEB with 100M fewer parameters. Likewise, Boizard et al. (2025) presented EuroBERT (210M, 610M, 2.1B), a multilingual encoder family featuring architectural changes similar to those of ModernBERT, but retaining some architectural details (RMSNorm layer normalization, SiLU activation function, Llama-style tokenizer) from the Llama family, resembling our LLäMmlein2Vec architecture. Antoun et al. (2025) compared French ModernBERT and DeBERTaV3, finding DeBERTaV3 to be superior on downstream tasks but significantly slower in training and inference.

Tuning decoder-only LLMs into Encoders Due to the lack of new encoder models (Weller et al., 2025), works like LLM2Vec (Section 3.2) have proposed strategies for turning decoder-only models into encoders (Section 3.2). Few works have investigated converting decoder-only LLMs into encoders, besides LLM2Vec (Section 3.2). Recent studies predominantly address either distilling text embedders (Li and Li, 2024; Lee et al., 2025, 2024; Ma et al., 2025) or fine-tuning LLMs as bidirectional encoders for specific tasks (Li et al., 2023; Dukić and Snajder, 2024), with evaluation primarily focused on English or multilingual settings.

Khosla et al. (2025) introduced MAGNET, a method for converting decoder-only models into foundational encoders, similarly to LLM2Vec. Unlike LLM2Vec, which enables bidirectional attention and masked next-token prediction, MAGNET uses a hybrid of bidirectional and causal attention and adds a missing-span generation objective, aiming for a more general pre-training signal. Concurrent to our work, the Ettin Suite (Weller et al., 2025) provides a systematic comparison of encoder-only and decoder-only architectures trained on identical data and recipes. Notably, they also experiment with objective switching, continuing training on 50B tokens, and directly comparing downstream task performance. Rather than modifying the base model, Goto et al. (2025) instead propose to supplement decoder-only LLMs with a smaller, separately trained backward language model to provide bidirectional context. However, evaluation in these works remains primarily focused on English or multilingual settings.

7. Conclusion

In this work we introduce two encoder-only families from scratch ModernGBERT and LLäMmlein2Vec, contributing: a) the first systematic MLM vs. LLM2Vec comparison for encoders using identical datasets, b) a novel QA-NIAH evaluation adapted for encoder models, and c) the first combination of LLM2Vec with context extension training.

Our main findings show that the proposed ModernGBERT family, especially the 1B variant, sets a new state-of-the-art for German encoders, outperforming previous models while remaining suitable for practical deployment as a drop-in replacement for GBERT, capable of handling sequences of up to 8,192 tokens. Our learning dynamics analysis confirms that larger encoder architectures can effectively exploit terabyte-scale German monolingual corpora, with performance consistently improving with increased model size

and data. A comparison of ModernGBERT and LLäMmlein2Vec (derived from LLäMmlein) shows that dedicated encoder training yields superior results, justifying its computational expense when parameter efficiency is essential. At the same time, LLM2Vec provides a resource-efficient alternative for scenarios where training a full encoder from scratch is too expensive. Notably, in some use cases, we find that fine-tuning a decoder with a bidirectional attention mask already delivers substantial gains. These trends suggest that even larger encoder models could yield further gains, which we leave to future work. By releasing ModernGBERT and LLäMmlein2Vec, along with full training transparency, intermediate checkpoints, detailed documentation, and accompanying resources, we aim to facilitate further development and understanding within the German LLM community.

Acknowledgements

The authors gratefully acknowledge the scientific support and HPC resources provided by the Erlangen National High Performance Computing Center (NHR@FAU) of the Friedrich Alexander-Universität Erlangen-Nürnberg (FAU) under the NHR project b233cb. NHR funding is provided by federal and Bavarian state authorities. NHR@FAU hardware is partially funded by the German Research Foundation (DFG) – 440719683. Furthermore, we gratefully acknowledge the HPC resources provided by the JuliaV2 cluster at the Universität Würzburg (JMU), which was funded as DFG project as "Forschungsgroßgerät nach Art 91b GG" under INST 93/1145-1 FUGG. The project staff is partially funded by the DFG – 529659926. The data science chair is part of the CAIDAS, the Center for Artificial Intelligence and Data Science, and is supported by the Bavarian High-Tech Agenda, which made this research possible. This publication was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under the project LitBERT, project no. 529659926. We would also like to thank Anton Vlasjuk for his valuable contributions and insights, particularly for his efforts in integrating LLM2Vec.

Limitations

Despite the ModernGBERT and LLäMmlein2Vec models being a notable advancement in the German NLP landscape, several limitations persist: **1) Monolingual focus.** Although the focus on German is a strength for this specific context, ModernGBERT is unable to utilize multilingual contexts or perform cross-lingual tasks, hindering applicability in some scenarios. **2) Limited coding capabilities.** High-quality German resources for coding are rare, and no code is included in the training dataset. This restricts its capabilities in code retrieval applications. **3) Evaluation scope.** While we rigorously evaluated our models on the German SuperGLEBer and MTEB benchmarks, these benchmarks are limited in their domain, and other domains such as literature, medical domains, or technical subjects were not tested. Furthermore, our benchmarks do not strictly probe for "German factual knowledge", for instance, knowledge about German geography, or common German TV shows. To our knowledge, no established German benchmark for knowledge-heavy tasks currently exists, leaving this an open gap for future work. **4) No custom tokenizer.** We utilized the original BERT-style GBERT tokenizer due to its availability and persistent usage. However, we did not invest in developing a custom tokenizer, like the BPE-style OLMo tokenizer used in ModernBERT. Consequently, ModernGBERT's tokenizer cannot, e.g., differentiate between various whitespace characters or encode emoji. Investigating whether a modern BPE-style tokenizer would benefit the model, remains for future work. **5) Evaluation of long-context understanding.** Due to the absence of high-quality native German evaluation datasets, we had to rely on non-natural QA–NIAH sequences, only broadly testing for long-context understanding. In contrast to English benchmarks such as ∞Bench-MC (Zhang et al., 2024) or LongBench-v2 (Bai et al., 2025), which include full novels along with questions that require attention to much information scattered throughout the novel, our synthetic haystack approach may overestimate real-world long-context capabilities. In future work, we plan on developing a dedicated high-quality non-synthetic German long-context evaluation benchmark.

8. Bibliographical References

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.

