ScandEval: A Benchmark for Scandinavian Natural Language Processing

Anonymous Author(s)

Abstract

This paper introduces a Scandinavian benchmarking platform, `ScandEval`, which can benchmark any pretrained model on four different tasks in the Scandinavian languages. The datasets used in two of the tasks, linguistic acceptability and question answering, are new. We develop and release a Python package and command-line interface, `scandeval`, which can benchmark any model that has been uploaded to the Hugging Face Hub, with reproducible results. Using this package, we benchmark more than 80 Scandinavian or multilingual models and present the results of these in an interactive online leaderboard[1], as well as provide an analysis of the results. The analysis shows that there is substantial cross-lingual transfer among the the Mainland Scandinavian languages (Danish, Swedish and Norwegian), with limited cross-lingual transfer between the group of Mainland Scandinavian languages and the group of Insular Scandinavian languages (Icelandic and Faroese). The benchmarking results also show that the investment in language technology in Norway and Sweden has led to language models that outperform massively multilingual models such as XLM-RoBERTa and mDeBERTaV3. We release the source code for both the package[2] and leaderboard[3].

1 Introduction

In recent years, there has been a significant increase in the number of monolingual language models in the Scandinavian languages (Møllerhøj, 2020; Højmark-Bertelsen, 2021; Sarnikowski, 2021; Enevoldsen et al., 2021; Abdaoui et al., 2020; Kummervold et al., 2021; Malmsten et al., 2020; Snæbjarnarson, 2021), to the extent that it becomes difficult both for the practioner to choose the correct model for the task at hand, as well as for language researchers to ensure that their research efforts are indeed improving upon past work.

Aside from the increasing number of models, Sahlgren et al. (2021) also emphasises that a joint Scandinavian language model is probably a better strategy for the Scandinavian countries, considering the similarity of their languages and culture. Indeed, Faarlund (2019) even argue that the Danish, Norwegian and Swedish languages are so similar that they should be considered a single language.

The languages included in the term "Scandinavian" is debatable (oxf, 2021). Following the distinction between *Mainland Scandinavian* (Danish, Swedish and Norwegian) and *Insular Scandinavian* (Icelandic and Faroese) (Haugen, 1976; Faarlund, 2019), a distinction based on mutual intelligibility and syntactical structure, we focus in this work on the Mainland Scandinavian languages, while still allowing support for the Insular Scandinavian languages. Aside from being a standard distinction, our choice is also based on experiments on the cross-lingual transfer between these two groups, which we present in Section 3.4. We will here use the term "Scandinavian" to mean the collection of all five languages, and use the Mainland/Insular distinction when applicable.

To help facilitate progress in both improving upon the monolingual Scandinavian models as well as the multilingual, we present `ScandEval`, a benchmark of Scandinavian models, along with a Python package and Command-Line Interface (CLI), and an associated online leaderboard. This leaderboard contains the results of language models benchmarked on datasets within the Mainland

[1] (redacted to preserve anonymity)

[2] `https://anonymous.4open.science/r/ScandEval-A412`

[3] (redacted to preserve anonymity)

Scandinavian languages, as described in Section 4.

Recent studies (Khanuja et al., 2021; Pires et al., 2019; Lauscher et al., 2020) have shown that multilingual models can outperform monolingual models when the languages are sufficiently similar, and also that they are worse than the monolingual models when the languages are too dissimilar. This shows that the Scandinavian languages could have something to gain by creating "local multilingual" models, rather than using the massively multilingual models such as `XLM-RoBERTa` (Conneau et al., 2020). Based on this, we test the following hypotheses:

- **Hypothesis 1:** There is a substantial cross-lingual transfer within the Mainland Scandinavian languages.
- **Hypothesis 2:** There is no notable cross-lingual transfer between Mainland Scandinavian languages and Insular Scandinavian languages.

To the best of our knowledge, this is the first benchmarking tool for any of the Scandinavian languages, as well as the first online leaderboard containing scores from such a tool. Our contributions are the following:

1. We construct a new question answering dataset for the Mainland Scandinavian languages, dubbed `ScandiQA`.
2. We construct a new linguistic acceptability dataset for all the Scandinavian languages, dubbed `ScaLA`.
3. We develop a Python package and CLI, `scandeval`, which allows reproducible benchmarking of language models on Scandinavian language datasets.
4. We uniformise all the datasets used in the benchmark, to enable consistent evaluation across languages and datasets. These uniformised datasets are also available on the Hugging Face Hub[4].
5. We benchmark all the Scandinavian and a selection of the multilingual language models on the Hugging Face Hub[5] on the Mainland Scandinavian datasets in the benchmark, and present all the scores in an online leaderboard.

[4] `https://huggingface.co/ScandEval`
[5] `https://hf.co`

2 Related Work

There has been a number of (non-English) NLU benchmarks published in recent years (Wang et al., 2018; Sarlin et al., 2020; Rybak et al., 2020; Ham et al., 2020a; Shavrina et al., 2020; Wilie et al., 2020; Xiang et al., 2021; Koto et al., 2020; Safaya et al., 2022; Augustyniak et al.; Khashabi et al., 2020; Ham et al., 2020b; Xu et al., 2020; Dumitrescu et al., 2021), with whom we share the same goal of advancing the state of NLP in our respective languages. Within the Scandinavian languages specifically, the SuperLim benchmark (Adesam et al., 2020) is a Swedish NLU Benchmark featuring several difficult tasks. Most of the datasets in the SuperLim benchmark only contain a test set, however.

The `XGLUE` (Liang et al., 2020) dataset is another multilingual NLU benchmark. That dataset is different from `ScandEval` in that all the training data in `XGLUE` is in English, and that the majority of the test sets are not available in any of the Scandinavian languages.

Isbister and Sahlgren (2020) present a Swedish similarity benchmark, achieved through machine translating the `STS-B` dataset from the GLUE benchmark (Wang et al., 2018). Aside from only dealing with a single task and a single language, the quality of the dataset is worse than a gold-standard corpus as a result of the translation, as the authors also point out.

3 Methodology

This section describes our benchmarking methodology in detail, including both the setup of the datasets, the evaluation procedure and the scoring of the models. We also describe how we conduct the cross-lingual transfer experiments.

3.1 Uniform Benchmarking Datasets

As we are interested in comparing the performance of the models across languages, we ensure that all the datasets used in the benchmark are of the same format and the same size.

Notably, we have enforced fixed train/val/test splits of the datasets, of sizes 1,024/256/2,048. The choice of 1,024 training samples is a trade-off between having as few samples as possible, enabling faster benchmarking, and having enough samples to distinguish between the models. Another benefit of using a small training dataset is that it emphasises the importance of the pre-trained weights of

the models, rather than the fine-tuning process. The experiments leading to this choice are described in Section 6.3. Further, we wanted the test dataset to be as large as possible, to ensure more robust evaluations of the models, which led to the choice of 2,048 test samples based on the number of available samples in the smallest dataset. All of these datasets with their splits are available on the Hugging Face Hub[6].

