Encoder vs Decoder: Comparative Analysis of Encoder and Decoder Language Models on Multilingual NLU Tasks

Dan Saattrup Nielsen
The Alexandra Institute
dan.nielsen@alexandra.dk

Kenneth Enevoldsen
University of Aarhus
kenneth.enevoldsen@cas.au.dk

Peter Schneider-Kamp
University of Southern Denmark
petersk@imada.sdu.dk

Abstract

This paper explores the performance of encoder and decoder language models on multilingual Natural Language Understanding (NLU) tasks, with a broad focus on Germanic languages. Building upon the ScandEval benchmark, initially restricted to evaluating encoder models, we extend the evaluation framework to include decoder models. We introduce a method for evaluating decoder models on NLU tasks and apply it to the languages Danish, Swedish, Norwegian, Icelandic, Faroese, German, Dutch, and English. Through a series of experiments and analyses, we also address research questions regarding the comparative performance of encoder and decoder models, the impact of NLU task types, and the variation across language resources. Our findings reveal that encoder models can achieve significantly better NLU performance than decoder models despite having orders of magnitude fewer parameters. Additionally, we investigate the correlation between decoders and task performance via a UMAP analysis, shedding light on the unique capabilities of decoder and encoder models. This study contributes to a deeper understanding of language model paradigms in NLU tasks and provides valuable insights for model selection and evaluation in multilingual settings.

1 Introduction

Language models have attained remarkable Natural Language Understanding (NLU) performance, both with encoder-based architectures like BERT (Devlin et al., 2018) and and decoder-based architectures like GPT-3 (Brown, Tom and Mann, Benjamin and Ryder, Nick and Subbiah, Melanie and Kaplan, Jared D and Dhariwal, Prafulla and Neelakantan, Arvind and Shyam, Pranav and Sastry, Girish and Askell, Amanda and others, 2020). The encoder models have excelled in capturing contextual information for downstream tasks through masked language modeling objectives, while decoder models have shown strong generative capabilities by autoregressively predicting subsequent tokens based on preceding context.

Since the "ChatGPT boom" in 2023, the research community has been increasingly focused on decoder models (Zhao et al., 2023) for both Natural Language Generation (NLG) and NLU tasks. However, few studies have systematically compared the performance of encoder and decoder models across a diverse range of NLU tasks, and the studies that exist have primarily focused on English. This leaves a gap in our understanding of how the two language model paradigms perform in multilingual settings across different languages and tasks.

Nielsen (2023) introduced the ScandEval benchmark and evaluated encoder language models on four different natural language understanding tasks in Danish, Swedish, Norwegian (Bokmål and Nynorsk), Icelandic and Faroese. In this paper, we bridge this gap by extending the ScandEval benchmark to encompass the evaluation of decoder models on multilingual NLU tasks, as well as expanding the language resources to include German, Dutch and English.

Our **main research question** is

> *Which language model paradigm is better suited for NLU?*

We will answer this question with the languages Danish, Swedish, Norwegian, Icelandic, Faroese, German, Dutch and English as a case study. To concretise our main question, we will study the following research questions in this paper:

(Q1) Can state-of-the-art finetuned encoder models

Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025), pages 561–572
March 3-4, 2025

achieve significantly better NLU performance than state-of-the-art decoder models?

(Q2) Does the answer to (Q1) depend on the type of NLU task?

(Q3) Does the answer to (Q1) vary along the language resource spectrum, from low- to high-resource?

Our main contributions of this paper are the following:

1. We extend the ScandEval benchmarking framework with few-shot evaluation of decoder models and release this extension open-source.

2. We extend the languages supported by the ScandEval benchmarking framework by German, Dutch and English. Together with Danish, Swedish, Norwegian, Icelandic and Faroese, ScandEval now provides coverage of all Germanic languages except Afrikaans and the Frisian languages.

3. We evaluate an extensive suite of both encoder and decoder models on NLU tasks in all of the supported languages and publish these on public leaderboards.

4. We give a positive answer to (Q1), showing that encoder models achieve significantly better NLU performance than encoder models in several languages. This depends on the language in question however, giving a partially positive answer to (Q3).

5. We also show that the decoder models are heavily biased towards the question answering task (even models that are not instruction tuned), and a UMAP analysis shows that the performance distribution of decoder models follow a different "path" than encoder models, from the worst to best performing models. This gives a positive answer to (Q2).

2 Related Work

2.1 Comparing Encoder and Decoder Models

There has been a number of studies in recent years comparing encoder models to decoder models. Zhong et al. (2023) compared GPT-3.5-turbo (January 2023 version) to (finetuned versions of) the base and large versions of BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019) on the English GLUE benchmark (Wang et al., 2018). They find that GPT-3.5-turbo is on average on par with the base-sized encoder models, but falls short of the large-sized ones. They also note that despite being on par with the base-sized models, there is a big discrepancy between the models on individual tasks, with GPT-3.5-turbo for instance being better on the inference tasks while being worse on the paraphrase tasks. We note however that they only evaluate the decoder model in a zero-shot setting, and furthermore they only evaluate the models on 25 samples for each class in the development split, leading to a potential lack of robustness in their evaluation.

