SuperGLEBer: German Language Understanding Evaluation Benchmark

Anonymous ACL submission

Abstract

We assemble a broad Natural Language Understanding benchmark suite for the German language and consequently evaluate a wide array of existing German-capable models in order to create a better understanding of the current state of German LLMs. Our benchmark consists of 29 different tasks ranging over different types like document classification, sequence tagging, document embedding and question answering. We evaluate 10 different German-pretrained models and thereby chart the landscape of German LLMs. In our comprehensive evaluation we find that encoder models are a good choice for most tasks, but also that the largest encoder model does not necessarily perform best for all tasks. We make our benchmark suite and a leaderboard publically available at upon-acceptance.com and encourage the community to contribute new tasks and evaluate more models on it.

1 Introduction

Fueled by the release of ChatGPT (OpenAI, 2022), the development of very capable, large language models (LLMs) has been accelerating, which also results in the release of more and more powerful models capable of the German language (Plüster, 2023; Jiang et al., 2023). From an NLP point of view, German is a language that apart from smaller, commonly BERT-based models traditionally has seen little attention when it comes to publicly available, explicitly for German pretrained foundational models. This now led to the situation that an increasing number of presumably very capable, German-pretrained LLMs are being released, but no established, diverse and systematic German evaluation suite for these models is available. To underline this point, we emphasize that, newly introduced German BERT-based models have historically only been evaluated on two tasks (Scheible et al., 2020; Chan et al., 2020) each, which is not enough to get a comprehensive understanding of the models capabilities. Such a German evaluation suite is desireable to properly compare and assess the abilities of existing but also newly developed models, like there is e.g. for English with GLUE (Wang et al., 2018), SuperGLUE (Wang et al., 2019) or even more recently OpenCompass (2023). Consequently researchers turned to these English evaluation suites to assess their German models and - for lack of a better solution - had to help themselves by translating very hard benchmark datasets from English to German using e.g. ChatGPT (Plüster, 2023). This arguably leads to unreliable results, as the models are evaluated on a task that has been machine-translated sometimes by the very same model these benchmarks were created to be hard to solve and understand for (Vago, 2023).

Our benchmark evaluation suite thus aims for both: 1. aggregating a diverse set of available German Natural Language Understanding (NLU) tasks, 2. identifying commonly used German-pretrained LLMs and evaluating the models on this benchmark. To this end, we select a wide range of different task types to make sure to properly assess the models' capabilities, such that our benchmark includes document classification, sequence tagging, document embedding and question answering tasks (Table 2). Like in existing LLM benchmarks for other languages (Wang et al., 2019; Hardalov et al., 2023) in this benchmark we challenge the models to perform well on a wide range of different tasks, which are not necessarily related to each other. These tasks focus on reasoning and language understanding, are sourced from public datasets across different domains. Inspired by SuperGLUE, we select tasks with a very simple input and output format to avoid "complex task-specific architectures" (Wang et al., 2019), as well as tasks that can be evaluated using a simple and intuitive metric. This rules out tasks like e.g. text generation, which is inherently hard to evaluate. In addition

to assembling this benchmark we also run an extensive evaluation of 4 encoder-only, 3 decoder-only, and 3 encoder-decoder German-capable transformer models as depicted in Table 1. In our comprehensive evaluation we find, that overall the encoder models perform best and usually consistently close to each other. Notably, the two largest models mBART and leo-7b are also performing well, despite not being encoder models, which is likely owed to their large size. Nevertheless, we did not find a clear advantage for the larger encoder model, as the gBERT-large model is not able to profit from its larger size, compared to its smaller counterparts.

We see the effort of this benchmark not as a done "once and for all" issue, but rather aim to introduce a foundation to be extended by further tasks and models in the future, to support and foster research for german LLMs. To this end we open-source our evaluation code, including a public leaderboard and aim to continously expand on this effort in the future.

Our contributions are as follows: 1. assembling a diverse benchmark for German NLU consisting of 29 different tasks, 2. comprehensively evaluating 10 different German-pretrained LLMs across various architectures on this benchmark, 3. providing this open-source evaluation framework to the community, allowing for easy extension of this benchmark in the future.

2 German Evaluation Tasks

In order to create a challenging and diverse benchmark for German NLU we select a wide range of different tasks from various different domains for our evaluation suite. We also list the included tasks as well as statistics for each dataset in the appendix in Table 2. In order to evaluate different capabilities of the pretrained models we select various different task types: text classification, sequence tagging, document embeddings and question answering.

2.1 Text Classification

Text classification describes the task of assigning a label to either an entire input document or a combination of input documents. We span a wide range of different domains and prediction targets, which we group into the following five categories.

Toxic & Offensive Language Identification Here we have two different datasets, which we evaluate separately: The task of *Offensive Language Identification* has been introduced by Wiegand et al. (2018), while *Toxic Comments Identification* has been introduced by Risch et al. (2021). For the first we evaluate on the fine-grained annotation distinguishing between three different types of offensive language ("profanity", "insult", and "abuse"), while the second is a binary classification task, where the model has to predict whether the input sentence contains toxic language or not.

Sentiment Analysis Here we cover two different levels of granularity: document-level and aspect-based sentiment analysis. The dataset introduced by Wojatzki et al. (2017) spans both granularities. First it is annotated with the sentiment expressed in the document towards the topic of "Deutsche Bahn", where all other sentiments expressed towards unrelated topics should be ignored. For a more detailed evaluation we also include the identification of sentiment expressed towards specific aspects within the input document in a multi-label classification task. There are overall 20 aspects, which can be e.g. "train_ride", "atmosphere" or "service" for which the model has to predict the sentiment towards each of these aspects as "positive", "negative" or "neutral". In the same spirit we select a second dataset for aspect-based sentiment analysis, introduced by Fehle et al. (2023), consisting of hotel reviews again annotated with the sentiment expressed towards specific aspects like "location", "food&drinks" or "service".

Text Pair Matching Next we evaluate the models ability to classify whether two input documents share a certain semantic relation. For this we select two datasets introduced in the cross-lingual benchmark XGLUE (Liang et al., 2020): Query-Ad Matching and Question-Answer Matching. Here the model has to predict whether the ad is a good fit for a given query, and whether a sentence is the answer for a given question. Furthermore we use the paraphrase identification dataset PAWS-X introduced by Yang et al. (2019), which consists of sentence pairs where the model has to predict whether the sentences are paraphrases of each other or not.

Word Sense Disambiguation The first dataset *WebCAGe* is a corpus annotated with senses from GermaNet (Henrich et al., 2012). The task defined on this dataset is to predict the correct sense of a given word in the context of the sentence; e.g. "bank" vs. "bank". Furthermore we select a second dataset by Ehren et al. (2021) focusing on the dis-

ambiguation of German verbal idioms, where the model has to predict from context whether a phrase is meant literally or figuratively; e.g. "hold your breath" vs. "hold your breath".