3.2 Finetuning Setup

When finetuning, we enforce a learning rate of $2 \cdot 10^{-5}$ with 100 warmup steps, and a batch size of 32. If there is not enough GPU memory to finetune the model with this batch size, we halve it and double the amount of gradient accumulation, resulting in the same effective batch size. This is repeated until the batches can fit in memory.

We impose a linear learning rate schedule with intercept after $10,000$ training steps (with a training step consisting of 32 samples), and we adopt early stopping (Plaut et al., 1986) to stop the training procedure if the validation loss has not decreased for 90 training steps. We use the `AdamW` optimiser (Loshchilov and Hutter, 2018) with first momentum $\beta_1 = 0.9$ and second momentum $\beta_2 = 0.999$, and we optimise the cross-entropy loss throughout all tasks. Further, random seeds are fixed throughout, to ensure reproducibility.

The finetuning itself uses the `transformers` package (Wolf et al., 2020). For the named entity recognition task we use the `AutoModelForTokenClassification` class, which linearly projects the embedding from the language model encoder for each token into the entity logits for that token. For the classification tasks we use the `AutoModelForSequenceClassification` class, which linearly projects the embedding from the language model encoder to each document into the class logits for that document. Lastly, for the question answering task we use the `AutoModelForQuestionAnswering` class, which linearly projects the embedding from the language model encoder for each token, into the logits of the start and end positions of the answer for that token.

[6]See (redacted to preserve anonymity).

3.3 Bootstrapping Evaluation

For each model and dataset, we repeat the following procedure 10 times, which generates a score for each model and dataset combination: (a) Fix a random seed unique to the given iteration; (b) Finetune the model on the training set; (c) Evaluate the model on a bootstrapped version of the test set. The evaluation score is then the mean μ of these scores, along with a 95% confidence interval I_{10}, computed as

$$I_N := \mu \pm \frac{1.96}{N-1} \sum_{i=1}^{N} \texttt{score}_i. \qquad (1)$$

The combination of varying the random seeds as well as using bootstrapped test datasets ensures that we capture the noise coming from both the random initialisation of the added layers to the model as well as the noise in the test set, resulting in a more reliable confidence interval of the true mean for each model and dataset combination.

To aggregate these scores across all datasets, we firstly compute the *language-specific task scores* for each (model, language, task) triple, which is the mean of the scores of the model on the tasks of the language.[7] From these language-specific scores we next compute the *language score* for each (model, language) pair as the mean of the language-specific task scores across all the tasks. A final `ScandEval` score is computed as the average of the language scores, to emphasise the training of Scandinavian models rather than monolingual ones.

3.4 Cross-lingual Transfer

To test Hypothesis 1 and 2, stated in Section 1, we introduce a way to measure the "joint cross-lingual transfer" of a group of languages, by which we mean an aggregate of the cross-lingual transfer between any two languages in the group.

To do this, we first introduce a control group of non-Scandinavian languages: English, German, Dutch, Finnish, Russian and Arabic. By considering the combined set of languages in the control group and the Scandinavian languages, we aim to find the best split of these languages into two groups: a ScandEval benchmark group and a non-benchmark group. The "goodness" of a split

[7]This mean is only non-trivial for the Norwegian language for the named entity recognition task and the linguistic acceptability task, as these tasks are available in both Norwegian Bokmål and Norwegian Nynorsk.

is measured by benchmarking a "representative" model from each language on datasets in each of the benchmark languages and measuring the quality of the two-cluster clustering of these benchmarking values.

As an example, if Danish and Swedish constitute the benchmark group and the rest of the languages are in the non-benchmark group, we would benchmark the representative models from each language on the Danish and Swedish part of ScandEval, and then compute the F-statistic of the the clustering $\{\{\texttt{da}, \texttt{sv}\}, \{\texttt{no}, \texttt{is}, \texttt{fo}, \ldots\}\}$ with these benchmarking values, computed as the ratio of the between-group variance to the within-group variance.[8] We can then compare this F-statistic to the F-statistic of the clustering where the benchmark group consists of Danish and Norwegian, for instance.

As for picking a representative model for each language, we found pretrained language models of roughly the same size on the Hugging Face Hub, each of which has been pretrained on solely monolingual data. We note that no Faroese language model exists, so for that language we do not include any model but still include Faroese benchmarking datasets when Faroese is part of the benchmarking group. See the full list of models in the appendix.

We can then restate our first hypothesis as the mainland Scandinavian languages are all in the best-performing benchmark group, and our second hypothesis as the Insular Scandinavian languages are not in the best-performing benchmark group.

4 ScandEval Tasks

To properly evaluate the performance of a pretrained model, we ideally need to evaluate it on many diverse tasks. Unfortunately, the Scandinavian languages do not have many openly available datasets for many downstream tasks.

To address this, we construct two new Scandinavian datasets, `ScaLA` and `ScandiQA`, being *Linguistic Acceptability* (`LA`) and *Question Answering* (`QA`) datasets, respectively. These new tasks are supplemented by existing benchmarking datasets within *Named Entity Recognition* (`NER`) and *Sentiment Classification* (`SENT`). Aside from downstream performance of these tasks, we also benchmark the inference speed of each model. We describe all of these in more detail in the subsections below.

4.1 Named Entity Recognition

For the NER task we use the four classes used in CONLL (Tjong Kim Sang and De Meulder, 2003): `PER`, `LOC`, `ORG` and `MISC`, corresponding to proper names, locations, organisations and miscellaneous entities.

Since this is a token classification task and that the language models usually use different tokenisers, we have to ensure a uniform treatment of these as well. We tokenise the documents using the pretrained tokeniser associated to the model that we are benchmarking, and to ensure consistency of the evaluation we replace all but the first token in each word with the empty entity `O`, ensuring that the models have to predict the same amount of labels per document.

In terms of evaluation metrics, we use the micro-average F1-score, which is standard for NER. We also report a *no-misc score*, which is the micro-average F1-score after we replace the `MISC` class in the predictions and labels with the "empty label" `O`. This *no-misc score* is not used in any of the aggregated scores and is purely used for comparison purposes on the individual datasets.

For Danish we use the `DaNE` dataset (Hvingelby et al., 2020), being a NER tagged version of the Danish Dependency Treebank (Kromann and Lynge, 2004). `DaNE` is already in the CONLL format, so we perform no preprocessing on the data.