Wissam Antoun, Benoît Sagot, and Djamé Seddah. 2025. ModernBERT or DeBERTaV3? examining architecture and data influence on transformer encoder models performance. ArXiv:2504.08716.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025. LongBench v2: Towards deeper under-

standing and reasoning on realistic long-context multitasks. ArXiv:2412.15204.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. LLM2Vec: Large language models are secretly powerful text encoders. In *First Conference on Language Modeling*, Philadelphia, PA, USA.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 2397–2430. PMLR.

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. 2025. EuroBERT: Scaling multilingual encoders for european languages. In *Second Conference on Language Modeling*.

Hamed Bonab, Mohammad Aliannejadi, Ali Vardasbi, Evangelos Kanoulas, and James Allan. 2021. Cross-market product recommendation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, page 110–119, New York, NY, USA. Association for Computing Machinery.

Luiz Bonifacio, Vitor Jeronymo, Hugo Queiroz Abonizio, Israel Campiotti, Marzieh Fadaee, Roberto Lotufo, and Rodrigo Nogueira. 2022. mMARCO: A multilingual version of the MS MARCO passage ranking dataset. ArXiv:2108.13897.

Loïck Bourdois. 2025. Model statistics of the 50 most downloaded entities on hugging face.

Lola Le Breton, Quentin Fournier, Mariam El Mezouar, and Sarath Chandar. 2025. NeoBERT: A next-generation BERT. ArXiv:2502.19587.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Xi Chen, Ali Zeynali, Chico Camargo, Fabian Flöck, Devin Gaffney, Przemyslaw Grabowicz, Scott A. Hale, David Jurgens, and Mattia Samory. 2022. SemEval-2022 task 8: Multilingual news article similarity. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1094–1106, Seattle, United States. Association for Computational Linguistics.

Aaron Chibb. 2022. German-english false friends in multilingual transformer models: An evaluation on robustness and word-to-word fine-tuning. `huggingface:aari1995/false_friends_en_de`.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Amin Dada, Aokun Chen, Cheng Peng, Kaleb Smith, Ahmad Idrissi-Yaghir, Constantin Seibold, Jianning Li, Lars Heiliger, Christoph Friedrich, Daniel Truhn, Jan Egger, Jiang Bian, Jens Kleesiek, and Yonghui Wu. 2023. On the impact of cross-domain data on German language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13801–13813, Singapore. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized llms. ArXiv:2305.14314.

David Dukić and Jan Snajder. 2024. Looking right is sometimes right: Investigating the capabilities

of decoder-only LLMs for sequence labeling. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14168–14181, Bangkok, Thailand. Association for Computational Linguistics.

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, Ömer Çağatan, Akash Kundu, Martin Bernstorff, Shitao Xiao, Akshita Sukhlecha, Bhavish Pahwa, Rafał Poświata, Kranthi Kiran GV, Shawon Ashraf, Daniel Auras, Björn Plüster, Jan Philipp Harries, Loïc Magne, Isabelle Mohr, Mariya Hendriksen, Dawei Zhu, Hippolyte Gisserot-Boukhlef, Tom Aarsen, Jan Kostkan, Konrad Wojtasik, Taemin Lee, Marek Šuppa, Crystina Zhang, Roberta Rocca, Mohammed Hamdy, Andrianos Michail, John Yang, Manuel Faysse, Aleksei Vatolin, Nandan Thakur, Manan Dey, Dipam Vasani, Pranjal Chitale, Simone Tedeschi, Nguyen Tai, Artem Snegirev, Michael Günther, Mengzhou Xia, Weijia Shi, Xing Han Lù, Jordan Clive, Gayatri Krishnakumar, Anna Maksimova, Silvan Wehrli, Maria Tikhonova, Henil Panchal, Aleksandr Abramov, Malte Ostendorff, Zheng Liu, Simon Clematide, Lester James Miranda, Alena Fenogenova, Guangyu Song, Ruqiya Bin Safi, Wen-Ding Li, Alessia Borghini, Federico Cassano, Hongjin Su, Jimmy Lin, Howard Yen, Lasse Hansen, Sara Hooker, Chenghao Xiao, Vaibhav Adlakha, Orion Weller, Siva Reddy, and Niklas Muennighoff. 2025. MMTEB: Massive multilingual text embedding benchmark. ArXiv:2502.13595.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a german named entity recognizer with semantic generalization. In *Conference on Natural Language Processing*.

Jack FitzGerald, Christopher Hench, Charith Peris, Scott Mackie, Kay Rottmann, Ana Sanchez, Aaron Nash, Liam Urbach, Vishesh Kakarala, Richa Singh, Swetha Ranganath, Laurie Crist, Misha Britan, Wouter Leeuwis, Gokhan Tur, and Prem Natarajan. 2023. MASSIVE: A 1M-example multilingual natural language understanding dataset with 51 typologically-diverse languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4277–4302, Toronto, Canada. Association for Computational Linguistics.

Tianyu Gao, Alexander Wettig, Howard Yen, and Danqi Chen. 2025. How to train long-context language models (effectively). ArXiv:2410.02660.

Takumi Goto, Hiroyoshi Nagao, and Yuta Koreeda. 2025. Acquiring bidirectionality via large and small language models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1711–1717, Abu Dhabi, UAE. Association for Computational Linguistics.

Naman Goyal, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. 2021. Larger-scale transformers for multilingual masked language modeling. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 29–33, Online. Association for Computational Linguistics.

Verena Henrich, Erhard Hinrichs, and Tatiana Vodolazova. 2012. WebCAGe – a web-harvested corpus annotated with GermaNet senses. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 387–396, Avignon, France. Association for Computational Linguistics.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. RULER: What's the real context size of your long-context language models? ArXiv:2404.06654.

Maor Ivgi, Uri Shaham, and Jonathan Berant. 2023. Efficient long-text understanding with short-text models. *Transactions of the Association for Computational Linguistics*, 11:284–299.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. The multilingual amazon reviews corpus. ArXiv:2010.02573.

Savya Khosla, Aditi Tiwari, Kushal Kafle, Simon Jenni, Handong Zhao, John Collomosse, and Jing Shi. 2025. MAGNET: Augmenting generative decoders with representation learning and infilling capabilities. ArXiv:2501.08648.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. NV-embed: Improved techniques for training llms as generalist embedding models. ArXiv:2405.17428.

Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R. Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, Yi Luan, Sai Meher Karthik Duddu, Gustavo Hernandez Abrego, Weiqiang Shi, Nithi Gupta, Aditya Kusupati, Prateek Jain, Siddhartha Reddy Jonnalagadda, Ming-Wei Chang, and Iftekhar Naim.