Wang et al. (2023) compares GPT-3.5-turbo (January 2023 version) to a finetuned version of the base-sized BERT model on 18 English benchmark datasets related to sentiment analysis. Like Zhong et al. (2023), they find that the zero-shot performance of GPT-3.5-turbo is on par with the base-sized BERT model, and that the few-shot performance of GPT-3.5-turbo (with 27 few-shot examples) is slightly better than BERT, on average. Their test sets contained, on average, 538 samples, which is a significant improvement over Zhong et al. (2023). However, the narrow focus on the evaluation tasks as well as only benchmarking a single encoder and decoder model makes it hard to generalise the results to other tasks and models.

Kocoń et al. (2023) built a benchmark suite of 25 tasks, where 21 of these tasks are classification tasks (binary, multi-class and multi-label), 3 being question answering tasks and the last one being a token classification task. Two of the classification tasks are in Polish and the rest in English. They compare the zero-shot and few-shot performance of GPT-3.5-turbo (January 2023 version) to the state-of-the-art encoder performance on each task. GPT-3.5-turbo is generally found to be worse than state-of-the-art encoder models. They also evaluate GPT-4 on five of the tasks (inference, question-answering and emotion datasets), and only find GPT-4 to be marginally better than GPT-3.5-turbo, still far off the encoder models.

Qiu and Jin (2024) compare GPT-3.5-turbo (January 2023 version) to a finetuned version of the base-sized BERT model on three manually curated English multi-class classification datasets with 19, 12 and 7 test samples, respectively, where they find that the BERT model performs marginally better

than GPT-3.5-turbo in a few-shot setting (and that the zero-shot performance is significantly worse). The tiny test sets make it hard to generalise the results, however.

2.2 Benchmarks of Generative Language Models

In recent times, several benchmarks of generative language models have been introduced. The major ones are EleutherAI's Evaluation Harness (Gao et al., 2023), Hugging Face's Open LLM Leaderboard (hug) which uses the Evaluation Harness as evaluation engine, and Stanford University's HELM (Bommasani et al., 2023). These are firstly all English-only benchmarks, making it hard to generalise the results to other languages, and they only include point estimates of the dataset performance, and thus do not necessarily provide a robust assessment of the models. Further, these benchmarks are exclusively for decoder models, and thus does not provide a way to compare encoders with decoders.

There has been several language-specific benchmarks introduced as well. NorBench (Samuel et al., 2023) is a collection of Norwegian evaluation datasets moreso than a dedicated evaluation framework. Further, several datasets in this collection (NorQuAD, NoReC and NorNE) are already part of ScandEval. SuperLim (Berdičevskis et al., 2023) falls into the same category for Swedish. DUMB (de Vries et al., 2023) is a Dutch benchmarking framework, which is only focused on encoder models. Danoliterate (Holm, 2024) is a Danish benchmarking framework which is solely focused on evaluating decoder models, and whose datasets largely overlap with the Danish datasets in ScandEval, albeit with a different evaluation methodology. Aside from language modelling performance, the Danoliterate benchmark also measures calibration, efficiency, toxicity and fairness. While the development of language-specific benchmarks is important, it leads to too little overview of trends across benchmarks and languages and incentivises model development focused on monolingual models ignoring a potential broader appeal. ScandEval provides a unified and robust approach for comparison across model categories and Germanic languages.

Benchmarking is not the only way to evaluate language models. A new "arena approach" has been popularised by the LMSYS Arena (Chiang et al.), where users can submit a prompt and get two responses from two anonymised models at random, and have to evaluate the responses. The Arena is predominantly used for English, but also currently supports six other languages. This approach is a promising way to evaluate language models, but we fear that it is not as suitable for low-resource languages due to the need of many volunteers to evaluate the responses.

Lastly, the Scandinavian Embedding Benchmark (Enevoldsen et al., 2024b) complements ScandEval and focuses on evaluating embedding models on a wide range of tasks in the Scandinavian languages.

3 Datasets

In this section we present the datasets that we are evaluating the models on, all of which are now included in the ScandEval framework. We should note that these datasets either (a) already existed prior to this publication or (b) are small extensions of existing datasets. An overview of all the datasets can be found in Table 1.

3.1 Named Entity Recognition

For Norwegian, Swedish and Icelandic we use the NorNE (Jørgensen et al., 2020), SUC 3.0 (Gustafson-Capková and Hartmann, 2006), MIM-GOLD-NER (Ingólfsdóttir et al., 2020) datasets, which were already included in the ScandEval framework. For Faroese we replace the previous WikiANN-fo dataset (Rahimi et al., 2019) with the new human annotated FoNE dataset (Snæbjarnarson et al., 2023). We also replace the previous DaNE dataset (Hvingelby et al., 2020) with the new DANSK dataset (Enevoldsen et al., 2024a) covering a wider variety of domains. For German, Dutch and English we add the established NER datasets GermEval (Benikova et al., 2014), the Dutch part of CoNLL-2002 (Sang, 2002), and the English CoNLL-2003 (Sang and De Meulder, 2003).