For Norwegian we use the Bokmål and Nynorsk `NorNE` datasets (Jørgensen et al., 2020), also being NER tagged versions of the Norwegian Dependency Treebanks (Øvrelid and Hohle, 2016). Aside from the `PER`, `LOC`, `ORG` and `MISC` tags, these also include `GPE_LOC`, `GPE_ORG`, `PROD`, `DRV` and `EVT` tags. We convert these to `LOC`, `ORG`, `MISC`, `MISC` and `MISC`, respectively.

Lastly, Swedish does not have a NER tagged version of the corresponding dependency treebank, but they instead have the `SUC3` dataset, a NER-enriched version of the *Stockholm-Umeå Corpus* (Gustafson-Capková and Hartmann, 2006). This dataset does not follow the CONLL format and is instead released in the XML format, with the `<name>` XML tags containing the NER tags for the words they span over [9]. This dataset contains

[8] Technically speaking, we get an F-statistic for each language in the benchmarking group, but we just use the mean of these F-statistics.

[9] The `<ne>` XML tags are also NER tags, but these have

the NER tags animal, event, inst, myth, other, person, place, product and work. These were converted to MISC, MISC, ORG, MISC, MISC, PER, LOC, MISC and MISC, respectively.

4.2 Sentiment Classification

We treat the sentiment classification task as a three-class classification task, with the classes positive, neutral and negative. Evaluation of the models is done using Matthew's Correlation Coefficient (Matthews, 1975) as the primary metric, as well as reporting the macro-average F1-score as a secondary metric.

For Danish we use the sentiment classification dataset AngryTweets (Pauli et al., 2021), which contains crowdsourced annotations of Danish tweets. To comply with Twitter's Terms of Use we have fully anonymised the tweets by replacing all user mentions with @USER and all links by [LINK], as well as shuffling the tweets.

For Norwegian we included the sentiment classification dataset NoReC (Norwegian Review Corpus) (Velldal et al., 2018), which are based on scraped reviews from Norwegian websites.

Lastly, for Swedish we use the sentiment classification dataset presented in Svensson (2017), which is based on reviews from the Swedish websites www.reco.se and se.trustpilot.com. In analogy with NoReC we dub this dataset the Swedish Review Corpus (SweReC).

4.3 Linguistic Acceptability

Based on the inclusion of the CoLA (Corpus of Linguistic Acceptability) dataset (Warstadt et al., 2019) in the GLUE benchmark (Wang et al., 2018), we construct new linguistic acceptability datasets for the Scandinavian languages. This task is often framed as a binary classification task, where the model is tasked with predicting whether a given sentence is grammatically correct or not.

We dub our new datasets Scandinavian Linguistic Acceptability (ScaLA), which we release for Danish, Norwegian Bokmål, Norwegian Nynorsk, Swedish, Icelandic and Faroese.[10] These are based on the Danish, Norwegian, Swedish, Icelandic and Faroese versions of the Universal Dependencies datasets (Kromann and Lynge, 2004; Øvrelid and Hohle, 2016; Nivre et al., 2006; Rögnvaldsson et al., 2012; Jónsdóttir and Ingason, 2020; Arnardóttir et al., 2020).

The documents in the datasets are assumed to be grammatically correct, and we create negative examples by *either* removing a word or swapping two consecutive words. In order to ensure that the resulting sentence is indeed correct, we enforce restrictions on the words that can be removed or swapped.

We do not remove adjectives, adverbs, punctuation, determiners or numbers, as the resulting sentence will still be grammatically correct in most cases. Further, we do not remove nouns or proper nouns if they have another noun or proper noun as neighbour, as again that usually does not make the sentence incorrect either. We are able to enforce these restrictions as we have gold-standard POS tokens available for these datasets. When swapping two neighbouring words, we require them to have different POS tags, and that they are neither punctuation nor symbols. If we swap the first word then we ensure that the swapped words have correct casing.[11]

4.4 Question Answering

We also construct new question answering datasets for the Mainland Scandinavian languages, as we are not aware of any existing datasets for these languages. We dub these datasets ScandiQA, which we release for each of the Mainland Scandinavian languages.[12]

These datasets are based on the MKQA dataset (Longpre et al., 2021), which is based on the *Natural Questions* (NQ) dataset (Kwiatkowski et al., 2019). The NQ dataset contains questions inputted to Google's search engine, associated with the HTML page of the search result. In many cases these questions have an answer associated with it (a so-called *short answer*) which appears in the HTML, and in some cases they also have the paragraph in which the short answer appears (a so-called *long answer*).

The MKQA dataset contains human translations of 10,000 questions and short answers into 26 languages, including Danish, Norwegian and Swedish. Aside from adding these translations, the MKQA dataset also corrects many mistakes in the original NQ dataset by including answers not present in the

been automatically produced by SpaCy (Honnibal et al., 2020) models and are thus not gold standard.

[10]These can be found at (redacted to preserve anonymity).

[11]The source code to create the dataset can be found in the src/scripts directory at https://anonymous.4open.science/r/ScandEval-A412.

[12]These can be found at (redacted to preserve anonymity).

original dataset, or by correcting the short answers chosen in the original dataset.

The main thing missing from the MKQA dataset is the context paragraph, which is what we add to the dataset as follows. For each MKQA sample, we first locate the corresponding sample in the NQ dataset. If that sample has a long answer then we use that as the initial (English) context. Otherwise, if neither the NQ dataset nor the MKQA dataset has an answer registered, then we use the paragraph in the HTML with the largest cosine similarity to the question, where we embed the documents using the Sentence Transformer (Reimers and Gurevych, 2019) model `all-mpnet-base-v2`.[13]

In the last case, where there is no long answer for the sample in NQ but there *is* an answer in MKQA, we want to identify the paragraph in the HTML containing the MKQA answer. Unfortunately, the MKQA answers do no appear verbatim in the HTML (for instance, all dates are standardised to the YYYY-MM-DD format). We thus start by forming a list of *answer candidates* based on the MKQA answer, which includes most of the ways dates and numerals are written in English. We then locate the paragraph containing any of the answer candidates and which has the largest cosine similarity to the question, where we embed the documents as described above.

The above procedure thus results in an English context paragraph containing the answer. We next translate this context paragraph to Danish and Swedish using the DeepL translation API[14]. As DeepL does not yet support Norwegian, we translate the context paragraph to Norwegian using the Google Translation API[15]. With the contexts translated, we next extract all the answer candidates for the translated context relevant to the given Mainland Scandinavian language, and change the answer to the answer candidate appearing in the translated context. If no answer candidate appears in the translated context then we discard the sample.

The MKQA dataset also contains samples with *no* answer, and we include these samples in the ScandiQA dataset as well. For these samples, we simply use the translated context paragraphs as described above. The final dataset contains 7,810 Danish samples, 7,798 Swedish samples and 7,813 Norwegian samples. We release this dataset separately[16], as well as build a ScandEval version of it with the same train/dev/test size as the other ScandEval datasets. In the ScandEval version we only include samples that contain an answer, as otherwise we found the 1,024 dataset size to be too small for this task.