Other Classification Tasks First, on the same dataset as the toxic comment identification task introduced previously (2.1), we also evaluate the models ability to identify whether the input comment is *fact-claiming* or *engaging* (Risch et al., 2021). Here, fact-claiming means that the sentence contains a claim that can or should be verified/refuted by a fact-checker, while secondly engaging comments are defined as making readers join a discussion. Next, the *argument mining* task by Romberg and Conrad (2021) consists of sentences annotated with whether the sentence contains "options for actions or decisions that occur in the discussion" (major positions), "reasons that attack or support a major position or another premise" (premise), both or none. On the same dataset as the sentiment analysis task introduced previously (2.1), we evaluate the models ability to identify whether the input document is *relevant to the topic* of "Deutsche Bahn". If the German railroad company is neither directly nor indirectly (e.g. via their services) mentioned in the entire input document the label is "false". Next, the MASSIVE dataset consists of annotated *voice assistant interactions* (FitzGerald et al., 2023). The utterances by users are annotated with the *intent of the user*, which the model has to predict e.g. the concrete intent of "setting an alarm", or the intent to "play music". We include the *Natural Language Inference (NLI)* task, where the model has to predict whether a hypothesis is entailed by a premise or not. The dataset has been introduced in XNLI (Conneau et al., 2018) and was intended as a cross-lingual evaluation dataset, but we use it as a monolingual dataset for German. Lastly, we include the *news classification* task from XGLUE (Liang et al., 2020), where the model has to predict the category of the news article.

2.2 Sequence Tagging

The task of sequence tagging describes annotating every word or token from the input document with its respective class. We again span a wide range of different domains and prediction targets, which we group into the following two categories.

Named Entity Recognition NER is a common sequence tagging task, refering to annotating every token in the input document with its respective named entity class. Named entities can be persons, locations, organizations, but also more abstract entities like time or monetary values.

The first dataset is taken from *historical biodiversity literature* annotated with named entities like "persons", "locations", "organizations" or "other", as well as time and taxonomic entities (Ahmed et al., 2019), while the *EuropaParl* dataset (Faruqui and Padó, 2010) are proceedings from the European Parliament annotated with NEs like "persons", "locations" or "organizations". The next dataset was introduced by Benikova et al. (2014) and is sourced from German *Wikipedia* articles as well as various *online news sources*. Next, we also select a dataset with *legal entities* annotated within German court decisions (Leitner et al., 2019). It consists of German court decisions annotated with 19 semantic classes, like e.g. "person", "lawyer", "country", "organization" but also more domain-specific classes like "European legal norm", "regulation" or "contract". Lastly, we take the NER datasets from the cross-lingual benchmark XGLUE (Liang et al., 2020), which is a subset of a German news dataset by Tjong Kim Sang and De Meulder (2003) annotated with "Person", "Location", "Organization" and "Miscellaneous" entities.

Other Sequence Tagging Tasks On the *Universal Proposition Banks* by (Akbik et al., 2015), we evaluate the models abilities to predict POS tags, as well as dependency parse tree labels in two separate tasks. Furthermore, again on the MASSIVE dataset introduced previously (2.1) we also evaluate the models ability to identify "arguments" in the user's utterance; e.g. "weck mich [date : diese woche] um [time : fünf uhr morgens] auf". Lastly, on the sentiment dataset by Wojatzki et al. (2017) also used in Section 2.1 we evaluate the models ability to identify the concrete opinion term expressing the sentiment in the input document.

2.3 Document Embeddings

Document embeddings tasks evaluate the models capabilities to generate semantically meaningful vector representations for the input documents. Semantically similar documents should be placed closer together in the model's embedding space than unrelated documents. For this we use the PAWS-X (Yang et al., 2019) dataset, which consists of sentence pairs annotated with whether the sentences are paraphrases of each other or not.

2.4 Question Answering

Our last task type is extractive question answering, where the model has to answer a question given an input document. We evaluate this on two different datasets: GermanQuAD (Möller et al., 2021) and MLQA (Lewis et al., 2020). MLQA was intended to be a cross-lingual evaluation dataset, but we use it as a mono-lingual dataset for German.

3 Training Methodology

3.1 Training Methodology by LLM Type

Depending on the of transformer architecture, we use different training approaches, each tailored to the specific model: we distinguish between encoder-only, decoder-only and encoder-decoder models and follow the established training approaches for the respective model type. For transformers following the *encoder* or *decoder* architecture, we finetune the text classification tasks using the standard approach of adding a linear layer on top of the output representation of the CLS token, while for sequence tagging tasks we use the same approach, but train the linear layer to predict the correct class on top of the output representation of each input token individually. For the document embedding we follow the SentenceBERT (Reimers and Gurevych, 2019) approach and finetune the model using a triplet loss with negative sampling on the mean-pooled final output representations of the model. When finetuning for extractive question answering, we again follow the standard approach of adding a linear layer on top of the output representations of the input tokens, and train the linear layer to predict the start and end token of the answer span. For transformer models following the *encoder+decoder* architecture, we follow common practice in discarding the models decoder entirely for classification, sequence tagging and embedding tasks, and only finetune the encoder part of the model as described above and for question answering tasks we add the span extraction head on top of the decoder output.

3.2 Training Procedure for the Task Types

For each of the task types we implement the training routine as described above using an established, publicly available library. That is, for text classification and sequence classification we use FLAIR (Akbik et al., 2019), for question answering and text generation we use the reference training loops provided by HuggingFace's Transformers (Wolf et al., 2020), and for document embeddings we use the reference script provided by the Sentence-Transformers (Reimers and Gurevych, 2019) library. For all models we use the same training procedure: We use the same default hyperparameters across all models and libraries, and the same fixed seed. These are: a batch size of 8, a learning rate of 5e-5, 5 epochs. We also introduce a maximum input sequence length of 512 tokens and class weighting for all classification tasks during training. Furthermore, we consequently opt to use QLoRA-training (Dettmers et al., 2023) for all models where it is supported by the HuggingFace library (2020). If not supported by the library we skip the quantization steps and fall back to LoRA (Hu et al., 2022), which in our case applies only to the BERT models. We do this, because not all models could be trained on a single A100 GPU, hence we use QLoRA-training to reduce the memory footprint of the larger models to make training them on a single GPU feasible. Consequently enabling (Q)LoRA for all models ensures comparability between different models and rules out the possibility that the performance difference between models stems from different training procedures. We again closely follow the hyperparameters given by Dettmers et al. (2023): 4-bit quantization, double quantization and NormalFloat4.

3.3 Evaluation Metrics

As mentioned previously, we select tasks that can be evaluated using a simple and intuitive metric. When a metric has been used on the original dataset, we keep this metric for this dataset. We list the metrics used for each task in the appendix in Table 2. Used metrics are micro F1, macro F1, accuracy for classification and tagging tasks, mean-token-F1 (Lewis et al., 2020) for QA tasks (all defined in the range of 0 to 1), as well as pearson correlation calculated on cosine similarity for document embedding tasks (defined in the range of -1 to 1). For all metrics higher values indicate better performance. For the sake of creating a benchmark evaluation suite we we follow other benchmarks (2019; 2020; 2023) and average across tasks and thereby also across metrics. For all tasks we calculate the metric with the native implementation included in the used framework.