3.2 Sentiment Classification

We re-use the sentiment classification datasets AngryTweets (Pauli et al., 2021), NoReC (Velldal et al., 2018) and SweReC (Svensson, 2017), for Danish, Norwegian and Swedish, respectively. For German, Dutch and English we add the existing datasets SB10k (Cieliebak et al., 2017), Dutch Social (Gupta, 2022) and SST5 (Socher et al., 2013). We convert SST5 to the standardised trinary (negative, neutral, positive) format by converting the

Dataset	Language	#Train	#Val	#Test	#Shots
NER					
DANSK (Enevoldsen et al., 2024a)	Danish	1,024	256	1,024	8
SUC 3.0 (Gustafson-Capková and Hartmann, 2006)	Swedish	1,024	256	2,048	8
NorNE-nb (Jørgensen et al., 2020)	Norwegian Bokmål	1,024	256	2,048	8
NorNE-nn (Jørgensen et al., 2020)	Norwegian Nynorsk	1,024	256	2,048	8
MIM-GOLD-NER (Ingólfsdóttir et al., 2020)	Icelandic	1,024	256	2,048	8
FoNE (Snæbjarnarson et al., 2023)	Faroese	1,024	256	2,048	8
GermEval (Benikova et al., 2014)	German	1,024	256	1,024	8
CoNLL-nl (Sang, 2002)	Dutch	1,024	256	1,024	8
CoNLL-en (Sang and De Meulder, 2003)	English	1,024	256	2,048	8
Sentiment Classification					
Angry Tweets (Pauli et al., 2021)	Danish	1,024	256	2,048	12
SweReC (Svensson, 2017)	Swedish	1,024	256	2,048	12
NoReC (Velldal et al., 2018)	Norwegian	1,024	256	2,048	12
SB10k (Cieliebak et al., 2017)	German	1,024	256	1,024	12
Dutch Social (Gupta, 2022)	Dutch	1,024	256	1,024	12
SST5 (Socher et al., 2013)	English	1,024	256	2,048	12
Linguistic Acceptability					
ScaLA-da (Nielsen, 2023)	Danish	1,024	256	2,048	12
ScaLA-sv (Nielsen, 2023)	Swedish	1,024	256	2,048	12
ScaLA-nb (Nielsen, 2023)	Norwegian Bokmål	1,024	256	2,048	12
ScaLA-nn (Nielsen, 2023)	Norwegian Nynorsk	1,024	256	2,048	12
ScaLA-is (Nielsen, 2023)	Icelandic	1,024	256	2,048	12
ScaLA-fo (Nielsen, 2023)	Faroese	1,024	256	1,024	12
ScaLA-de (Nielsen, 2023)	German	1,024	256	2,048	12
ScaLA-nl (Nielsen, 2023)	Dutch	1,024	256	2,048	12
ScaLA-en (Nielsen, 2023)	English	1,024	256	2,048	12
Question Answering					
ScandiQA-da (Nielsen, 2023)	Danish	1,024	256	2,048	4
ScandiQA-sv (Nielsen, 2023)	Swedish	1,024	256	2,048	4
NorQuAD (Ivanova et al., 2023)	Norwegian Bokmål	1,024	256	2,048	2
NQiI (Snæbjarnarson and Einarsson, 2022)	Icelandic	1,024	256	1,024	4
GermanQuAD (Möller et al., 2021)	German	1,024	256	2,048	4
SQuAD-nl (Havinga, 2023)	Dutch	1,024	256	2,048	4
SQuAD (Rajpurkar et al., 2016)	English	1,024	256	2,048	4

Table 1: All the datasets used in the NLU evaluation. Note that these have been re-sized and do not represent the sizes of the original dataset.

"very negative" and "very positive" labels to "negative" and "positive", respectively.

3.3 Linguistic Acceptability

For linguistic acceptability we re-use the ScaLA datasets for all the Scandinavian languages, and extend the ScaLA datasets by applying the ScaLA method from Nielsen (2023) to German, Dutch and English by using the German (McDonald et al., 2013), Dutch (van der Beek et al., 2002) and English (Zeldes, 2017) dependency treebanks.

3.4 Extractive Question Answering

Here we use the ScandiQA dataset (Nielsen, 2023) for Danish and Swedish, but replace the manually translated Norwegian ScandiQA dataset with the new curated NorQuAD dataset (Ivanova et al., 2023). We further add the new Natural Questions in Icelandic dataset (Snæbjarnarson and Einarsson, 2022) for Icelandic. For German and English we add the existing extractive question-answering datasets GermanQuAD (Möller et al., 2021) and SQuAD (Rajpurkar et al., 2016), respectively. For Dutch we add the machine translated version of SQuAD to Dutch (Havinga, 2023).

4 Methodology

4.1 Formulating NLU Tasks as Generative Tasks

In this section we describe how we rephrase the NLU tasks as text-to-text tasks, which makes it possible to evaluate generative models on the tasks. We formulate all the tasks as few-shot tasks, generally formatted as follows:

> [prefix prompt]
>
> [document prefix]: [document]
> [label prefix]: [label]
>
> (...)
>
> [document prefix]: [document]
> [label prefix]:

We found that the separation of the few-shot examples with double newlines makes it easier to know when to stop the generation - for the same reason, we ensure that there are no double newlines in any of the documents. See the prompts used for the English datasets in Table 2; a full table of the prompts used for all the tasks in all the languages can be found in (Nielsen et al., 2024).