We note that since this dataset is a translated version of a dataset originally written in English, it is not a perfect representation of the Mainland Scandinavian languages, as many of the questions and answers are concerned with topics specific to the USA. This might mean that pretrained multilingual models might have an advantage over monolingual models, but we leave this question for future work.

4.5 Inference Speed

Aside from the predictive performance of the models we also benchmarked the inference speed of the finetuned models using the `pyinfer` package (Pierse, 2020), and report the mean number of inferences per second. This is done by recording the mean inference time of running a document with 2,600 characters[17] through the model one hundred times, and repeating that process 10 times. We also compute the confidence interval as described in Section 3.3. These have all been computed using an AMD Ryzen Threadripper 1920X 12-Core CPU.

5 Benchmarking Package and CLI

To enable every language researcher to benchmark their language models in a reproducible and consistent manner, we have developed a Python package called `scandeval`[18], which can benchmark any pretrained language model available on the Hugging Face Hub[19].

The `scandeval` package is implemented as both a CLI and a Python package, which enables ease of use as both a stand-alone benchmarking tool as well as enabling integration with other Python scripts. The package follows a very *opinionated* approach to benchmarking, meaning that very few parameters can be changed. This is a deliberate design decision to enable consistent benchmarking

[13] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

[14] https://www.deepl.com/pro-api

[15] https://cloud.google.com/translate/

[16] This can be found at (redacted to preserve anonymity) and the source code is available at https://anonymous.4open.science/r/ScandiQA-8A71.

[17] The document is "This is a dummy document. ", repeated 100 times.

[18] https://anonymous.4open.science/r/ScandEval-A412

[19] https://hf.co/

of all models. The package follows the hyperparameter choices described in Section 3.2. See more in the `scandeval` documentation[20].

6 Experiments

Using the `scandeval` package we have benchmarked more than 80 pretrained models in the Scandinavian languages which were available on the Hugging Face Hub. Aside from these models we also included several multilingual models to enable a fair comparison. Lastly, to enable better interpretability of the results, we also benchmark a randomly initialised XLM-RoBERTa-base model (Conneau et al., 2020) and an ELECTRA-small model (Clark et al., 2019) on the datasets, which will make it more transparent how much "external knowledge" the pretrained models are able to utilise in their predictions. Benchmarking all these models approximately required 1000 GPU hours on a GeForce RTX 2080 Ti GPU.

6.1 Benchmarking Results

We have presented all of the benchmarked results along with their associated confidence intervals in an online leaderboard[21]. These scores have been computed as described in Section 3, and the top-5 performing models for each language, as well as overall, can be found in Table 1.

We see from Table 1 that NB-BERT-large[22] (Kummervold et al., 2021) is the best performing model in Danish, Norwegian as well as overall, with the KB-BERT-large model[23] (Malmsten et al., 2020) having the best performance in Swedish.

The massively multilingual models in the top 5 scores are RemBERT[24] (Chung et al., 2020), mDeBERTaV3[25] (He et al., 2021) and XLM-RoBERTa-large[26] (Conneau et al., 2020). The remaining models in the top 5 are NB-RoBERTa-base-scandi[27], NB-BERT-base[28] (Kummervold et al., 2021), NorBERT2[29] (Kutuzov et al., 2021), KB-BERT-base[30] (Malmsten et al., 2020) and AI-Nordics-BERT-large[31].

6.2 Cross-lingual Transfer

This experiment investigated the cross-lingual transfer capabilities of the Scandinavian models, and tested our two hypotheses from Section 1. This used the methodology described in Section 3.4. The tasks included here are the Icelandic and Faroese versions of the `ScaLA` dataset, the Icelandic `NER` dataset `MIM-GOLD-NER` (Ingólfsdóttir et al., 2020) and the Faroese part of the `NER` dataset `WikiANN` (Rahimi et al., 2019). The resulting benchmark results can be found in Table 2 and all the raw scores can be found in the appendix. The results affirm our two hypotheses, as we see that the group of languages with the largest F-statistic is the group of Mainland Scandinavian languages. We note here that the Danish score seems to contribute only slightly to the final result, which may be due to the Danish model having a substantially lower performance than the other models in the group.

6.3 Training Data Size

We aimed to choose a training data size that would be a balance between being able to differentiate between the models and being able to benchmark the models in a reasonable amount of time. In this experiment we benchmarked the same models as in Section 6.2 on truncations of named entity recognition datasets, sentiment classification datasets and linguistic acceptability datasets. The plot of one of the experiments, for the `AngryTweets` dataset (Pauli et al., 2021), can be seen in Figure 1. The plots from all the experiments can be found in the appendix.

7 Discussion

We note that the benchmarking results presented Section 6.1 show that the efforts of the National Libraries in Norway and Sweden have paid off, in the sense that their models NB-BERT-large (Kummervold et al., 2021) and KB-BERT-large (Malmsten

[20] https://anonymous.4open.science/r/ScandEval-A412
[21] This is available at (redacted to preserve anonymity).
[22] https://huggingface.co/NbAiLab/nb-bert-large
[23] https://huggingface.co/KBLab/megatron-bert-large-swedish-cased-165k
[24] https://huggingface.co/google/rembert
[25] https://huggingface.co/microsoft/mdeberta-v3-base
[26] https://huggingface.co/xlm-roberta-large
[27] https://huggingface.co/NbAiLab/nb-roberta-base-scandi
[28] https://huggingface.co/NbAiLab/nb-bert-base
[29] https://huggingface.co/ltgoslo/norbert2
[30] https://huggingface.co/KB/bert-base-swedish-cased
[31] https://huggingface.co/AI-Nordics/bert-large-swedish-cased

Rank	Overall	Danish	Norwegian	Swedish
1	NB-BERT-large	NB-BERT-large	NB-BERT-large	KB-BERT-large
2	RemBERT	RemBERT	NB-BERT-base	NB-BERT-large
3	mDeBERTaV3-base	mDeBERTaV3-base	NB-RoBERTa-base-scandi	KB-BERT-base
4	NB-RoBERTa-base-scandi	NB-RoBERTa-base-scandi	NorBERT2	AI-Nordics-BERT-large
5	NB-BERT-base	XLM-RoBERTa-large	mDeBERTaV3	RemBERT

Table 1: The five best performing pretrained models in the Mainland Scandinavian language categories.