4 Evaluated Models

In our evaluation we aim to cover a large number of different models and model types available for the German language (Table 1) and evaluate these models on the tasks introduced in Section 2. We evaluate a range of different models and architectures, including encoder-only, decoder-only, and encoder-decoder models. The models have been pretrained on different datasets, some of which are multilingual, while others are monolingual German. We refer to the models by their respective HuggingFace (2020) model identifier and compare their parameter count in Table 3 in the appendix.

We evaluate *three different BERT* models, one being "bert-base-german-cased", pretrained on 12 GB of wikipedia, legal documents and news. The other two BERTs have been pretrained by Chan et al. (2020) and only differ in size: "deepset/gbert-base" and "deepset/gbert-large". Both models have been pretrained on 163.4 GB of German text, mostly consisting of OSCAR, enriched with OPUS, Wikipedia and legal documents. We also evaluate "uklfr/gottbert-base" (Scheible et al., 2020), which is a *RoBERTa* model pretrained on 145 GB of OSCAR, Wikipedia and a book corpus.

For decoder models we evaluate "dbmdz/german-gpt2" (Schweter, 2020), which is a GPT2 model pretrained on about 16 GB of German text, consisting of subtitles, and a diverse set of web crawls like CommonCrawl and news. "LeoLM/leo-hessianai-7b" is a very recent, comparably large language model, finetuned from a LLaMA2 checkpoint using German text (Plüster, 2023) mostly sourced from OSCAR and has only been evaluated on a machine-translated version of the English OpenLLM dataset. Furthermore, we also consider the multilingual-trained "bigscience/bloomz-560m" model (Muennighoff et al., 2023). It was trained in two steps: first on a 1.5 TB multilingual corpus of 45 languages and 12 programming languages using causal language modeling (Workshop et al., 2023), then further multilingual, multi-task pretraining using supervised tasks (Muennighoff et al., 2023).

We also evaluate the encoder-decoder multilingual-trained "bigscience/mt0-small" model (Muennighoff et al., 2023), which was finetuned analogously to the previously introduced Bloomz model, but is instead finetuned from the "google/mt5-small" checkpoint. This model in turn was trained on 101 languages, including German, using the "span-corruption" objective (Xue et al., 2021) on the C4 corpus (Raffel et al., 2020) and is also included in our evaluation. Lastly we evaluate the multilingual-trained "facebook/mbart-large-50" model, trained on 50 languages, including German, using the translation objective (Liu et al., 2020). In contrast to BART, the mBART model was only trained on the translation objective between any pair of languages and not additionally on the denoising objective, thus never saw German text as input and target at the same time.

5 Evaluation

We extensively evaluate the models from Section 4 on the tasks introduced in Section 2 resulting in Table 1. Here the results are averaged by the various task types at varying levels of granularity. The columns reading "avg" have been averaged across the averages of the respective task types, in order to not overweight any task type for which more datasets exist, i.e. all "NER" tasks have been averaged into a single value before averaging across all tagging tasks. We also list the results for the individual tasks in the appendix in Appendix D. In the following we will discuss the results under various different aspects.

5.1 Performance by Model and Task Type

For **classification** tasks we find that the *encoder-models* all perform overall very similar to each other (70.1 to 72.7), despite differences in the training data and even model size and architecture. Interestingly, within the classification tasks the models don't perform equally well on all tasks. For example the gBERT-large model performs above average for NLI, sentiment analysis, text pair matching, as well as word sense disambiguation, but at the same time below average for toxicity detection. On average the largest encoder model is thus even the worst performing encoder model. For the *encoder+decoder* models there is a clear distinction in performance between the mT5 and mT0 models (46.6 and 53.4) on the one hand and the mBART model (63.2) on the other hand. The mBART model performs much better across most classification tasks, often even being competetive with the encoder models. We find that mT5 performs consistently worse than its further pretrained mT0 counterpart, with the only exception being the sentiment analysis task. Within the *decoder* models GPT2 model performs similarly to the bloomz

type	model	classification						tagging			embedding	QA
		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
encoder	gbert-base	0.548	0.626	0.725	0.774	0.758	0.725	0.739	0.810	0.796	0.533	0.813
	gbert-large	0.433	0.704	0.812	0.851	0.702	0.701	0.754	0.806	0.795	0.651	0.826
	gottbert	0.551	0.538	0.725	0.816	0.746	0.714	0.699	0.800	0.779	0.558	0.762
	bert-base-german-cased	0.531	0.638	0.680	0.823	0.760	0.727	0.712	0.795	0.778	0.534	0.803
	encoder average	0.516	0.627	0.736	0.816	0.741	0.717	0.726	0.802	0.787	0.569	0.801
enc+dec	mbart-large-50	0.506	0.561	0.770	0.815	0.615†	0.632†	0.741	0.800	0.788	0.620	0.829
	mt5-small	0.181	0.361	0.571	0.704	0.473	0.466	0.380	0.680	0.620	0.321	0.700
	mt0-small	0.332	0.344	0.617	0.763	0.545	0.534	0.455	0.690	0.643	0.512	0.789
	enc+dec average	0.339	0.422	0.653	0.760	0.544†	0.544†	0.526	0.723	0.684	0.484	0.772
decoder	german-gpt2	0.453	0.600	0.670	0.799	0.733	0.696	0.619	0.746	0.721	0.353	0.815
	bloomz-560m	0.463	0.431	0.734	0.806	0.700	0.667	0.154‡	0.615‡	0.522‡	0.329	0.784
	leo-hessianai-7b	0.603	0.764	0.812	0.895	0.836	0.812	0.733	0.666♡	0.680♡	0.587	unsup.◇
	decoder average	0.506	0.598	0.739	0.833	0.756	0.725	0.502	0.676‡♡	0.641‡♡	0.423	0.533◇
	overall average	0.460	0.557	0.712	0.804	0.687†	0.667†	0.598‡	0.741‡♡	0.712‡♡	0.500	0.712◇

Table 1: Results of our models on various tasks, averaged at varying levels of granularity. The columns reading "avg" have been averaged across the averages of the respective task types, in order to not overweight any task type for which more datasets exist, i.e. all "NER" tasks have been averaged into a single value before averaging across all tagging tasks. The second row gives the type of metric used for the respective task type. Here "mixed" means that - like in other benchmarks (2019; 2020; 2023) - at least two kind of metrics have been averaged together. The results marked with † have been averaged over tasks for which a "Cuda OOM" error occured on an A100 80GB GPU (only mBART). The results marked with ‡ have been averaged over tasks where a "ShapeError" occured (only Bloomz). The results marked with ♡ have been averaged over tasks for which the results could not be calculated in time for the submission deadline. This is only the case for a single task for the comparably large leo-7b model - this result will be included in the final version of this paper. The results marked with ◇ have been averaged over tasks where the HuggingFace implementation does not (yet) support the task type. All these symbols have been placed at all averages this affects transitively. All missing values have been treated as a 0.0 when calculating the average.

model (69.6 and 66.7), while the leo-7b model performs significantly better (81.2). Here the leo-7b model comfortably ranks first place across all models, which is likely owed to its significantly larger size and training data. The GPT2 model also performs reasonably well, but is still outperformed by all encoder models.