For the sentiment classification task, we simply have the models generate translations of the three labels (positive, negative and neutral). For the linguistic acceptability task, also a text classification task, we use the translations of "yes" and "no" as the two labels, corresponding to whether the document is grammatically correct or not. For the extractive question answering task, we have the model output the answer directly. For this task we found that changing the label prefix from "Answer" to "Answer in max 3 words" resulted in a drastic improvement, due to many of the answers of instruction tuned models starting with unnecessary text akin to "The answer is". Lastly, for the named entity recognition task, we require the output to be a JSON dictionary (ISO/IEC 21778:2017), with keys being the translated named entity tags, and values being lists of named entities of that category. To ensure that we are not biasing the evaluation toward models knowing the JSON format, we employ structured generation using the `outlines` package (Louf, 2023), which modifies the logits outputted by the model to ensure that the output is always a valid JSON dictionary in the aforementioned format.

4.2 Evaluation Methodology

We keep the evaluation methodology for the generative models to be as close to the methodology for encoder models in Nielsen (2023). We think of the few-shot examples as analogous to training examples for encoder models. Indeed, as von Oswald et al. (2023) shows, this assumption is theoretically grounded. We thus evaluate the models 10 times, where on each iteration we sample few-shot examples at random from the training split, and we evaluate the model on a bootstrapped version of the test split. As with the encoder models, this allows us to take into account more noise in evaluation process, resulting in more robust evaluation scores.

The number of few-shot examples for each dataset was determined on a heuristic basis, where we wanted to include as many examples as possible, while making sure that the token count was sufficiently low to not bias the evaluation towards models with a longer context length. All the NER, sentiment classification and linguistic acceptability datasets have prompt sizes around 1,000 tokens with the Mistral-7B-v0.1 tokeniser (Jiang et al., 2023), with the question answering datasets having around 2,000 tokens. This is also the reason for

Task	Prefix Prompt	Example Prompt
Named entity recognition	Below are sentences and JSON dictionaries with the named entities that occur in the given sentence.	Sentence: [text] Named entities: [label]
Sentiment classification	The following are tweets are their sentiment, which can be 'positive', 'neutral' or 'negative'.	Tweet: [text] Sentiment: [label]
Linguistic acceptability	The following are sentences and whether they are grammatically correct.	Sentence: [text] Grammatically correct: [label]
Question answering	The following are texts with accompanying questions and answers.	Text: [text] Question: [question] Answer in max 3 words: [label]

Table 2: The English prompt templates used for the datasets. See all the prompt templates in (Nielsen et al., 2024).

the discrepancy with the NorQuAD dataset, as the samples are much longer than the other question answering datasets.

4.3 Score Aggregation Method

From the raw scores of the 10 evaluations per dataset, we need to aggregate the model scores into a single score. We want an aggregation method that satisfies the following criteria:

1. **Task Fairness:** Each task should be weighted equally.
2. **Comparison:** If we evaluate models in multiple languages, then it should be possible to meaningfully compare the language scores of these models with each other.
3. **Robustness:** If two models do not have a significantly different score on a dataset, then the aggregated score should reflect this.
4. **Magnitude Preservation:** The magnitude of the difference between the dataset score of two models should be reflected in the aggregated score.
5. **Minimal Change:** Adding a new model should minimally affect the aggregated scores of the other models.

Before we introduce our chosen aggregation method, we will briefly discuss some common aggregation methods and how they do not satisfy the criteria.

The **mean score** is the most common aggregation method, which would simply be the mean of the 10 scores for each dataset, and then the mean of the dataset scores for each task. This method does not satisfy the Task Fairness criterion, as it does not take into account that metrics have different ranges and variances. The Comparison criterion is also not satisfied, as datasets vary from language to language, with some datasets being more difficult than others. It *does*, however, satisfy the Robustness, Magnitude Preservation and Minimal Change criteria.

The **mean rank** is another common aggregation method, where we compute the rank of each model on each dataset, and then take the mean of the ranks. This method satisfies the Task Fairness criterion, as it re-casts the scores into a common comparable framework, which therefore weights each task equally. For the same reason, it also satisfies the Comparison criterion (it is important here that we evaluate all the models on all the languages for this to be satisfied). It does not satisfy the Robustness and Magnitude Preservation criteria, by definition of rank. It partially satisfies the Minimal Change criterion, since it only affects the scores of the models which are worse than the new model.

We thus see that the mean score and mean rank methods satisfy a disjoint set of the criteria, but that they together satisfy all the criteria. Based on this observation, we introduce the **mean rank score** method, defined as follows. For each dataset, we start by sorting the models by their mean score on the dataset. As with a rank, we assign the best model with rank score 1. For the next best model, we conduct a one-tailed Welch's t-test to see if the next best model is significantly worse than the first model ($p < 0.05$). If so, we compute the absolute difference between the mean score of the two models, and divide that by the standard deviation of all the mean scores of the models on the dataset.

We then add this to the rank score of the first model. We continue this process for all the models to get the rank scores for the dataset, and to

compute the overall score for the model, we take the mean of the rank scores for the datasets. An overview of this aggregation method can be found in (Nielsen et al., 2024). We note that the mean rank score has an intuitive interpretation: it is the average number of standard deviations from the best scoring model (+1).