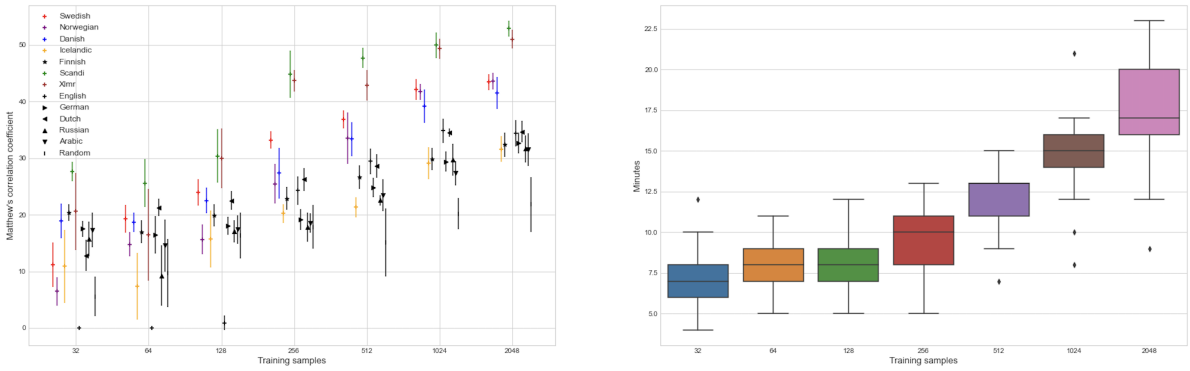

Figure 1: Plot showing the performance of different models on the `AngryTweets` dataset with varying number of training samples (left), and a boxplot showing the training time of the models with varying number of training samples (right).

Benchmark group	F-statistic	Benchmark group	F-statistic
da, no	8.94	da, sv, is	1.73
da, sv	6.58	da, sv, fo	4.41
da, is	0.85	da, is, fo	0.69
da, fo	2.81	no, sv, is	5.29
no, sv	20.05	no, sv, fo	13.71
no, is	4.35	no, is, fo	4.85
no, fo	5.55	sv, is, fo	5.80
sv, is	5.46	da, no, sv, is	5.02
sv, fo	14.89	da, no, sv, fo	18.14
is, fo	15.80	da, no, is, fo	1.79
da, no, sv	**24.18**	da, sv, is, fo	1.56
da, no, is	2.08	no, sv, is, fo	6.15
da, no, fo	6.00	da, no, sv, is, fo	4.42

Table 2: F-statistics showing the cross-lingual transfer between the Scandinavian language models. Here `da` is Danish, `no` is Norwegian, `sv` is Swedish, `is` is Icelandic and `fo` is Faroese.

et al., 2020) are outperforming the multilingual models.

This seems to indicate that investing in language technologies at a national level can be worthwhile. We also see from the same table that the Norwegian model is within the top two best models in Danish, Norwegian and Swedish, indicating a potentially large amount of language transfer, supported by the cross-lingual transfer experiment in Section 6.2. This indicates that a joint Mainland Scandinavian approach could improve the results of the current monolingual models within the Mainland Scandinavian languages.

8 Conclusion

In this paper we have presented a benchmarking framework for the Scandinavian languages, together with a Python package and CLI, `scandeval`,[32] which can be used to benchmark any model available on the Hugging Face Hub. The benchmark features four tasks: named entity recognition, sentiment classification, linguistic acceptability and question answering. We have also released two new datasets, `ScaLA` and `ScandiQA`, which constitute the linguistic acceptability and question answering tasks, respectively. We have benchmarked more than 80 models on the Mainland Scandinavian datasets in the benchmark and presented these results in an online leaderboard[33]. In our analysis of the benchmarking results we have shown substantial cross-lingual transfer between the Mainland Scandinavian languages, and no notable transfer between the group of Mainland Scandinavian languages and the group of Insular Scandinavian languages. This is the justification for including only the Mainland Scandinavian languages in the online leaderboard while maintaining support for the Insular Scandinavian languages in the `scandeval` package.

[32] `https://anonymous.4open.science/r/ScandEval-A412`

[33] (redacted to preserve anonymity)

References

2021. *Oxford English Dictionary*. Oxford University Press. https://www.lexico.com/definition/scandinavia.

Amine Abdaoui, Camille Pradel, and Grégoire Sigel. 2020. Load What You Need: Smaller Versions of Mutililingual BERT. In *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pages 119–123, Online. Association for Computational Linguistics.

Yvonne Adesam, Aleksandrs Berdicevskis, and Felix Morger. 2020. Swedishglue–towards a swedish test set for evaluating natural language understanding models.

Þórunn Arnardóttir, Hinrik Hafsteinsson, Einar Freyr Sigurðsson, Kristín Bjarnadóttir, Anton Karl Ingason, Hildur Jónsdóttir, and Steinþór Steingrímsson. 2020. A Universal Dependencies conversion pipeline for a Penn-format constituency treebank. In *Proceedings of the Fourth Workshop on Universal Dependencies (UDW 2020)*, pages 16–25.

Lukasz Augustyniak, Kamil Tagowski, Albert Sawczyn, Denis Janiak, Roman Bartusiak, Adrian Dominik Szymczak, Arkadiusz Janz, Piotr Szymański, Marcin Wątroba, Mikołaj Morzy, et al. This is the way: designing and compiling LEPISZCZE, a comprehensive NLP benchmark for Polish. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2020. Rethinking Embedding Coupling in Pre-trained Language Models. In *International Conference on Learning Representations*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2019. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *International Conference on Learning Representations*.

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

Stefan Daniel Dumitrescu, Petru Rebeja, Beata Lorincz, Mihaela Gaman, Andrei Avram, Mihai Ilie, Andrei Pruteanu, Adriana Stan, Lorena Rosia, Cristina Iacobescu, et al. 2021. LiRo: Benchmark and leaderboard for romanian language tasks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Kenneth Enevoldsen, Lasse Hansen, and Kristoffer L Nielbo. 2021. DaCy: A Unified Framework for Danish NLP. *Proceedings http://ceur-ws. org ISSN*, 1613:0073.

Jan Terje Faarlund. 2019. *The syntax of mainland Scandinavian*. Oxford University Press.

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

Jiyeon Ham, Yo Joong Choe, Kyubyong Park, Ilji Choi, and Hyungjoon Soh. 2020a. KorNLI and KorSTS: New benchmark datasets for Korean natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 422–430, Online. Association for Computational Linguistics.

Jiyeon Ham, Yo Joong Choe, Kyubyong Park, Ilji Choi, and Hyungjoon Soh. 2020b. KorNLI and KorSTS: New Benchmark Datasets for Korean Natural Language Understanding. In *Findings*.

Einar Haugen. 1976. *The Scandinavian languages: An introduction to their history*. Cambridge, Mass., Harvard University Press.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Matthew Honnibal, Ines Montani, Sofie Van Landeglem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

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, Marseille, France. European Language Resources Association.