Overall we find that the encoder models perform best across all classification tasks, and rank overall places 2-5 across all models, with the best performing encoder model being bert-base-german-cased, only getting beat by leo-7b. mT5 and mT0 perform worst across all models, with mT0 performing better than mT5.

For **sequence tagging** tasks the *encoder* models again perform very similar to each other, with the gBERT-large model performing as good as its smaller counterpart. Here the encoder-models rank places 1,2,4 and 5 across all models. Along the *encoder+decoder* models the mBART model again performs clearly best, with the mT0 and mT5 again placing at the bottom of the ranking. mBART is even competetive with the encoder-only models, ranking place 3 across all models, while GPT2 is the best performing *decoder* and bloomz is performing worst overall (52.2). The leo-7b model always performed slightly below or roughly at average of all other models, only dominating by a large margin for the NER task on the EuroParl dataset. GPT2 is the best performing decoder model for sequence tagging, but is again outperformed by the encoder models and mBART.

Analysing the **document embedding** tasks the encoder models performance varies drastically (53.3 to 65.1), with gBERT-large performing best by a large margin (rank 1). The other three encoder models are comfortably outperformed by two non-encoder models, namely mBART (rank 2) and leo-7b (rank 3). We find that GPT2, bloomz and mT5 perform similarly bad, while mT0 is closer to the small encoder models.

For **QA** performance all models are very close to each other. We find mBART to perform best (82.9), followed by gBERT-large (82.6) and GPT2 (81.5).

Overall we find that depending on the task type different models perform best, but a clear trend is that the encoder models are always among the top. The size of the encoder models does not seem to have a large impact on the performance, as the gBERT-large model does not have a clear advantage over its smaller counterpart, except in the document embedding tasks. The mBART model performs best across the evaluated encoder-decoder models, often being competetive with the encoder models, only being outperformed by them on the classification tasks. Furthermore, the pretraining of the mT0 model seems to have a positive effect on the performance for German, as it very consistently performs better than the mT5 model across all task types, often by a large margin. It is clear that the leo-7b model performs best across all decoder models for most task types, while the bloomz model clearly performs worst. Given that mBART and leo-7b are both the largest models in the benchmark, it is not surprising that they perform best across most task types. At the same time gBERT-large is not able to profit from its larger size, as it is commonly outperformed or matched by the smaller encoder models.

5.2 Performance Stability Across Seeds

To make sure that the results are not a fluke of the random initialization of the models, we evaluate the models on the same tasks using different random seeds. At the size of this benchmark running the entire evaluation for all models and tasks for multiple seeds becomes computationally prohibitive (Appendix A), so we select one encoder and one decoder model, as well as three tasks to evaluate the stability of the results on. We run the entire fine-tuning and evaluation an additional four times for each selected model and task, using a different random seed each time. For this experiment, we select the gBERT-base model, as well as the german-GPT2 model and for the task types we select the verbal idioms classification task, the biodiversity NER task and the PAWS-X document embedding task. We list detailed results in the appendix in Table 5 and find the results to be very stable across the different seeds with an average standard deviation of the results being below 0.012 across tasks and models.

5.3 Performance w. and w/o. (Q)LoRA

As we exclusively use (Q)LoRA for our training in order to keep the models small and the results comparable across models, we also conduct a small evaluation of the performance of the models with and without (Q)LoRA training. For this we select the same models and tasks as in Section 5.2 and train them without (Q)LoRA once. For this we use the same hyperparameter configuration and seed as for the (Q)LoRA training, but train the

models using full precision. We list the results alongside in Table 5 and find that there is a significant performance difference between the (Q)LoRA and non-(Q)LoRA training. The performance drop ranges from 0.019 to 0.090 across tasks and models. We explicitly welcome non-(Q)LoRA trained models in the benchmark evaluation leaderboards, but also encourage further research into the performance of (Q)LoRA training and its impact on the performance of the models. We also plan on differentiating between various training approaches in the benchmark, making it possible to compare the performance across different training methods.

6 Related Work

GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019) are two of the most prominent LLM benchmarks, consisting of 11 and 10 different NLU tasks respectively. These benchmarks only being available in English has quickly been identified as an issue for the evaluation of non-English models by the NLP community. Thus the development of various similar benchmarks for other languages followed, like e.g. for Russian (Shavrina et al., 2020), Persian (Khashabi et al., 2021), or recently for Bulgarian (Hardalov et al., 2023). These benchmarks are all similar in their setup, aiming to assess the models abilities on a wide range of different tasks.

Cross- and multilingual benchmarks like XTREME (Hu et al., 2020) and XGLUE (Liang et al., 2020) on the other hand have been designed to evaluate the models' cross-lingual capabilities. For this they consist of 9 tasks spread across 5 to 40 languages for XTREME and 11 tasks across 3 to 18 languages for XGLUE. Thus they also include tasks in German, but neither the focus of the evaluation nor for the model itself is on German. The general idea behind these benchmarks is to evaluate the models' ability to transfer knowledge from one language to another, but not to evaluate the models' capabilities in a single language. Using these benchmarks as a basis for evaluating German models is thus not ideal, as the tasks are commonly accompanied by a rather small German training set, because the focus is on learning from the combined training data of all languages.

As mentioned earlier, in the advent of increasingly large LMs, the need for German evaluation benchmarks has been recognized, but in the absence of German focused benchmarks, the evaluation is commonly done by machine-translating existing English evaluation datasets (Plüster, 2023), which can give an estimate of the performance of a model, but is not a reliable evaluation of the models' capabilities (Vago, 2023).

Although there exists no diverse and comprehensive evaluation benchmark for German LLMs, on which the various capabilities of different models are evaluated, there have been efforts to evaluate German models on a specific task, like sentiment analysis (Cieliebak et al., 2017), coreference resolution (Schröder et al., 2021), utterance similarity (Asaadi et al., 2022), inclusive language (Pomerenke, 2022) or document clustering (Wehrli et al., 2023). The evaluation of models on these benchmarks is usually not comprehensive, with only few models evaluated on a single task, and usually only a single model architecture - commonly encoder models - being evaluated. Overall, there is no established, easily runnable evaluation framework for multiple German tasks, which makes it hard to compare results across different models.

7 Conclusion

We introduce the first large and diverse German language understanding benchmark for language models, consisting of 29 different tasks and covering 4 different task types: text classification, sequence tagging, document embeddings and question answering. The text classification and sequence tagging tasks themselves contain a wide range of different language understanding tasks, covering various different domains and prediction targets.