2024. Gecko: Versatile text embeddings distilled from large language models. ArXiv:2403.20327.

Haoran Li, Abhinav Arora, Shuohui Chen, Anchit Gupta, Sonal Gupta, and Yashar Mehdad. 2021. MTOP: A comprehensive multilingual task-oriented semantic parsing benchmark. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2950–2962, Online. Association for Computational Linguistics.

Xianming Li and Jing Li. 2024. BeLLM: Backward dependency enhanced large language model for sentence embeddings. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 792–804, Mexico City, Mexico. Association for Computational Linguistics.

Zongxi Li, Xianming Li, Yuzhang Liu, Haoran Xie, Jing Li, Fu lee Wang, Qing Li, and Xiaoqin Zhong. 2023. Label supervised LLaMA finetuning. ArXiv:2310.01208.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.

Xueguang Ma, Xi Victoria Lin, Barlas Oguz, Jimmy Lin, Wen tau Yih, and Xilun Chen. 2025. DRAMA: Diverse augmentation from large language models to smaller dense retrievers. ArXiv:2502.18460.

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

Philip May. 2021. Machine translated multilingual sts benchmark dataset. `huggingface:PhilipMay/stsb_multi_mt`.

Timo Möller, Julian Risch, and Malte Pietsch. 2021. GermanQuAD and GermanDPR: Improving non-English question answering and passage retrieval. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 42–50, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. MTEB: Massive text embedding benchmark. ArXiv:2210.07316.

James O'Neill, Polina Rozenshtein, Ryuichi Kiryo, Motoko Kubota, and Danushka Bollegala. 2021. I wish i would have loved this one, but i didn't – a multilingual dataset for counterfactual detection in product reviews. ArXiv:2104.06893.

Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Martin Jaggi, Leandro von Werra, and Thomas Wolf. 2024. Fineweb2: A sparkling update with 1000s of languages. Huggingface:HuggingFaceFW/fineweb-2.

Jan Pfister and Andreas Hotho. 2024. SuperGLEBer: German language understanding evaluation benchmark. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7904–7923, Mexico City, Mexico. Association for Computational Linguistics.

Jan Pfister, Julia Wunderle, and Andreas Hotho. 2024. LLäMmlein: Compact and competitive german-only language models from scratch. ArXiv:2411.11171.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Julian Risch, Anke Stoll, Lena Wilms, and Michael Wiegand. 2021. Overview of the GermEval 2021 shared task on the identification of toxic, engaging, and fact-claiming comments. In *Proceedings of the GermEval 2021 Shared Task on the Identification of Toxic, Engaging, and Fact-Claiming Comments*, pages 1–12, Duesseldorf, Germany. Association for Computational Linguistics.

Issa Sugiura, Kouta Nakayama, and Yusuke Oda. 2025. llm-jp-modernbert: A ModernBERT model trained on a large-scale japanese corpus with long context length. ArXiv:2504.15544.

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

Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. 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. In *Advances in Neural Information Processing Systems*, volume 37, pages 116462–116492. Curran Associates, Inc.

Silvan Wehrli, Bert Arnrich, and Christopher Irrgang. 2023. German text embedding clustering benchmark. In *Proceedings of the 19th Conference on Natural Language Processing (KONVENS 2023)*, pages 187–201, Ingolstadt, Germany. Association for Computational Lingustics.

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.

Michael Wojatzki, Eugen Ruppert, Sarah Holschneider, Torsten Zesch, and Chris Biemann. 2017. Germeval 2017: Shared task on aspect-based sentiment in social media customer feedback. In *Proceedings of the GermEval 2017 – Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*, pages 1–12, Berlin, Germany.

Marco Wrzalik and Dirk Krechel. 2021. GerDaLIR: A German dataset for legal information retrieval. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 123–128, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A cross-lingual adversarial dataset for paraphrase identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3687–3692, Hong Kong, China. Association for Computational Linguistics.

Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Hao, Xu Han, Zhen Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. 2024. ∞Bench: Extending long context evaluation beyond 100K tokens. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15262–15277, Bangkok, Thailand. Association for Computational Linguistics.

Xinyu Zhang, Nandan Thakur, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Mehdi Rezagholizadeh, and Jimmy Lin. 2023. MIRACL: A multilingual retrieval dataset covering 18 diverse languages. *Transactions of the Association for Computational Linguistics*, 11:1114–1131.

Appendix

9. Model Architecture and Training Details

Table 6 provides an overview of the model architectures for the ModernGBERT (134M, 1B) and LLäMmlein2Vec (120M, 1B, 7B) model families. Detailed training settings regarding the pre-training phase, context extension phase one and two, for ModernGBERT are listed in Table 7, and those for LLäMmlein2Vec, covering MNTP training, can be found in Table 8.

10. Evaluation Results

In the following we present the full evaluation results on SuperGLEBer, MTEB, NIAH, and our efficiency benchmarks for German-capable encoder models, specifically our ModernGBERT (134M, 1B) and LLäMmlein2Vec (120M, 1B, 7B).

10.1. SuperGLEBer

In Figure 3 we illustrate the training progress, evaluating several intermediate checkpoints of ModernGBERT 134M and 1B. Notably, while the smaller model did not show significant improvements after approximately 15% of it's training data, the 1B model improves performance until 67%.

Table 10 presents the results on the full SuperGLEBer benchmark, comparing various German-capable encoder models. Notably, ModernGBERT 1B sets a new state of the art, surpassing the previously leading encoder model $\text{GBERT}_{\text{Large}}$ as well as the seven times larger LLäMmlein2Vec 7B. Transforming the LLäMmlein decoders into encoders (in particular, the +ext1 variant) improves the average score and yields notable gains on similarity and sequence tagging tasks.

10.2. Massive Text Embedding Benchmark (MTEB)

We summarize all tasks included in the *MTEB(deu„v1)* benchmark in Table 11 and report the corresponding results in Table 13. In addition to the base model outcomes, we also present scores for models after supervised training on the mMARCO dataset. Notably, the fine-tuned versions consistently outperform their base counterparts, with particularly strong improvements in reranking, retrieval, and s2s tasks. LLäMmlein2Vec 7B achieves the best results closely followed by ModernGBERT 1B.