This metric satisfies Task Fairness since we normalise all the scores by dividing by the standard deviation of the dataset scores. The Robustness criterion is satisfied due to our use of a one-tailed Welch's t-test. The Magnitude Preservation criterion is also satisfied, as the magnitude of the difference between the dataset score of two models is reflected in the rank score. It also satisfies Comparison, as we compare the models on a common scale (same argument as the mean rank method). Finally, the Minimal Change criterion is partially satisfied, as adding new models only minimally changes the score of existing models. Concretely, adding new scores will affect the standard deviation normalising factor (this effect tends to zero as the number of models grows, however), and if the model beats all the other models then all the scores will be affected, due to the relative nature of the metric.

5 Analysis

5.1 Comparative Performance Analysis on High- and Low-resource Languages

Excerpts of the English, Danish and Icelandic leaderboards can be found in Table 3, Table 4 and Table 5, respectively. We found that these three represent three main categories of languages with respect to the open-closed source divide. Similar excerpts for the remaining languages (Swedish, Norwegian, Faroese, German and Dutch) can be found in (Nielsen et al., 2024). The full leaderboards for all the languages can be found at `https://scandeval.com`.

From the English results we see that the state-of-the-art decoder model GPT-4-0613 (Achiam et al., 2023) is still outperformed by the DeBERTa-v3-large and DeBERTa-v3-base models (He et al., 2020) as well as the ELECTRA-base model (Clark et al., 2020). Here GPT-4-0613 is, on average, 0.44 standard deviations worse than the best model. The same pattern is seen for Norwegian, Dutch, German and Faroese; see (Nielsen et al., 2024) for the corresponding leaderboard excerpts.

In contrast, on the Danish leaderboard, the top-3 models are all decoder models, with GPT-4-0613 and GPT-4-1106-preview (OpenAI, 2023b) in the lead, followed by the closed-source DanskGPT-Chat-Llama3-70B model from Syv.AI[1], being a continuation of the Llama-3-70B model (AI@Meta, 2024). The GPT-4-0613 model is, on average, 0.24 standard deviations from the best model. Similar results were found with Swedish; see (Nielsen et al., 2024) for the corresponding leaderboard excerpt.

Model ID	Decoder	Score (↓)
microsoft/deberta-v3-large	✗	1.09
microsoft/deberta-v3-base	✗	1.29
google/electra-base-discriminator	✗	1.39
gpt-4-0613	✓	1.44
FacebookAI/roberta-large	✗	1.46
FacebookAI/roberta-base	✗	1.51
microsoft/mdeberta-v3-base	✗	1.53
gpt-4-1106-preview	✓	1.54
gpt-4o-2024-05-13	✓	1.64
AI-Sweden-Models/roberta-large-1160k	✗	1.64
gpt-3.5-turbo-0613	✓	1.78
mistralai/Mistral-7B-v0.1	✓	1.91

Table 3: Excerpt of the English ScandEval leaderboard.

Lastly, for Icelandic, we see that the encoders and decoders are tied in performance, with the mDeBERTa-v3-base model and the GPT-4-1106-preview model being the top models. The GPT-4-1106-preview model is, on average, 0.24 standard deviations from the best model. We note that Icelandic is the *only* language where the switch from GPT-4 (gpt-4-0613) to GPT-4-turbo (gpt-4-1106-preview) resulted in a significant *increase* in performance. We speculate that this is due to the collaboration between OpenAI and Iceland (OpenAI, 2023a).

We can thus give an affirmative answer to research question (Q1), showing that encoder models *can* achieve significantly better NLU performance than decoder models, even though they have an order of magnitude fewer model parameters. For (Q3), we see that this varies between languages, but without being correlated to the language resource spectrum.

5.2 Task Analysis

In this section we analyse our research question (Q2), asking whether the NLU performance results from the previous section is dependent on the type of NLU task.

Firstly, we analyse whether the score distribution across the four NLU tasks is different for the encoder and decoder models. This is done by applying a UMAP (McInnes et al., 2018) to the results of a given leaderboard, which is a dimensionality

[1] `https://www.syv.ai/`

Model ID	Decoder	Score (↓)
gpt-4-0613	✓	1.24
gpt-4-1106-preview	✓	1.25
syvai/danskgpt-chat-llama3-70b	✓	1.29
AI-Sweden-Models/roberta-large-1160k	✗	1.39
danish-foundation-models/encoder-large-v1	✗	1.40
meta-llama/Meta-Llama-3-70B	✓	1.40
AI-Sweden-Models/Llama-3-8B-instruct	✓	1.44
gpt-4o-2024-05-13	✓	1.46
ltg/norbert3-large	✗	1.50
NbAiLab/nb-bert-large	✗	1.54
vesteinn/DanskBERT	✗	1.56
google/rembert	✗	1.61
intfloat/multilingual-e5-large	✗	1.62
gpt-3.5-turbo-0613	✓	1.68
FacebookAI/xlm-roberta-large	✗	1.71

Table 4: Excerpt of the Danish ScandEval leaderboard.