We evaluate 10 different models, including 4 encoder-only, 3 decoder-only and 3 encoder-decoder models on our newly introduced benchmark. In our comprehensive evaluation we find, that on average the encoder models perform best and are usually close to each other in performance on the classification and sequence tagging tasks. Despite not being encoder models, the two largest evaluated models mBART and leo-7b are also performing well. In contrast, we did not find a clear advantage for the larger encoder model, as the gBERT-large model is not able to profit from its larger size, often being outperformed or matched by its smaller counterparts. We make the benchmark and leaderbord publicly available and encourage the community to contribute tasks as well as models to the benchmark, thereby mapping the landscape of German LLMs.

Limitations

7.1 Training Procedure

Some of the used frameworks (FLAIR & Sentence-Transformers) only support training on a single GPU, which inherently limits the size of the models we can evaluate using our framework. We thus opt for QLoRA-training here to reduce the memory footprint of the larger models and make training them on a single GPU feasible.

As mentioned in Table 1 we encounter some issues with the training procedure of the mBART model (OutOfMemory), as well as the training of the bloomz model (ShapeError). The first seem to be an issue between the bitsandbytes quantization library and the mBART model, while the second seems to be imcompatibilities between the used framework and the respective model, which we could not easily resolve. We will investigate these issues further and update the results accordingly, if we find a solution. Furthermore, for the LLaMa2 architecture no QA-model is implemented within the HuggingFace library, but we will update the results once a QA-model is available.

7.2 Representativeness of the Results

As we train and evaluate all models using QLoRA, we cannot make any statements about the performance of the models without QLoRA. Our exemplary evaluation of the models with and without QLoRA training (Section 5.3) shows that there is a performance difference between the two training procedures, which is acceptable for our purposes, as we evaluate all models using the same training procedure, thus keeping the results comparable. Furthermore we do not limit our leaderboard to QLoRA-trained models, but also explicitly welcome non-QLoRA-trained models, or even the same models trained without QLoRA.

Next, we only evaluate a single hyperparameter configuration for each model, which is the default configuration of the respective library. We leave the evaluation of different hyperparameter configurations to future work and do not limit the leaderboard to the default configuration of the respective library.

We only report the results for the same random seed for each model and task and conduct a small evaluation of the stability of the results across different seeds (Section 5.2). We find the results to be stable across different seeds, such that we are confident in our results reported in Table 1.

For some models, like the mT0, mT5, bloomz and leo-7b we evaluated only the smallest model size, as otherwise computing the benchmark results for all model sizes would have been computationally prohibitive (Appendix A). Nevertheless we encourage the community to contribute results for the larger model sizes, but also plan to add larger versions of used models to the benchmark in the future ourselves.

Ethics Statement

As we only include publicly available datasets and models, we do not see any ethical issues with this work. We only select datasets and tasks, where the intended use of the data is clearly to be used for research.

Intended Use We intend this benchmark to be used for the evaluation of German LLMs. To this end we make the benchmark and leaderboard publicly available and encourage the community to contribute tasks as well as models to the benchmark. For this we provide an open-source evaluation framework, which can be easily extended to include new tasks and models and publish it under an open-source license.

Acknowledgments

upon acceptance

References

Sajawel Ahmed, Manuel Stoeckel, Christine Driller, Adrian Pachzelt, and Alexander Mehler. 2019. BIOfid dataset: Publishing a German gold standard for named entity recognition in historical biodiversity literature. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 871–880, Hong Kong, China. Association for Computational Linguistics.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, Minneapolis, Minnesota. Association for Computational Linguistics.

Alan Akbik, Laura Chiticariu, Marina Danilevsky, Yunyao Li, Shivakumar Vaithyanathan, and Huaiyu Zhu. 2015. Generating high quality proposition Banks for multilingual semantic role labeling. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International*

Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 397–407, Beijing, China. Association for Computational Linguistics.

Shima Asaadi, Zahra Kolagar, Alina Liebel, and Alessandra Zarcone. 2022. GiCCS: A German in-context conversational similarity benchmark. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 351–362, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Padó. 2014. Germeval 2014 named entity recognition shared task.

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.

Mark Cieliebak, Jan Milan Deriu, Dominic Egger, and Fatih Uzdilli. 2017. A Twitter corpus and benchmark resources for German sentiment analysis. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 45–51, Valencia, Spain. 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.

OpenCompass Contributors. 2023. Opencompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

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

Rafael Ehren, Timm Lichte, Jakub Waszczuk, and Laura Kallmeyer. 2021. Shared task on the disambiguation of german verbal idioms at konvens 2021. *Proceedings of the Shared Task on the Disambiguation of German Verbal Idioms at KONVENS*.

Manaal Faruqui and Sebastian Padó. 2010. Training and evaluating a german named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010*, Saarbrücken, Germany.

Jakob Fehle, Leonie Münster, Thomas Schmidt, and Christian Wolff. 2023. Aspect-based sentiment analysis as a multi-label classification task on the domain of german hotel reviews. In *Proceedings of KONVENS 2023*, Ingolstadt, Germany.

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.

Momchil Hardalov, Pepa Atanasova, Todor Mihaylov, Galia Angelova, Kiril Simov, Petya Osenova, Veselin Stoyanov, Ivan Koychev, Preslav Nakov, and Dragomir Radev. 2023. bgGLUE: A Bulgarian general language understanding evaluation benchmark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8733–8759, Toronto, Canada. 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.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. *CoRR*, abs/2003.11080.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Daniel Khashabi, Arman Cohan, Siamak Shakeri, Pedram Hosseini, Pouya Pezeshkpour, Malihe Alikhani, Moin Aminnaseri, Marzieh Bitaab, Faeze Brahman, Sarik Ghazarian, Mozhdeh Gheini, Arman Kabiri, Rabeeh Karimi Mahabagdi, Omid Memarrast, Ahmadreza Mosallanezhad, Erfan Noury, Shahab Raji, Mohammad Sadegh Rasooli, Sepideh Sadeghi, Erfan Sadeqi Azer, Niloofar Safi Samghabadi, Mahsa Shafaei, Saber Sheybani, Ali Tazarv, and Yadollah Yaghoobzadeh. 2021. ParsiNLU: A Suite of Language Understanding Challenges for Persian. *Transactions of the Association for Computational Linguistics*, 9:1147–1162.

Elena Leitner, Georg Rehm, and Julian Moreno-Schneider. 2019. Fine-grained Named Entity Recognition in Legal Documents. In *Semantic Systems. The Power of AI and Knowledge Graphs. Proceedings of the 15th International Conference (SEMANTiCS 2019)*, number 11702 in Lecture Notes in Computer Science, pages 272–287, Karlsruhe, Germany. Springer. 10/11 September 2019.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

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.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

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, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2023. Crosslingual generalization through multitask finetuning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15991–16111, Toronto, Canada. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt.

Björn Plüster. 2023. Leolm: Igniting german-language llm research. We assume preprint to be available until publication.