10.3. Needle-in-a-Haystack

The results on the Question-Answering Needle-in-a-Haystack test are presented in Table 14. ModernGBERT 1B performs strongly across sequence lengths, surpassed only by the eight-times larger LLaMA 3.2. For LLäMmlein 120M and 1B, MNTP training on the*LONG-Head* or *LONG-Head/Middle* datasets improves performance on longer contexts.

10.4. Efficiency

Finally, we depict the outcomes of our model efficiency tests in Table 15. The smaller ModernGBERT model is the most efficient on variable-length input, while the 1B variant substantially outperforms its LLäMmlein2Vec counterpart.

	ModernGBERT		LLäMmlein2Vec		
Parameters	134M	1B	120M	1B	7B
Vocabulary	31,168	31,168	32,064	32,064	32,064
Unused Tokens	66	66	54	54	54
Layers	22	28	12	22	32
Hidden Size	768	2,048	768	2,048	4,096
Transformer Block	Pre-Norm	Pre-Norm	Post-Norm	Post-Norm	Post-Norm
Activation Function	GeLU	GeLU	SiLU	SiLU	SiLU
Attention Heads	12	32	12	32	32
Head Size	64	64	64	64	128
Intermediate Size	1,152	3,072	2,048	5,632	11,008
Normalization	LayerNorm	LayerNorm	RMSNorm	RMSNorm	RMSNorm
Norm Epsilon	1×10^{-5}	1×10^{-5}	1×10^{-5}	1×10^{-5}	1×10^{-5}
RoPE theta	160,000	160,000	160,000	160,000	160,000
Global Attention	Every three layers	Every three layers	Every layer	Every layer	Every layer
Local Attention Window	128	128	—	—	—
Local Attn RoPE theta	10,000	10,000	—	—	—

Table 6: Model design of the ModernGBERT and LLäMmlein2Vec model family.

	Pretraining Phase		Context Extension: Phase One		Context Extension: Phase Two	
	134M	1B	134M	1B	134M	1B
Training Tokens	0.47T	1.27T	52B	90B	14.4B	
Max Sequence Length	1,024		8,192		8,192	
RoPE Theta	10,000		160,000		160,000	
Batch Size	4,608	4,928	96	96	96	96
Warmup (tokens)	3×10^9		—	—	—	—
Microbatch Size	96	28	8	3	8	3
Learning Rate	8×10^{-4}	5×10^{-5}	3×10^{-4}	5×10^{-5}	3×10^{-4}	5×10^{-6}
Schedule	Trapezoidal		—	—	1-sqrt	
Warmup (tokens)	15×10^9		—	—	—	—
Decay (tokens)	—	—	—	—	12.8×10^9	
Training Time (hours)	31.3	446.1	5.9	42.1	2.0	8.3
Optimizer	StableAdamW					
Betas	(0.90, 0.98)					
Epsilon	1×10^{-6}					
Training Hardware	16× H100					
Training Strategy	Distributed DataParallel, bfloat16					

Table 7: ModernGBERT training settings. Dropout and below are shared across all phases.

	LLäMmlein2Vec 120M		LLäMmlein2Vec 1B		LLäMmlein2Vec 7B	
	Ext1	Ext2	Ext1	Ext2	Ext1	Ext2
Training Tokens	52B	14.4B	90B	14.4B	90B	14.4B
Max Sequence Length	8,192		8,192		8,192	
RoPE theta	160,000		160,000		160,000	
Batch Size	32	32	32	32	16	16
Training Hardware	64× H200		64× H200		256 × H200	128 × H200
Training Duration	10h41	3h40	37h24	6h40	14h25†	9h39

Table 8: LLäMmlein2Vec training settings. Due to limited resources we had to terminate the 7B model training on the first extension dataset early †.

type	model	params	classification						tagging			similarity	QA	Average
			tox. macro F1	sent. micro F1	match ACC	WSD micro F1	other mixed	avg mixed	NER micro F1	other micro F1	avg micro F1	pearson corr	m. t. F1	mixed
enc	$GBERT_{Base}$	111M	0.537	0.620	0.738	0.814	0.749	0.723	0.705	0.806	0.786	0.561	0.803	0.718
enc	$GBERT_{Large}$	337M	0.604	0.673	0.810	0.837	0.816	0.785	0.744	0.813	0.799	0.654	0.832	0.768
enc	gerturax-3	135M	0.561	0.617	0.788	0.804	0.762	0.736	0.707	0.810	0.789	0.603	0.832	0.740
enc	$GeBERTa_{Base}$ †	139M	0.530	0.619	0.766	0.789	0.737	0.715	0.687	0.801	0.778	0.559	0.813	0.716
enc	$GeBERTa_{Large}$ †	406M	0.551	0.623	0.783	0.795	0.775	0.743	0.736	0.804	0.791	0.619	0.844	0.749
enc	$GeBERTa_{XLarge}$ †	887M	0.606	0.671	0.796	0.830	0.795	0.770	0.763	0.818	0.807	0.643	0.848	0.767
enc	$XLM\text{-}RoBERTa_{Base}$	279M	0.530	0.546	0.741	0.790	0.717	0.693	0.648	0.780	0.754	0.505	0.802	0.689
enc	$XLM\text{-}RoBERTa_{Large}$	561M	0.512	0.559	0.795	0.814	0.739	0.714	0.720	0.804	0.787	0.583	0.837	0.730
enc	$XLM\text{-}RoBERTa_{XLarge}$	3.48B	0.558	0.612	0.815	0.820	0.778	0.750	0.746	0.816	0.802	0.656	0.822	0.758
dec	Llama 3.2	1B	0.519	0.629	0.768	0.808	0.768	0.737	0.648	0.733	0.716	0.551	0.835	0.710
dec	Llama 3.1	8B	0.586	0.713	0.819	0.849	0.817	0.790	0.708	0.753	0.744	0.573	0.868	0.744
enc	mmBERT-small	142M	0.546	0.641	0.777	0.843	0.791	0.759	0.766	0.809	0.800	0.606	0.828	0.748
enc	mmBERT-base	309M	0.543	0.667	0.793	0.851	0.782	0.757	0.786	0.809	0.804	0.622	0.849	0.758
enc	EuroBERT-210m	212M	0.418	0.491	0.738	0.766	0.691	0.659	0.550	0.734	0.698	0.607	0.859	0.706
enc	EuroBERT-610m	609M	0.434	0.479	0.753	0.777	0.646	0.633	0.677	0.768	0.750	0.633	0.869	0.721
enc	EuroBERT-2b	2.11B	0.444	0.456	0.718	0.772	0.652	0.633	0.681	0.785	0.764	0.655	0.873	0.731