Malte Højmark-Bertelsen. 2021. Ælæctra - A Step Towards More Efficient Danish Natural Language Processing. https://github.com/MalteHB/-l-ctra/.

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.

Tim Isbister and Magnus Sahlgren. 2020. Why Not Simply Translate? A First Swedish Evaluation Benchmark for Semantic Similarity. *arXiv preprint arXiv:2009.03116*.

Hildur Jónsdóttir and Anton Karl Ingason. 2020. Creating a parallel Icelandic dependency treebank from raw text to Universal Dependencies. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2924–2931.

Fredrik Jørgensen, Tobias Aasmoe, Anne-Stine Ruud Husevåg, Lilja Øvrelid, and Erik Velldal. 2020. NorNE: Annotating named entities for Norwegian. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4547–4556, Marseille, France. European Language Resources Association.

Simran Khanuja, Melvin Johnson, and Partha Talukdar. 2021. MergeDistill: Merging language models using pre-trained distillation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2874–2887, Online. Association for Computational Linguistics.

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 Mahabadi, 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. 2020. ParsiNLU: A Suite of Language Understanding Challenges for Persian. *Transactions of the Association for Computational Linguistics*, 9:1147–1162.

Fajri Koto, Afshin Rahimi, Jey Han Lau, and Timothy Baldwin. 2020. IndoLEM and IndoBERT: A benchmark dataset and pre-trained language model for Indonesian NLP. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 757–770, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Matthias Trautner Kromann and Stine Kern Lynge. 2004. The Danish Dependency Treebank v. 1.0.

Per E Kummervold, Javier De la Rosa, Freddy Wetjen, and Svein Arne Brygfjeld. 2021. Operationalizing a National Digital Library: The Case for a Norwegian Transformer Model. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 20–29, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Andrey Kutuzov, Jeremy Barnes, Erik Velldal, Lilja Øvrelid, and Stephan Oepen. 2021. Large-scale contextualised language modelling for norwegian. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 30–40.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, 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 datasetfor 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.

Shayne Longpre, Yi Lu, and Joachim Daiber. 2021. MKQA: A linguistically diverse benchmark for multilingual open domain question answering. *Transactions of the Association for Computational Linguistics*, 9:1389–1406.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

Martin Malmsten, Love Börjeson, and Chris Haffenden. 2020. Playing with Words at the National Library of Sweden – Making a Swedish BERT.

Brian W Matthews. 1975. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451.

Jens Dahl Møllerhøj. 2020. Nordic BERT. `https://github.com/certainlyio/nordic_bert`.

Joakim Nivre, Jens Nilsson, and Johan Hall. 2006. Talbanken05: A Swedish treebank with phrase structure and dependency annotation. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy. European Language Resources Association (ELRA).

Lilja Øvrelid and Petter Hohle. 2016. Universal Dependencies for Norwegian. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1579–1585, Portorož, Slovenia. European Language Resources Association (ELRA).

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, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Charles Pierse. 2020. Pyinfer. `https://github.com/cdpierse/pyinfer`.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

David C Plaut et al. 1986. Experiments on Learning by Back Propagation.

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, Florence, Italy. Association for Computational Linguistics.

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.

Eiríkur Rögnvaldsson, Anton Karl Ingason, Einar Freyr Sigurðsson, and Joel Wallenberg. 2012. The Icelandic parsed historical corpus (IcePaHC). In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 1977–1984.

Piotr Rybak, Robert Mroczkowski, Janusz Tracz, and Ireneusz Gawlik. 2020. KLEJ: Comprehensive benchmark for Polish language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1191–1201, Online. Association for Computational Linguistics.

Ali Safaya, Emirhan Kurtulucs, Arda Goktougan, and Deniz Yuret. 2022. Mukayese: Turkish NLP Strikes Back. In *Findings*.

Magnus Sahlgren, Fredrik Carlsson, Fredrik Olsson, and Love Börjeson. 2021. It's basically the same language anyway: the case for a nordic language model. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 367–372, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. 2020. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947.

Philip Tamimi Sarnikowski. 2021. Danish Transformers. GitHub. `https://github.com/sarnikowski`.

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.

Vésteinn Snæbjarnarson. 2021. Automated methods for Question-Answering in Icelandic. Master's thesis, University of Iceland, Reykjavík.

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

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.

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).

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.

Alex Warstadt, Amanpreet Singh, and Samuel Bowman. 2019. Neural Network Acceptability Judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Bryan Wilie, Karissa Vincentio, Genta Indra Winata, Samuel Cahyawijaya, Xiaohong Li, Zhi Yuan Lim, Sidik Soleman, Rahmad Mahendra, Pascale Fung, Syafri Bahar, and Ayu Purwarianti. 2020. IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 843–857, Suzhou, China. Association for Computational Linguistics.

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.

Beilei Xiang, Changbing Yang, Yu Li, Alex Warstadt, and Katharina Kann. 2021. CLiMP: A benchmark for Chinese language model evaluation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2784–2790, Online. Association for Computational Linguistics.

Liang Xu, Xuanwei Zhang, Lu Li, Hai Hu, Chenjie Cao, Weitang Liu, Junyi Li, Yudong Li, Kai Sun, Yechen Xu, Yiming Cui, Cong Yu, Qianqian Dong, Yin Tian, Dian Yu, Bo Shi, Jun jie Zeng, Rongzhao Wang, Weijian Xie, Yanting Li, Yina Patterson, Zuoyu Tian, Yiwen Zhang, He Zhou, Shaoweihua Liu, Qipeng Zhao, Cong Yue, Xinrui Zhang, Zhen-Yi Yang, Kyle Richardson, and Zhenzhong Lan. 2020. CLUE: A Chinese Language Understanding Evaluation Benchmark. In *International Conference on Computational Linguistics*.

A Cross-lingual transfer experiment

Language	Hugging Face Model ID	# Parameters
Danish	`Maltehb/danish-bert-botxo`	111M
Swedish	`KB/bert-base-swedish-cased`	125M
Norwegian	`patrickvonplaten/norwegian-roberta-base`	125M
Icelandic	`mideind/IceBERT`	124M
English	`roberta-base`	125M
German	`deepset/gbert-base`	110M
Dutch	`pdelobelle/robbert-v2-dutch-base`	117M
Finnish	`TurkuNLP/bert-base-finnish-cased-v1`	125M
Russian	`DeepPavlov/rubert-base-cased`	178M
Arabic	`asafaya/bert-base-arabic`	111M

Table 3: The Hugging Face Hub model IDs of the models used in the cross-lingual transfer experiment.