David Pomerenke. 2022. Inclusify: A benchmark and a model for gender-inclusive german.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1).

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.

Julia Romberg and Stefan Conrad. 2021. Citizen involvement in urban planning - how can municipalities be supported in evaluating public participation processes for mobility transitions? In *Proceedings of the 8th Workshop on Argument Mining*, pages 89–99, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Raphael Scheible, Fabian Thomczyk, Patric Tippmann, Victor Jaravine, and Martin Boeker. 2020. Gottbert: a pure german language model.

Fynn Schröder, Hans Ole Hatzel, and Chris Biemann. 2021. Neural end-to-end coreference resolution for German in different domains. In *Proceedings of the 17th Conference on Natural Language Processing (KONVENS 2021)*, pages 170–181, Düsseldorf, Germany. KONVENS 2021 Organizers.

Stefan Schweter. 2020. German gpt-2 model.

Tatiana Shavrina, Alena Fenogenova, Emelyanov Anton, Denis Shevelev, Ekaterina Artemova, Valentin Malykh, Vladislav Mikhailov, Maria Tikhonova, Andrey Chertok, and Andrey Evlampiev. 2020. RussianSuperGLUE: A Russian language understanding evaluation benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4717–4726, Online. Association for Computational Linguistics.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Vago. 2023. Vagosolutions/mt-bench-truegerman.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. *SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems*. Curran Associates Inc., Red Hook, NY, USA.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Silvan Wehrli, Bert Arnrich, Thomas Schmidt, and Christopher Irrgang. 2023. German text embedding clustering benchmark. In *Proceedings of KONVENS 2023*, Ingolstadt, Germany.

Michael Wiegand, Melanie Siegel, and Josef Ruppenhofer. 2018. *Overview of the GermEval 2018 Shared Task on the Identification of Offensive Language*. oeaw, Vienna.

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.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, Jonathan Tow, Alexander M. Rush, Stella Biderman, Albert Webson, Pawan Sasanka Ammanamanchi, Thomas Wang, Benoît Sagot, Niklas Muennighoff, Albert Villanova del Moral, Olatunji Ruwase, Rachel Bawden, Stas Bekman, Angelina McMillan-Major, Iz Beltagy, Huu Nguyen, Lucile Saulnier, Samson Tan, Pedro Ortiz Suarez, Victor Sanh, Hugo Laurençon, Yacine Jernite, Julien Launay, Margaret Mitchell, Colin Raffel, Aaron Gokaslan, Adi Simhi, Aitor Soroa, Alham Fikri Aji, Amit Alfassy, Anna Rogers, Ariel Kreisberg Nitzav, Canwen Xu, Chenghao Mou, Chris Emezue, Christopher Klamm, Colin Leong, Daniel van Strien, David Ifeoluwa Adelani, Dragomir Radev, Eduardo González Ponferrada, Efrat Levkovizh, Ethan Kim, Eyal Bar Natan, Francesco De Toni, Gérard Dupont, Germán Kruszewski, Giada Pistilli, Hady Elsahar, Hamza Benyamina, Hieu Tran, Ian Yu, Idris Abdulmumin, Isaac Johnson, Itziar Gonzalez-Dios, Javier de la Rosa, Jenny Chim, Jesse Dodge, Jian Zhu, Jonathan Chang, Jörg Frohberg, Joseph Tobing, Joydeep Bhattacharjee, Khalid Almubarak, Kimbo Chen, Kyle Lo, Leandro Von Werra, Leon Weber, Long Phan, Loubna Ben allal, Ludovic Tanguy, Manan Dey, Manuel Romero Muñoz, Maraim Masoud, María Grandury, Mario Šaško, Max Huang, Maximin Coavoux, Mayank Singh, Mike Tian-Jian Jiang, Minh Chien Vu, Mohammad A. Jauhar, Mustafa Ghaleb, Nishant Subramani, Nora Kassner, Nurulaqilla Khamis, Olivier Nguyen, Omar Espejel, Ona de Gibert, Paulo Villegas, Peter Henderson, Pierre Colombo, Priscilla Amuok, Quentin Lhoest, Rheza Harliman, Rishi Bommasani, Roberto Luis López, Rui Ribeiro, Salomey Osei, Sampo Pyysalo, Sebastian Nagel, Shamik Bose, Shamsuddeen Hassan Muhammad, Shanya Sharma, Shayne Longpre, Somaieh Nikpoor, Stanislav Silberberg, Suhas Pai, Sydney Zink, Tiago Timponi Torrent, Timo Schick, Tristan Thrush, Valentin Danchev, Vassilina Nikoulina, Veronika Laippala, Violette Lepercq, Vrinda Prabhu, Zaid Alyafeai, Zeerak Talat, Arun Raja, Benjamin Heinzerling, Chenglei Si, Davut Emre Taşar, Elizabeth Salesky, Sabrina J. Mielke, Wilson Y. Lee, Abheesht Sharma, Andrea Santilli, Antoine Chaffin, Arnaud Stiegler, Debajyoti Datta, Eliza Szczechla, Gunjan Chhablani, Han Wang, Harshit Pandey, Hendrik Strobelt, Jason Alan Fries, Jos Rozen, Leo Gao, Lintang Sutawika, M Saiful Bari, Maged S. Al-shaibani, Matteo Manica, Nihal Nayak, Ryan Teehan, Samuel Albanie, Sheng Shen, Srulik Ben-David, Stephen H. Bach, Taewoon Kim, Tali Bers, Thibault Fevry, Trishala Neeraj, Urmish Thakker, Vikas Raunak, Xiangru Tang, Zheng-Xin Yong, Zhiqing Sun, Shaked Brody, Yallow Uri, Hadar Tojarieh, Adam Roberts, Hyung Won Chung, Jaesung Tae, Jason Phang, Ofir Press, Conglong Li, Deepak Narayanan, Hatim Bourfoune, Jared Casper, Jeff Rasley, Max Ryabinin, Mayank Mishra, Minjia Zhang, Mohammad Shoeybi, Myriam Peyrounette, Nicolas Patry, Nouamane Tazi, Omar Sanseviero, Patrick von Platen, Pierre Cornette, Pierre François Lavallée, Rémi Lacroix, Samyam Rajbhandari, Sanchit Gandhi, Shaden Smith, Stéphane Requena, Suraj Patil, Tim Dettmers, Ahmed Baruwa, Amanpreet Singh, Anastasia Cheveleva, Anne-Laure Ligozat, Arjun Subramonian, Aurélie Névéol, Charles Lovering, Dan Garrette, Deepak Tunuguntla, Ehud Reiter, Ekaterina Taktasheva, Ekaterina Voloshina, Eli Bogdanov, Genta Indra Winata, Hailey Schoelkopf, Jan-Christoph Kalo, Jekaterina Novikova, Jessica Zosa Forde, Jordan Clive, Jungo Kasai, Ken Kawamura, Liam Hazan, Marine Carpuat, Miruna Clinciu, Najoung Kim, Newton Cheng, Oleg Serikov, Omer Antverg, Oskar van der Wal, Rui Zhang, Ruochen Zhang, Sebastian Gehrmann, Shachar Mirkin, Shani Pais, Tatiana Shavrina, Thomas Scialom, Tian Yun, Tomasz Limisiewicz, Verena Rieser, Vitaly Protasov,