Table 9: SuperGLEBer results, averaged at varying levels of granularity, following (Pfister and Hotho, 2024). The columns reading “Average” have been averaged across the averages of the respective task types, in order to not overweight any task type for which more datasets exist. † indicates that these models were evaluated without fp16. ModernGBERT models marked with “w/o ext.” refer to the model checkpoint after pre-training but before the context extension phases. We also exemplary evaluated LLM2Vec after 20% of LONG Head or Head/Middle training data (“+ 1/5 ext1“).

type	model	params	classification						tagging					
			tox. macro F1	sent. micro F1	match ACC	WSD micro F1	other mixed	avg mixed	NER micro F1	other micro F1	avg micro F1	similarity pearson corr	QA m. t. F1	Average mixed
dec	LLäMmlein	120M	0.510	0.580	0.710	0.798	0.732	0.702	0.613	0.737	0.712	0.477	0.812	0.676
enc	LLäMmlein2Vec (1/5 ext1)	120M	0.480	0.571	0.696	0.793	0.738	0.700	0.627	0.754	0.729	0.471	0.821	0.680
enc	LLäMmlein2Vec (ext1)	120M	0.491	0.564	0.711	0.797	0.739	0.703	0.660	0.761	0.741	0.472	0.819	0.684
enc	LLäMmlein2Vec (ext2)	120M	0.471	0.568	0.686	0.772	0.739	0.697	0.620	0.757	0.730	0.455	0.818	0.675
enc	LLäMmlein2Vec (ext1+2)	120M	0.448	0.490	0.581	0.749	0.708	0.657	0.505	0.728	0.683	0.439	0.730	0.627
dec	LLäMmlein	1B	0.603	0.710	0.790	0.839	0.806	0.781	0.728	0.785	0.773	0.548	0.828	0.733
enc	LLäMmlein2Vec (1/5 ext1)	1B	0.587	0.694	0.827	0.820	0.806	0.779	0.795	0.815	0.811	0.615	0.842	0.762
enc	LLäMmlein2Vec (ext1)	1B	0.575	0.702	0.775	0.835	0.808	0.776	0.790	0.818	0.812	0.615	0.843	0.762
enc	LLäMmlein2Vec (ext2)	1B	0.528	0.688	0.754	0.815	0.789	0.755	0.775	0.812	0.804	0.580	0.836	0.744
enc	LLäMmlein2Vec (ext1+2)	1B	0.503	0.566	0.692	0.787	0.732	0.697	0.727	0.799	0.784	0.557	0.828	0.717
dec	LLäMmlein	7B	0.632	0.739	0.821	0.873	0.835	0.810	0.796	0.808	0.805	0.524	0.851	0.747
enc	LLäMmlein2Vec (ext1)	7B	0.633	0.742	0.813	0.839	0.822	0.799	0.851	0.835	0.838	**0.670**	0.842	0.787
enc	LLäMmlein2Vec (ext2)	7B	0.612	0.743	0.808	0.848	0.822	0.797	0.848	0.834	0.837	0.657	0.842	0.783
enc	LLäMmlein2Vec (ext1+2)	7B	0.554	0.647	0.610	0.786	0.744	0.709	0.747	0.804	0.793	0.639	0.821	0.740
enc	ModernGBERT	134M	0.503	0.617	0.769	0.780	0.744	0.716	0.734	0.794	0.782	0.589	0.833	0.730
enc	ModernGBERT + ext1+2	134M	0.526	0.647	0.779	0.806	0.760	0.735	0.791	0.809	0.805	0.621	0.836	0.749
enc	ModernGBERT	1B	0.610	**0.746**	**0.827**	0.858	0.833	0.806	0.849	0.837	0.839	0.681	0.874	0.800
enc	ModernGBERT + ext1+2	1B	**0.635**	0.745	0.826	**0.876**	**0.836**	**0.812**	**0.868**	**0.840**	**0.845**	0.699	**0.876**	**0.808**

Table 10: SuperGLEBer results, averaged at varying levels of granularity, following (Pfister and Hotho, 2024). The columns reading "Average" have been averaged across the averages of the respective task types, in order to not overweight any task type for which more datasets exist. † indicates that these models were evaluated without fp16. ModernGBERT models marked with "w/o ext." refer to the model checkpoint after pre-training but before the context extension phases. We also exemplary evaluated LLM2Vec after 20% of LONG Head or Head/Middle training data ("+ 1/5 ext1").