Model ID	Decoder	Score (↓)
microsoft/mdeberta-v3-base	✗	1.33
gpt-4-1106-preview	✓	1.34
gpt-4o-2024-05-13	✓	1.43
vesteinn/ScandiBERT-no-faroese	✗	1.48
google/rembert	✗	1.57
vesteinn/XLMR-ENIS	✗	1.59
gpt-4-0613	✓	1.79
mideind/IceBERT-large	✗	1.85
vesteinn/FoBERT	✗	1.87
meta-llama/Meta-Llama-3-70B	✓	2.03
FacebookAI/xlm-roberta-large	✗	2.34
gpt-3.5-turbo-0613	✓	2.51
mistralai/Mistral-7B-v0.1	✓	2.96

Table 5: Excerpt of the Icelandic ScandEval leaderboard.

reduction method that both takes into account the global and local structure of the underlying data - it can thus be viewed as a middle ground between a principal component analysis (Pearson, 1901) and a t-distributed stochastic neighbour embedding (Hinton and Roweis, 2002). The resulting reduction thus contains a single two-dimensional representation of each model. UMAP plots for the English, Danish, Swedish, Norwegian, German and Dutch leaderboards can be found in Figure 1, where we also mark the mean rank score for each model, as well as whether the model is generative.

We see that the worst and best performing models have similar distributions, irrespective of whether they are generative or not. However, we also note that the rest of encoder and decoder models follow different "paths" in the UMAP space, leading to our hypothesis that the different architectures have different task preferences.

In Figure 2 we show the correlation between a model being generative and its performance on the four NLU tasks. We see that being generative is a strong predictor for good question answering performance, as well as poor named entity recognition and linguistic acceptability performance. The correlation is weaker for sentiment classification and varies across languages. We also see that these findings seem to generalise across languages, both high- and low-resource. The large question answering performance persists for non-instruction-tuned decoder models (see the leaderboards at `https://scandeval.com`), showing a likely side-effect of the pre-training algorithm or the architecture of decoder models making them better at this task. We also note that generative models perform substantially better at the English sentiment classification dataset SST5 compared to the other sentiment classification datasets - we will return to this in the discussion.

6 Discussion

Having a good mean rank score is not the only thing that matters when choosing a model for a given task. Model size, inference speed and whether the model has publicly available weights are all important factors to consider. For this reason, we also include these metadata in the leaderboard, and we encourage the community to consider these factors when choosing a model for a given task.

Some of the datasets in the benchmark are translations of American datasets, which we acknowledge is not ideal and encourage the development of gold-standard replacements of these. This concerns the Dutch question answering dataset, which is machine translated, as well as the Danish and Swedish question answering datasets, where the questions and answers have been manually translated. Manual translations are typically better than machine translations, but it nevertheless means that the content is biased towards questions pertinent to the American context. Some datasets are furthermore missing. This concerns Icelandic and Faroese sentiment analysis, as well as Faroese question answering. Efforts are currently underway to remedy this.

Lastly, we note that the English sentiment classification dataset SST5 is the only dataset where generative models perform substantially better than encoder models. We speculate that this is either due to the dataset simply being significantly easier than the others, or that the test data has leaked into the pretraining datasets of the generative models. The dataset is part of the FLAN collection (Wei et al.), which is for instance included in the Dolma dataset (Soldaini et al., 2024), which is used to pretrain the OLMo model (Groeneveld et al., 2024), being one of the generative models that is performing very

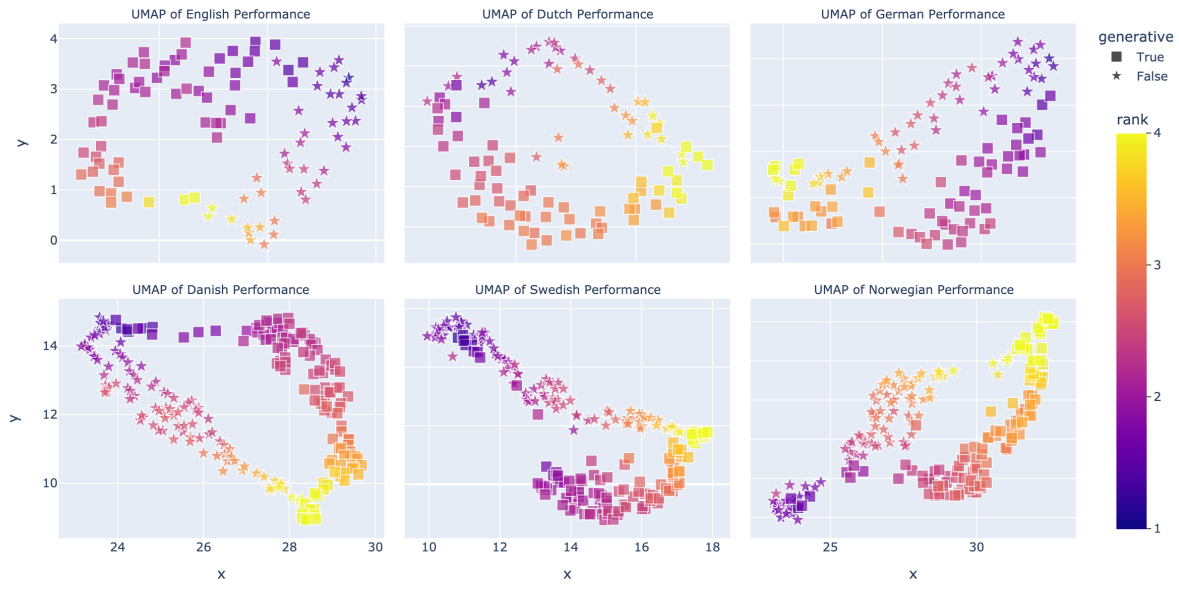

Figure 1: UMAP plots of the models on the ScandEval leaderboards.