Vladislav Mikhailov, Yada Pruksachatkun, Yonatan Belinkov, Zachary Bamberger, Zdeněk Kasner, Alice Rueda, Amanda Pestana, Amir Feizpour, Ammar Khan, Amy Faranak, Ana Santos, Anthony Hevia, Antigona Unldreaj, Arash Aghagol, Arezoo Abdollahi, Aycha Tammour, Azadeh HajiHosseini, Bahareh Behroozi, Benjamin Ajibade, Bharat Saxena, Carlos Muñoz Ferrandis, Daniel McDuff, Danish Contractor, David Lansky, Davis David, Douwe Kiela, Duong A. Nguyen, Edward Tan, Emi Baylor, Ezinwanne Ozoani, Fatima Mirza, Frankline Ononiwu, Habib Rezanejad, Hessie Jones, Indrani Bhattacharya, Irene Solaiman, Irina Sedenko, Isar Nejadgholi, Jesse Passmore, Josh Seltzer, Julio Bonis Sanz, Livia Dutra, Mairon Samagaio, Maraim Elbadri, Margot Mieskes, Marissa Gerchick, Martha Akinlolu, Michael McKenna, Mike Qiu, Muhammed Ghauri, Mykola Burynok, Nafis Abrar, Nazneen Rajani, Nour Elkott, Nour Fahmy, Olanrewaju Samuel, Ran An, Rasmus Kromann, Ryan Hao, Samira Alizadeh, Sarmad Shubber, Silas Wang, Sourav Roy, Sylvain Viguier, Thanh Le, Tobi Oyebade, Trieu Le, Yoyo Yang, Zach Nguyen, Abhinav Ramesh Kashyap, Alfredo Palasciano, Alison Callahan, Anima Shukla, Antonio Miranda-Escalada, Ayush Singh, Benjamin Beilharz, Bo Wang, Caio Brito, Chenxi Zhou, Chirag Jain, Chuxin Xu, Clémentine Fourrier, Daniel León Periñán, Daniel Molano, Dian Yu, Enrique Manjavacas, Fabio Barth, Florian Fuhrimann, Gabriel Altay, Giyaseddin Bayrak, Gully Burns, Helena U. Vrabec, Imane Bello, Ishani Dash, Jihyun Kang, John Giorgi, Jonas Golde, Jose David Posada, Karthik Rangasai Sivaraman, Lokesh Bulchandani, Lu Liu, Luisa Shinzato, Madeleine Hahn de Bykhovetz, Maiko Takeuchi, Marc Pàmies, Maria A Castillo, Marianna Nezhurina, Mario Sänger, Matthias Samwald, Michael Cullan, Michael Weinberg, Michiel De Wolf, Mina Mihaljcic, Minna Liu, Moritz Freidank, Myungsun Kang, Natasha Seelam, Nathan Dahlberg, Nicholas Michio Broad, Nikolaus Muellner, Pascale Fung, Patrick Haller, Ramya Chandrasekhar, Renata Eisenberg, Robert Martin, Rodrigo Canalli, Rosaline Su, Ruisi Su, Samuel Cahyawijaya, Samuele Garda, Shlok S Deshmukh, Shubhanshu Mishra, Sid Kiblawi, Simon Ott, Sinee Sang-aroonsiri, Srishti Kumar, Stefan Schweter, Sushil Bharati, Tanmay Laud, Théo Gigant, Tomoya Kainuma, Wojciech Kusa, Yanis Labrak, Yash Shailesh Bajaj, Yash Venkatraman, Yifan Xu, Yingxin Xu, Yu Xu, Zhe Tan, Zhongli Xie, Zifan Ye, Mathilde Bras, Younes Belkada, and Thomas Wolf. 2023. Bloom: A 176b-parameter open-access multilingual language model.

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.

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.

A Putting the Compute into Perspective

We list the number of trainable parameters for each model in Table 3. This includes the number of parameters of the base model as well as the number of trainable parameters after (Q)LoRA has been applied.

Estimating the GPU hours for our experiments - especially including development and debugging - is difficult, as we did not keep track of all time spent on GPUs. Nevertheless we estimate the total GPU hours spent on the development of this benchmark to be around $1500\,\mathrm{h}$ of A100 GPU time.

B Dataset Domains and Licenses

The datasets we use in our benchmark are listed in Table 2, and are described in Section 2. In Table 4 we list the domains and licenses of the datasets.

C Training Stability

Table 5 lists the results of the training stability experiment described in Section 5.2, as well as the results of a single run without (Q)LoRA training for comparison (Section 5.3).

D Individual results

We list the detailed results of every task for every model in Tables 6 to 8. Models achieving a 0.0 score on for multi-class classification tasks are a known instability within the Flair library and occur only for large number of output classes for certain models: https://github.com/flairNLP/flair/issues/678

task type	target	task name	|Train|	|Dev|	|Test|	metric
text classification	tox.	offensive language	4508	501	3532	macro F1
		toxic comments	2920	324	944	
	sent.	sentiment polarity	20 941	2584	2566	micro F1
		DB aspect sentiment	16 200	1930	2095	
		Hotel aspect sentiment	3446	383	425	
	match	Query => Ad Matching	9000	1000	10 000	ACC
		Quest. => Ans. Matching	9000	1000	10 000	
		Paraphrase Matching	49 129	2000	2000	
	WSD	WebCAGe	8339	926	1030	micro F1
		Verbal Idioms	6902	1488	1511	
	other	Factclaiming Comments	2920	324	944	macro F1
		Engaging Comments	2920	324	944	macro F1
		CIMT: Arg. Min.	14 460	1607	1785	macro F1
		Topic Relevance	20 941	2584	2566	micro F1
		Intent Identification	13 382	1487	1652	micro F1
		NLI	2245	250	5010	ACC
		News Classification	9000	1000	10 000	ACC
sequence tagging	NER	Historical Biodiversity	12 668	1584	1584	micro F1
		EuropaParl	3184	354	858	
		Wikipedia & News	24 000	2200	5100	
		Legal	53 384	6666	6673	
		News	2587	287	3007	
	other	DEP Univ. Prop. Bank	14 118	799	977	
		POS Univ. Prop. Bank	14 118	799	977	
		MASSIVE Arguments	13 382	1487	1652	
		GermEval Opinions	19 432	2369	2566	
embedding		PAWS-X	49 129	2000	2000	pearson corr.
question answering		MLQA	512	-	4517	mean-token F1
		GermanQuAD	11 518	-	2204	

Table 2: The different datasets and tasks making up the benchmark and their associated task type.