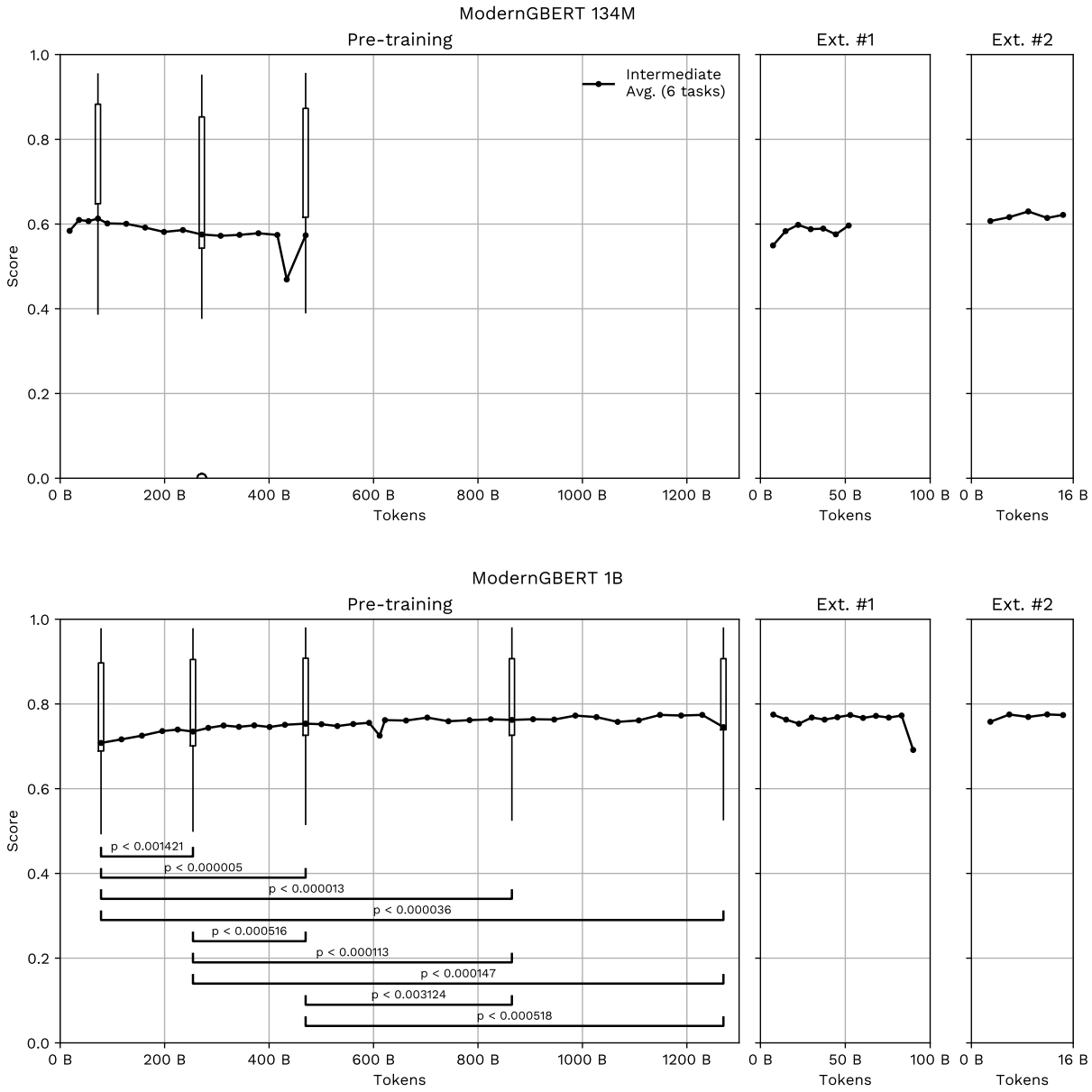

Figure 3: Intermediate Checkpoint Evaluation. Note that the solid line represents the mean of the six tasks selected for the intermediate checkpoint evaluation (NLI, FactClaiming Comments, DB Aspect, WebCAGe, EuroParl, PAWSX Similarity). The box plots show the distribution of scores for those checkpoints evaluated across all 29 SuperGLEBer tasks. We compared each pair of these checkpoints using Wilcoxon signed-rank tests, and highlighted significant increases with brackets. Brackets of pairs without significant increases are not displayed. (Accordingly, all pairs of 134M checkpoints show no significant increase.) Four checkpoints failed to converge during fine-tuning on some task, leading to the visible outliers. Similar behavior has been observed by Antoun et al. (2025).

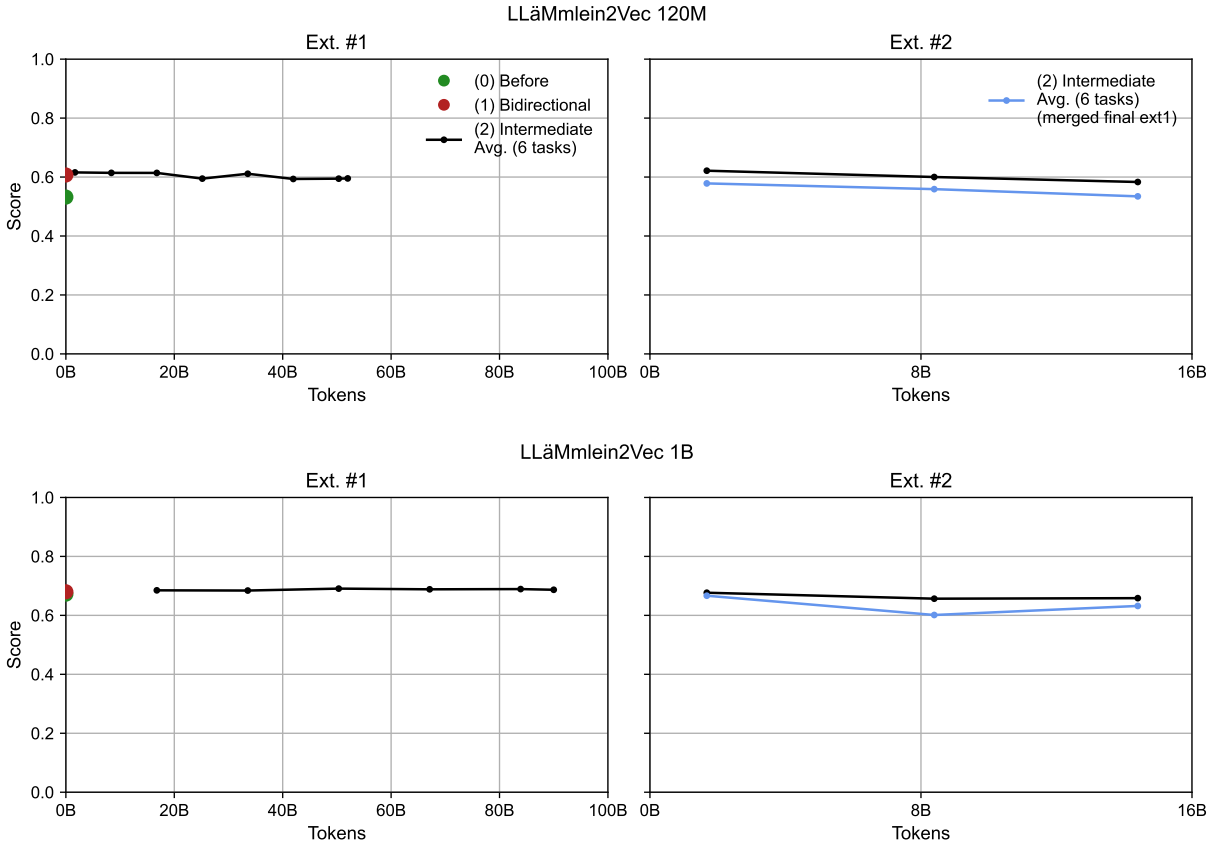

Figure 4: Intermediate checkpoint evaluation on six tasks SuperGLEBer tasks (Nli, FactClaiming Comments, DB Aspect, WEbCAGe EuroParl, PAWSX Similarity). We illustrate the average training progress for each dataset individually (left: ext1, right: ext2) as well as for a merged configuration (ext1 & ext2), shown in blue. In the merged setting, the final checkpoint of ext1 is combined with each intermediate checkpoint of ext2. Additionally, we highlight the standard decoder average in green and the decoder average with the switched bidirectional mask in red.