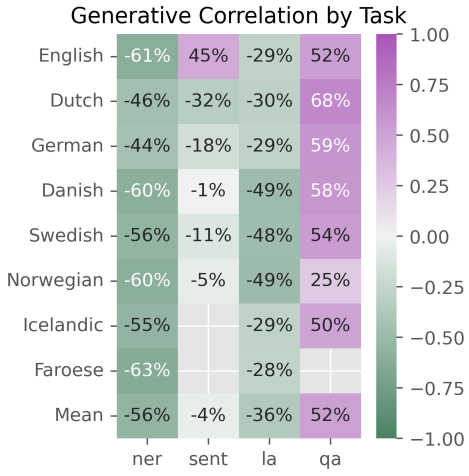

Figure 2: The correlation between a model being generative and its performance on the NLU tasks.

well on this dataset. Leakage is therefore possible, and we encourage the community to investigate this further.

7 Conclusion

We have extended the ScandEval benchmark to include the evaluation of decoder models, as well as including three new languages: German, Dutch and English. From the analysis of the corresponding results we found that encoder models can achieve significantly better NLU performance than decoder models despite having orders of magnitude fewer parameters, but that this varies between languages. We have also shown that being generative is strongly correlated with both good question answering performance and poor performance for named entity recognition and linguistic acceptability. Our analysis showed that the "path" from the worst to the best-performing models in the UMAP space is different for encoder and decoder models, indicating an architecture-specific task-preference.

Ethics Statement

We have made efforts towards making the evaluation as fair and unbiased as possible, both through our selection of the datasets in the benchmark as well as through our choice of aggregation method of the scores. However, we have not conducted extensive bias analyses on the individual datasets.

Acknowledgements

This work has received funding by the European Union's Horizon 2023 Research and Innovation Actions, as part of the Artificial Intelligence and Robotics programme, for the project "TrustLLM" (grant agreement number 101135671). Furthermore, this work reflects only the authors' view and the European Research Executive Agency (REA) is not responsible for any use that may be made of the information it contains.

References

Open LLM Leaderboard - a Hugging Face Space by open-llm-leaderboard — huggingface.co. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard. [Accessed 12-06-2024].

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI@Meta. 2024. Llama 3 model card.

L van der Beek, G Bouma, R Malouf, and G van Noord. 2002. The alpino dependency treebank. In *12th Meeting on Computational Linguistics in the Netherlands (CLIN)*, pages 8–22. Rodopi.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. Nosta-d named entity annotation for german: Guidelines and dataset. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2524–2531.

Aleksandrs Berdičevskis, Gerlof Bouma, Robin Kurtz, Felix Morger, Joey Öhman, Yvonne Adesam, Lars Borin, Dana Dannélls, Markus Forsberg, Tim Isbister, et al. 2023. Superlim: A swedish language understanding evaluation benchmark. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8137–8153.

Rishi Bommasani, Percy Liang, and Tony Lee. 2023. Holistic evaluation of language models. *Annals of the New York Academy of Sciences*, 1525(1):140–146.

Brown, Tom and Mann, Benjamin and Ryder, Nick and Subbiah, Melanie and Kaplan, Jared D and Dhariwal, Prafulla and Neelakantan, Arvind and Shyam, Pranav and Sastry, Girish and Askell, Amanda and others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*.

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.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Kenneth Enevoldsen, Emil Trenckner Jessen, and Rebekah Baglini. 2024a. Dansk and dacy 2.6. 0: Domain generalization of danish named entity recognition. *arXiv preprint arXiv:2402.18209*.

Kenneth Enevoldsen, Márton Kardos, Niklas Muennighoff, and Kristoffer Laigaard Nielbo. 2024b. The scandinavian embedding benchmarks: Comprehensive assessment of multilingual and monolingual text embedding. *arXiv preprint arXiv:2406.02396*.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2023. A framework for few-shot language model evaluation.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. Olmo: Accelerating the science of language models. *arXiv preprint arXiv:2402.00838*.

Aakash Gupta. 2022. dutchsocial · Datasets at Hugging Face — huggingface.co. https://huggingface.co/datasets/dutch_social. [revision: 8b7bc6230ebd78f04aa3661acb912f4567f21c76].

Sofia Gustafson-Capková and Britt Hartmann. 2006. Manual of the Stockholm Umeå corpus version 2.0. *Stockholm University*.

Yeb Havinga. 2023. squadv2dutch · Datasets at Hugging Face — huggingface.co. [revision: af494fe1b62762178d37c0b71b4a7160f0534f1a].