Model	Total Params	Trainable Params	Trainable %
gbert-base	110,222,592	294,912	0.268%
gbert-large	336,522,240	786,432	0.234%
gottbert	126,279,936	294,912	0.234%
bert-base-german-cased	109,376,256	294,912	0.270%
mbart-large-50	612,059,136	1,179,648	0.193%
mt0-small	147,055,296	114,688	0.078%
mt5-small	147,055,296	114,688	0.078%
german-gpt2	124,740,864	294,912	0.236%
bloomz-560m	560,001,024	786,432	0.140%
leo-hessianai-7b	6,611,537,920	4,194,304	0.063%

Table 3: Number of parameters as well as number of trainable parameters per model after QLoRA

dataset	domain	license
EuroParl	protocol	GNU GPL
Hist. Bio. Div.	bio literature	cc-by-4.0
Legal	legal texts	cc-by-4.0
NLI	misc	OANC
WebCAGe	misc	N/A
Verbal Idioms	misc	N/A
XGLUE datasets	misc	usable for non-commercial research (N/A)
MASSIVE	spoken language, misc	cc-by-4.0
CIMT Arg Min.	dialogue	CC BY-SA
Univ. Prop. Bank	misc	CDLA-Sharing-1.0
GermanQuAD	misc	cc-by-4.0
DB Sentiment	Blogs & News	N/A
Hotel Sentiment	Reviews	N/A
XGLUE datasets	misc	N/A
PAWS-X	misc	"may be freely used" (N/A)
MLQA	misc	CC-BY-SA 3.0
toxic, fact, engag. com.	user comments	N/A
NERWikipedia & News	Wikipedia & News	CC-BY
NER News	news	N/A

Table 4: Domains and licenses for the used datasets, more details in Section 2. For our benchmark we made sure to only use datasets where the intended use of the data set clearly allows for the use in our benchmark. Nevertheless, where no license could be found (N/A), we will contact the authors to clarify the license.

amount of runs	train type		Verbal Idioms avg	Verbal Idioms sd	Bio Hist NER avg	Bio Hist NER sd	embd avg	embd sd
5	LoRA	gbert-base	0.918	0.017	0.640	0.013	0.557	0.015
	QLoRA	german-GPT2	0.902	0.007	0.499	0.016	0.355	0.003
1	no (Q)LoRA	gbert-base	0.937		0.704		0.639	
		german-GPT2	0.937		0.589		0.419	

Table 5: Training stability across five different seeds. We evaluate on the two models on the three datasets and task types described in Section 5.2. We report the average and standard deviation across the five runs. Furthermore we report the performance of a single run without (Q)LoRA training for comparison (Section 5.3).

	toxicity		sentiment			matching			WSD		other						
	macro F1 Toxic Comments	macro F1 Offensive Lang	micro F1 DB Aspect	micro F1 Hotel Aspect	micro F1 Polarity	ACC Query-Ad	ACC Quest. Ans.	ACC PAWS-X	micro F1 WebCAGe	micro F1 Verbal Idioms	macro F1 Engaging Comments	macro F1 FactClaiming Comments	ACC News Class	ACC NLI	macro F1 Argument Mining	micro F1 MASSIVE: Intents	micro F1 Topic Relevance
gbert-base	0.667	0.428	0.568	0.522	0.788	0.735	0.618	0.823	0.624	0.924	0.673	0.710	0.886	0.443	0.855	0.789	0.949
gbert-large	0.386	0.480	0.620	0.675	0.818	0.786	0.745	0.905	0.754	0.948	0.670	0.755	0.896	0.739	0.863	0.027	0.961
gottbert	0.675	0.427	0.523	0.300	0.792	0.736	0.633	0.807	0.701	0.930	0.677	0.730	0.888	0.408	0.845	0.724	0.951
bert-base-german-cased	0.628	0.434	0.581	0.563	0.771	0.716	0.591	0.734	0.722	0.923	0.687	0.717	0.883	0.569	0.842	0.675	0.948
mbart-large-50	0.639	0.372	0.490	0.416	0.776	0.775	0.699	0.836	0.714	0.915	0.660	0.700	OutOfMemory	0.475	0.844	0.700	0.927
mt0-small	0.271	0.090	0.479	0.000	0.605	0.591	0.548	0.574	0.598	0.810	0.596	0.581	0.307	0.334	0.591	0.021	0.883
mt5-small	0.502	0.162	0.479	0.000	0.552	0.643	0.593	0.616	0.715	0.810	0.610	0.567	0.699	0.334	0.592	0.117	0.894
german-gpt2	0.599	0.306	0.525	0.506	0.769	0.670	0.584	0.755	0.697	0.901	0.669	0.706	0.871	0.449	0.806	0.690	0.942
bloomz-560m	0.564	0.362	0.066	0.514	0.713	0.748	0.629	0.826	0.736	0.876	0.667	0.667	0.843	0.391	0.747	0.667	0.918
leo-hessianai-7b	0.678	0.528	0.672	0.778	0.841	0.793	0.737	0.906	0.839	0.951	0.691	0.757	0.898	0.806	0.868	0.877	0.956

Table 6: Individual results for classification tasks per model and task.

	NER					other			
	micro F1 News	micro F1 EuroParl	micro F1 BioFID	micro F1 Wiki & News	micro F1 Legal	micro F1 UP	micro F1 UP	micro F1 MASSIVE	micro F1 GermEval Opinions
gbert-base	0.657	0.633	0.637	0.841	0.925	0.939	0.906	0.905	0.489
gbert-large	0.688	0.632	0.646	0.861	0.942	0.939	0.912	0.91	0.462
gottbert	0.546	0.588	0.603	0.833	0.923	0.938	0.904	0.889	0.467
bert-base-german-cased	0.628	0.588	0.593	0.819	0.931	0.935	0.899	0.882	0.463
mbart-large-50	0.679	0.651	0.614	0.827	0.936	0.937	0.905	0.914	0.442
mt0-small	0.115	0.078	0.317	0.699	0.692	0.904	0.814	0.807	0.196
mt5-small	0.269	0.263	0.352	0.688	0.703	0.907	0.824	0.836	0.194
german-gpt2	0.518	0.524	0.477	0.735	0.841	0.909	0.847	0.859	0.370
bloomz-560m	0.203	ShapeError	ShapeError	0.566	ShapeError	0.853	0.762	0.843	ShapeError
leo-hessianai-7b	0.619	0.744	0.575	0.773	0.952	0.897	0.854	0.914	Running

Table 7: Individual results for sequence tagging tasks per model and task.

	mean token F1	
	MLQA	GermanQuAD
gbert-base	0.843	0.783
gbert-large	0.847	0.805
gottbert	0.736	0.787
bert-base-german-cased	0.836	0.769
mbart-large-50	0.849	0.808
mt0-small	0.725	0.675
mt5-small	0.836	0.741
german-gpt2	0.851	0.778
bloomz-560m	0.847	0.721
leo-hessianai-7b	unsupported	

Table 8: Individual results for extractive QA tasks per model and task.