Category	Task	Metric	Reference
Classification	AmazonCounterfactual	Accuracy	O'Neill et al. (2021)
	AmazonReviews	Accuracy	Keung et al. (2020)
	MTOPDomain	Accuracy	Li et al. (2021)
	MTOPIntent	Accuracy	Li et al. (2021)
	MassiveIntent	Accuracy	FitzGerald et al. (2023)
	MassiveScenario	Accuracy	FitzGerald et al. (2023)
PairClassification	FalseFriendsGermanEnglish	Average Precision	Chibb (2022)
	PawsXPairClassification	Average Precision	Yang et al. (2019)
Clustering	BlurbsClusteringP2P	V-measure	Wehrli et al. (2023)
	BlurbsClusteringS2S	V-measure	Wehrli et al. (2023)
	TenKGnadClusteringP2P	V-measure	Wehrli et al. (2023)
	TenKGnadClusteringS2S	V-measure	Wehrli et al. (2023)
Reranking	MIRACLReranking	nDCG@10	Zhang et al. (2023)
Retrieval	GermanQuAD-Retrieval	MRR@5	Möller et al. (2021)
	GermanDPR	DCG@10	Möller et al. (2021)
	Xmarket	nDCG@10	Bonab et al. (2021)
	GerDaLIR	nDCG@10	Wrzalik and Krechel (2021)
STS	GermanSTSBenchmark	Spearman	May (2021); Cer et al. (2017)
	STS22	Spearman	Chen et al. (2022)

Table 11: Overview of tasks included in the German *MTEB(deu, v1)* benchmark, grouped by six categories: classification, pairclassification, clustering, reranking, retrieval and STS.

Model	Params	Classification Average	PairClassification Average	Clustering Average	Reranking Average	Retrieval Average	STS Average	Average
$GBERT_{Base}$	111M	0.634	0.504	0.274	0.118	0.226	0.402	0.360
$GBERT_{Base}$ †		0.632	0.601	0.318	0.374	0.461	0.613	0.500
$GBERT_{Large}$	337M	0.649	0.544	0.336	0.206	0.297	0.438	0.412
$GBERT_{Large}$ †		0.646	**0.662**	0.334	0.389	0.493	0.603	0.521
gerturax-3	135M	0.623	0.554	0.269	0.141	0.187	0.375	0.358
gerturax-3†		0.620	0.614	0.328	0.346	0.389	0.538	0.472
$GeBERTa_{Base}$	139M	0.632	0.535	0.312	0.174	0.213	0.429	0.382
$GeBERTa_{Base}$ †		0.611	0.613	0.318	0.374	0.430	0.611	0.493
$GeBERTa_{Large}$	406M	0.642	0.533	0.287	0.223	0.274	0.424	0.397
$GeBERTa_{Large}$ †		0.618	0.611	0.311	0.374	0.432	0.616	0.494
$GeBERTa_{XLarge}$	887M	0.626	0.536	0.278	0.108	0.058	0.342	0.325
$GeBERTa_{XLarge}$ †		0.638	0.631	0.323	0.414	0.462	0.655	0.521
$XLM\text{-}RoBERTa_{Base}$	279M	0.442	0.506	0.173	0.024	0.008	0.333	0.248
$XLM\text{-}RoBERTa_{Base}$ †		0.555	0.529	0.247	0.247	0.299	0.539	0.403
$XLM\text{-}RoBERTa_{Large}$	561M	0.510	0.510	0.172	0.048	0.026	0.320	0.264
$XLM\text{-}RoBERTa_{Large}$ †		0.576	0.574	0.259	0.343	0.416	0.593	0.460
$XLM\text{-}RoBERTa_{XLarge}$	3.48B	0.456	0.519	0.225	0.090	0.142	0.372	0.301
$XLM\text{-}RoBERTa_{XLarge}$ †		0.609	0.564	0.342	0.362	0.407	0.590	0.479
mmBERT-small	142M	0.562	0.506	0.163	0.091	0.075	0.337	0.289
mmBERT-small †	142M	0.538	0.559	0.167	0.390	0.359	0.568	0.430
mmBERT-base	309M	0.612	0.519	0.237	0.092	0.071	0.378	0.318
mmBERT-base †	309M	0.602	0.585	0.214	0.428	0.406	0.615	0.475
EuroBERT-210m	212M	0.436	0.509	0.240	0.070	0.118	0.384	0.293
EuroBERT-210m †	212M	0.516	0.541	0.189	0.404	0.371	0.566	0.431
EuroBERT-610m	609M	0.408	0.518	0.232	0.093	0.134	0.408	0.299
EuroBERT-610m †	609M	0.483	0.516	0.288	0.263	0.352	0.610	0.419
EuroBERT-2b	2.11B	0.304	0.501	0.186	0.060	0.065	0.265	0.230
EuroBERT-2b †	2.11B	0.437	0.570	0.191	0.460	0.420	0.637	0.452

Table 12: Results on *MTEB(deu, v1)* of the German MTEB Benchmark. For each task type, scores were averaged across respective unique tasks. We provide results for basis models as well as after supervised training on mMARCO †. In all cases, evaluation was done in a zero-shot fashion without further finetuning on the above tasks. Best scores are indicated in bold.