Model	Danish Score	Norwegian Score	Swedish Score	Icelandic Score	Faroese Score
`Danish`	**50.76 ± 1.75**	39.61 ± 2.12	34.92 ± 2.45	5.58 ± 1.99	36.87 ± 1.03
`Norwegian`	46.30 ± 2.83	**58.78 ± 1.44**	46.90 ± 2.79	28.85 ± 1.45	43.35 ± 2.20
`Swedish`	45.81 ± 2.96	47.32 ± 2.66	**69.29 ± 1.40**	28.69 ± 1.61	43.63 ± 1.90
`Icelandic`	30.20 ± 1.23	28.68 ± 2.91	36.80 ± 2.14	**71.00 ± 1.50**	**48.26 ± 4.76**
`Finnish`	32.55 ± 1.47	30.71 ± 2.14	38.94 ± 1.51	16.33 ± 1.89	36.87 ± 1.17
`English`	34.11 ± 2.11	30.92 ± 2.69	39.24 ± 1.92	28.39 ± 2.41	40.75 ± 1.59
`German`	28.13 ± 2.04	27.58 ± 2.90	37.62 ± 4.18	26.13 ± 1.63	41.02 ± 1.46
`Dutch`	31.78 ± 1.62	28.27 ± 2.51	35.06 ± 1.87	26.21 ± 1.79	40.83 ± 1.70
`Russian`	33.91 ± 1.88	33.55 ± 2.17	39.14 ± 2.33	29.96 ± 1.58	43.17 ± 1.66
`Arabic`	22.89 ± 1.82	19.98 ± 2.24	25.40 ± 2.69	10.33 ± 2.19	35.33 ± 1.57

Table 4: The raw benchmarking results used in the cross-lingual transfer experiment.

B Training Data Size Experiment

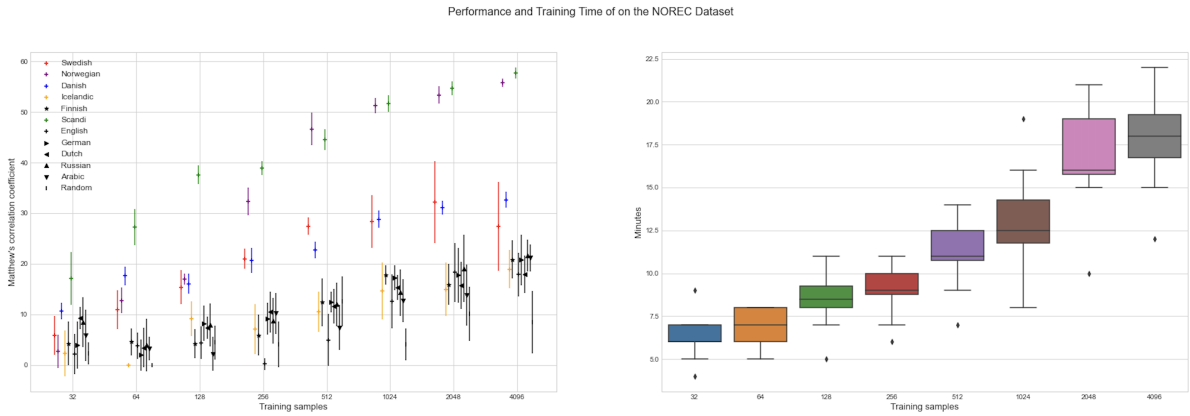

Figure 2: The results from the training data size experiment for the `NoReC` dataset.

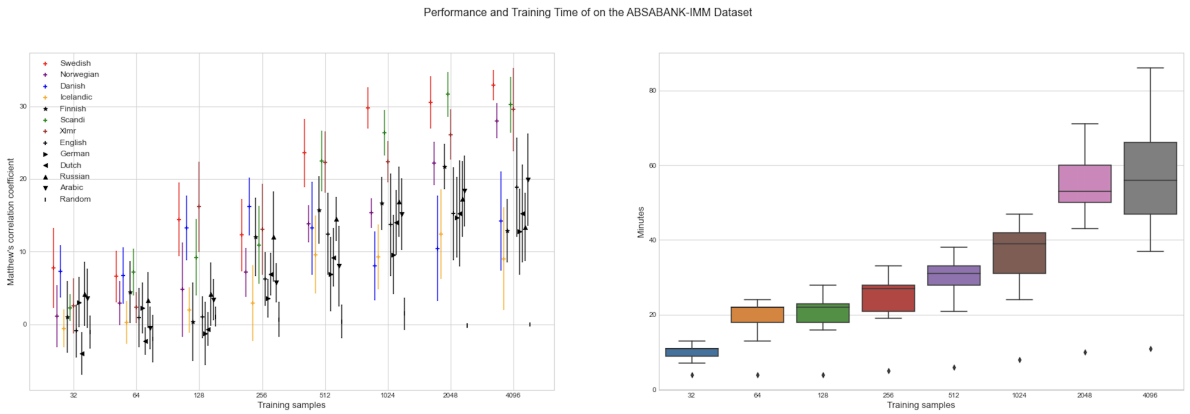

Figure 3: The results from the training data size experiment for the `Absabank-Imm` dataset (Adesam et al., 2020).

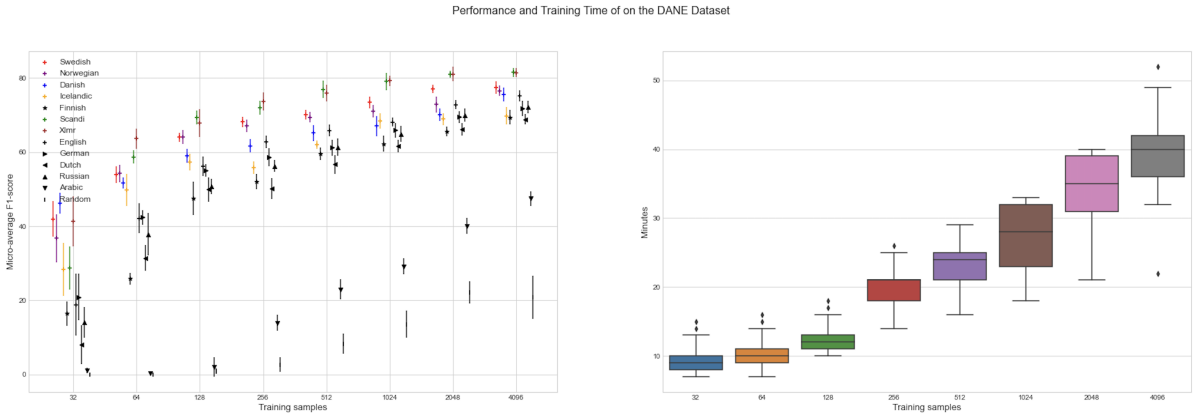

Figure 4: The results from the training data size experiment for the `DaNE` dataset.