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Geoffrey E Hinton and Sam Roweis. 2002. Stochastic neighbor embedding. *Advances in neural information processing systems*, 15.

Søren Vejlgaard Holm. 2024. Are gllms danoliterate? benchmarking generative nlp in danish.

Rasmus Hvingelby, Amalie Brogaard Pauli, Maria Barrett, Christina Rosted, Lasse Malm Lidegaard, and Anders Søgaard. 2020. Dane: A named entity resource for danish. In *Proceedings of the 12th language resources and evaluation conference*, pages 4597–4604.

Svanhvít L Ingólfsdóttir, Ásmundur A Gudjónsson, and Hrafn Loftsson. 2020. Named Entity Recognition for Icelandic: Annotated Corpus and Models. In *International Conference on Statistical Language and Speech Processing*, pages 46–57. Springer.

ISO/IEC 21778:2017. 2017. The JSON data interchange syntax. Standard, International Organization for Standardization, Geneva, CH.

Sardana Ivanova, Fredrik Andreassen, Matias Jentoft, Sondre Wold, and Lilja Øvrelid. 2023. Norquad: Norwegian question answering dataset. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 159–168.

AQ Jiang, A Sablayrolles, A Mensch, C Bamford, DS Chaplot, D de las Casas, F Bressand, G Lengyel, G Lample, L Saulnier, et al. 2023. Mistral 7b (2023). *arXiv preprint arXiv:2310.06825*.

Fredrik Jørgensen, Tobias Aasmoe, Anne-Stine Ruud Husevåg, Lilja Øvrelid, and Erik Velldal. 2020. NorNE: Annotating named entities for Norwegian. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4547–4556, Marseille, France. European Language Resources Association.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. *Information Fusion*, 99:101861.

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

Rémi Louf. 2023. Outlines. `https://github.com/outlines-dev/outlines`.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, et al. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

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.

Dan Saattrup Nielsen. 2023. ScandEval: A Benchmark for Scandinavian Natural Language Processing. In *The 24rd Nordic Conference on Computational Linguistics*.

Dan Saattrup Nielsen, Kenneth Enevoldsen, and Peter Schneider-Kamp. 2024. Encoder vs decoder: Comparative analysis of encoder and decoder language models on multilingual nlu tasks. *arXiv preprint arXiv:2406.13469*.

OpenAI. 2023a. Government of Iceland: How Iceland is using GPT-4 to preserve its language. `https://openai.com/index/government-of-iceland`. [Accessed 12-06-2024].

OpenAI. 2023b. New models and developer products announced at DevDay. https://openai.com/blog/new-models-and-developer-products-announced-at-devday. [Accessed 12-06-2024].

Johannes von Oswald, Eyvind Niklasson, Maximilian Schlegel, Seijin Kobayashi, Nicolas Zucchet, Nino Scherrer, Nolan Miller, Mark Sandler, Max Vladymyrov, Razvan Pascanu, et al. 2023. Uncovering mesa-optimization algorithms in transformers. *arXiv preprint arXiv:2309.05858*.

Amalie Brogaard Pauli, Maria Barrett, Ophélie Lacroix, and Rasmus Hvingelby. 2021. Danlp: An open-source toolkit for danish natural language processing. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 460–466.

Karl Pearson. 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572.

Yunjian Qiu and Yan Jin. 2024. Chatgpt and finetuned bert: A comparative study for developing intelligent design support systems. *Intelligent Systems with Applications*, 21:200308.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for ner. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392.

David Samuel, Andrey Kutuzov, Samia Touileb, Erik Velldal, Lilja Øvrelid, Egil Rønningstad, Elina Sigdel, and Anna Palatkina. 2023. Norbench–a benchmark for norwegian language models. *arXiv preprint arXiv:2305.03880*.

Erik Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*.

Erik 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.

Vésteinn Snæbjarnarson and Hafsteinn Einarsson. 2022. Natural questions in icelandic. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4488–4496.

Vésteinn Snæbjarnarson, Annika Simonsen, Goran Glavaš, and Ivan Vulić. 2023. Transfer to a low-resource language via close relatives: The case study on faroese. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 728–737.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.

Kristoffer Svensson. 2017. Sentiment Analysis With Convolutional Neural Networks: Classifying sentiment in Swedish reviews. Bachelor's thesis.

Erik Velldal, Lilja Øvrelid, Eivind Alexander Bergem, Cathrine Stadsnes, Samia Touileb, and Fredrik Jørgensen. 2018. NoReC: The Norwegian review corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Wietse de Vries, Martijn Wieling, and Malvina Nissim. 2023. Dumb: A benchmark for smart evaluation of dutch models. *arXiv preprint arXiv:2305.13026*.

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.

Zengzhi Wang, Qiming Xie, Yi Feng, Zixiang Ding, Zinong Yang, and Rui Xia. 2023. Is chatgpt a good sentiment analyzer? a preliminary study. *arXiv preprint arXiv:2304.04339*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Amir Zeldes. 2017. The gum corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Qihuang Zhong, Liang Ding, Juhua Liu, Bo Du, and Dacheng Tao. 2023. Can chatgpt understand too? a comparative study on chatgpt and fine-tuned bert. *arXiv preprint arXiv:2302.10198*.