Model	Params	Classification Average	PairClassification Average	Clustering Average	Reranking Average	Retrieval Average	STS Average	Average
LLäMmlein2Vec (ext1)	120M	0.546	0.529	0.261	0.139	0.224	0.188	0.315
LLäMmlein2Vec † (ext1)	120M	0.599	0.575	0.308	0.325	0.425	0.592	0.471
LLäMmlein2Vec (ext2)	120M	0.457	0.525	0.202	0.118	0.117	0.205	0.271
LLäMmlein2Vec † (ext2)	120M	0.607	0.588	0.295	0.305	0.339	0.498	0.439
LLäMmlein2Vec (ext1+2)	120M	0.339	0.530	0.098	0.046	0.009	0.107	0.188
LLäMmlein2Vec † (ext1+2)	120M	0.517	0.563	0.263	0.251	0.355	0.554	0.417
LLäMmlein2Vec (ext1)	1B	0.641	0.542	0.308	0.183	0.276	0.442	0.399
LLäMmlein2Vec † (ext1)	1B	0.670	0.625	**0.343**	0.433	0.511	0.660	0.540
LLäMmlein2Vec (ext2)	1B	0.617	0.541	0.299	0.189	0.280	0.431	0.393
LLäMmlein2Vec † (ext2)	1B	0.666	0.622	0.330	0.433	0.499	0.644	0.532
LLäMmlein2Vec (ext1+2)	1B	0.337	0.538	0.075	0.062	0.010	0.217	0.206
LLäMmlein2Vec † (ext1+2)	1B	0.647	0.611	0.325	0.421	0.481	0.640	0.521
LLäMmlein2Vec (ext1)	7B	0.683	0.558	0.249	0.169	0.266	0.333	0.376
LLäMmlein2Vec † (ext1)	7B	**0.687**	0.636	0.339	**0.477**	**0.522**	**0.682**	**0.557**
LLäMmlein2Vec (ext2)	7B	0.679	0.555	0.323	0.187	0.309	0.462	0.419
LLäMmlein2Vec † (ext2)	7B	0.683	0.628	0.337	0.471	0.517	0.680	0.553
LLäMmlein2Vec (ext1+2)	7B	0.349	0.525	0.072	0.047	0.005	0.182	0.197
LLäMmlein2Vec † (ext1+2)	7B	0.677	0.615	0.327	0.460	0.506	0.663	0.541
ModernGBERT	134M	0.639	0.537	0.293	0.139	0.241	0.449	0.383
ModernGBERT †		0.602	0.606	0.303	0.364	0.432	0.602	0.485
ModernGBERT + ext1+2	134M	0.642	0.536	0.296	0.120	0.213	0.445	0.376
ModernGBERT † + ext1+2		0.629	0.612	0.312	0.404	0.446	0.606	0.501
ModernGBERT	1B	0.665	0.544	0.318	0.097	0.199	0.418	0.374
ModernGBERT †		0.659	0.641	0.339	0.463	0. 511	0.681	0.549
ModernGBERT + ext1+2	1B	0.659	0.540	0.307	0.088	0.191	0.410	0.366
ModernGBERT † + ext1+2		0.659	0.654	0.338	0.459	0.513	**0.682**	0.551

Table 13: Results on *MTEB(deu, v1)* of the German MTEB Benchmark. For each task type, scores were averaged across respective unique tasks. We provide results for basis models as well as after supervised training on mMARCO †. In all cases, evaluation was done in a zero-shot fashion without further finetuning on the above tasks. Best scores are indicated in bold.

Model	Params	<1,024 tok.	1,024 to 4,095 tok.	4,096 to 8,192 tok.	Overall
LLäMmlein	120M	0.286	0.124	0.049	0.091
LLäMmlein	1B	0.517	0.230	0.088	0.165
LLäMmlein	7B	0.529	0.310	0.122	0.216
LLäMmlein2Vec (ext1)	120M	0.315	0.206	0.044	0.120
LLäMmlein2Vec (ext2)	120M	0.252	0.047	0.000	0.031
LLäMmlein2Vec (ext1+2)	120M	0.055	0.001	0.000	0.003
LLäMmlein2Vec (ext1)	1B	0.588	0.448	0.232	0.333
LLäMmlein2Vec (ext2)	1B	0.555	0.297	0.003	0.144
LLäMmlein2Vec (ext1+2)	1B	0.462	0.209	0.033	0.123
LLäMmlein2Vec (ext1)	7B	0.597	0.207	0.000	0.111
LLäMmlein2Vec (ext2)	7B	0.605	0.176	0.000	0.099
LLäMmlein2Vec (ext1+2)	7B	0.580	0.327	0.000	0.156
ModernGBERT	134M	0.552	0.168	0.013	0.105
ModernGBERT + ext1	134M	0.536	0.410	0.238	0.323
ModernGBERT + ext1+2	134M	0.540	0.393	0.201	0.296
ModernGBERT	1B	0.556	0.233	0.023	0.136
ModernGBERT + ext1	1B	**0.617**	0.506	**0.406**	**0.457**
ModernGBERT + ext1+2	1B	0.601	**0.526**	0.383	0.451

Table 14: QA-NIAH results. Metric is Exact Match. All tokens are counted per the model's respective tokenizer. ModernGBERT models marked with "w/o ext" refer to the model checkpoint after pre-training but before the context extension phases, those marked with "w/o ext2" to the models after extension phase one, but before phase two.

		Short		Long	
Model	Params	Fixed Length	Variable Length	Fixed Length	Variable Length
$GBERT_{Base}$	111M	2.33 ± 0.13	5.25 ± 0.27	–	–
$GBERT_{Large}$	337M	7.25 ± 0.77	15.70 ± 1.66	–	–
gerturax-3	135M	4.13 ± 0.40	8.26 ± 0.81	–	–
$GeBERTa_{Base}$ †	139M	9.79 ± 0.04	19.40 ± 0.08	–	–
$GeBERTa_{Large}$ †	406M	27.30 ± 0.09	54.10 ± 0.40	–	–
$GeBERTa_{XLarge}$ †	887M	42.20 ± 0.36	83.80 ± 0.70	–	–
$XLM\text{-}RoBERTa_{Base}$	279M	2.28 ± 0.08	5.05 ± 0.19	–	–
$XLM\text{-}RoBERTa_{Large}$ †	561M	7.27 ± 0.53	15.90 ± 1.04	–	–
$XLM\text{-}RoBERTa_{XLarge}$ †	3.48B	57.70 ± 0.37	123.00 ± 0.71	–	–
LLäMmlein2Vec	120M	3.74 ± 0.75	7.17 ± 0.53	6.69 ± 0.14	8.39 ± 0.35
LLäMmlein2Vec	1B	27.30 ± 0.16	53.90 ± 0.37	42.70 ± 0.12	59.70 ± 0.30
LLäMmlein2Vec	7B	143.00 ± 0.22	288.00 ± 0.52	180.00 ± 0.19	304.00 ± 0.41
ModernGBERT	134M	3.60 ± 0.29	3.70 ± 0.74	5.42 ± 0.33	4.71 ± 0.75
ModernGBERT	1B	22.60 ± 0.40	22.50 ± 0.18	28.70 ± 0.31	26.20 ± 0.36

Table 15: Model Throughput. Numbers are seconds per million tokens. All models were run on an RTX A6000 with Bfloat16 data type and with Flash Attention 2, except models with †, which did not implement Flash Attention 2. Reported uncertainty is the empirical standard deviation on 10 repetitions.