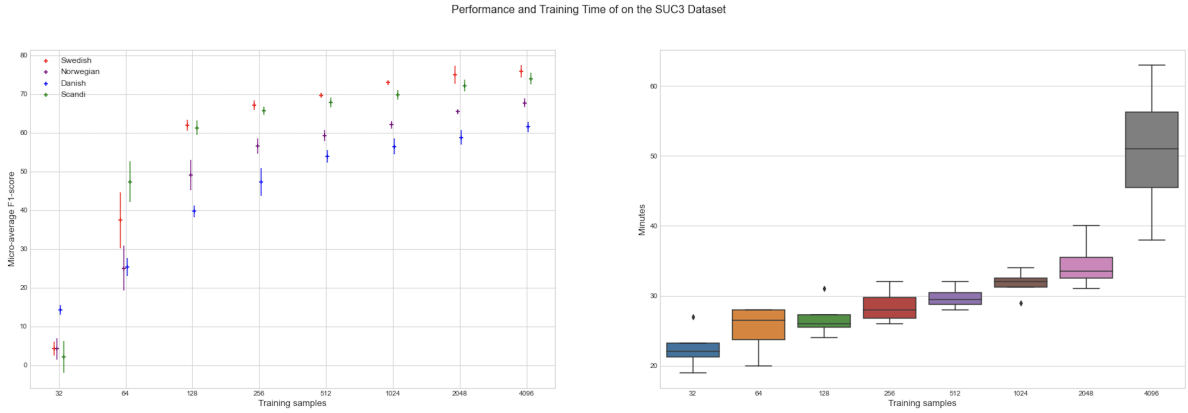

Figure 5: The results from the training data size experiment for the `SUC3` dataset.

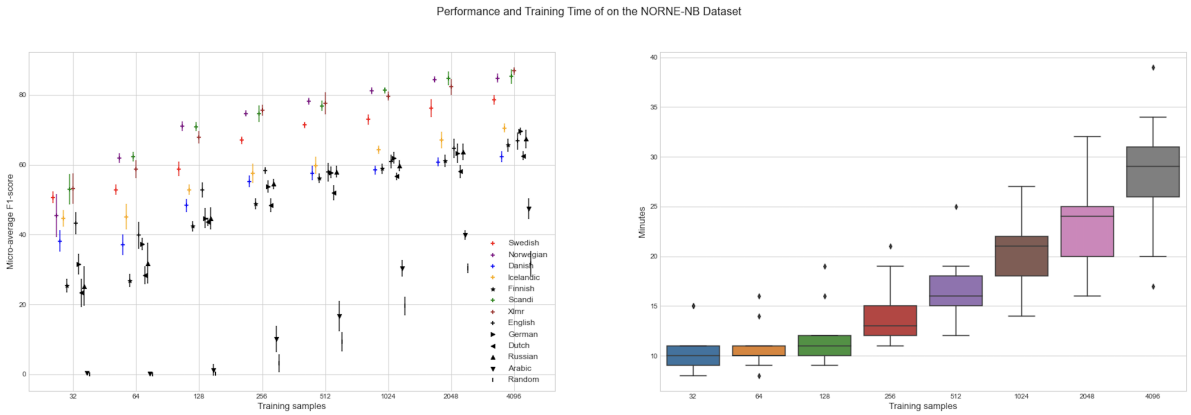

Figure 6: The results from the training data size experiment for the `NorNE-NB` dataset.

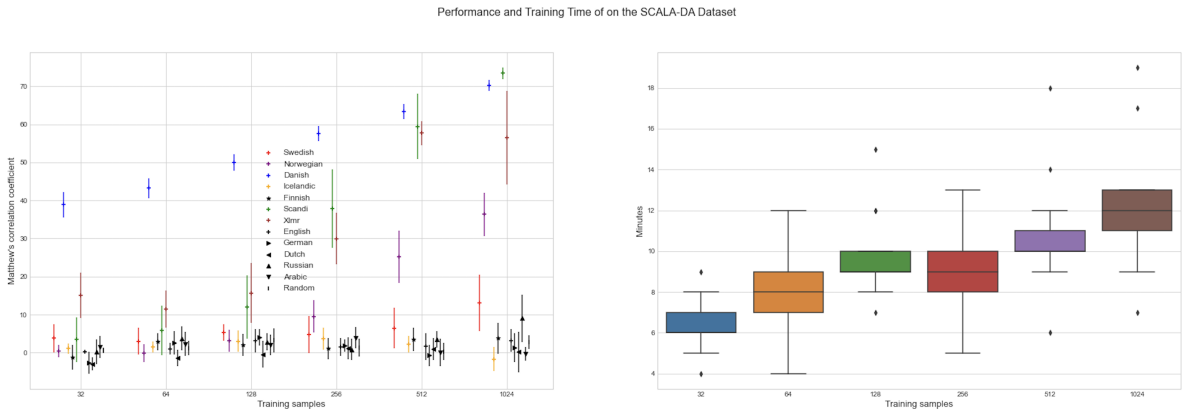

Figure 7: The results from the training data size experiment for the `ScaLA-DA` dataset.

Figure 8: The results from the training data size experiment for the `ScaLA-SV` dataset.

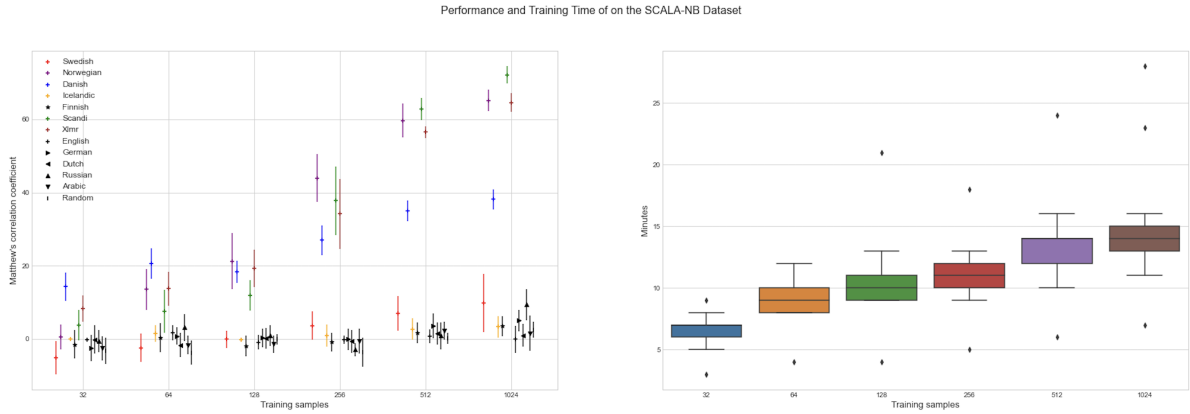

Figure 9: The results from the training data size experiment for the `ScaLA-NB` dataset.

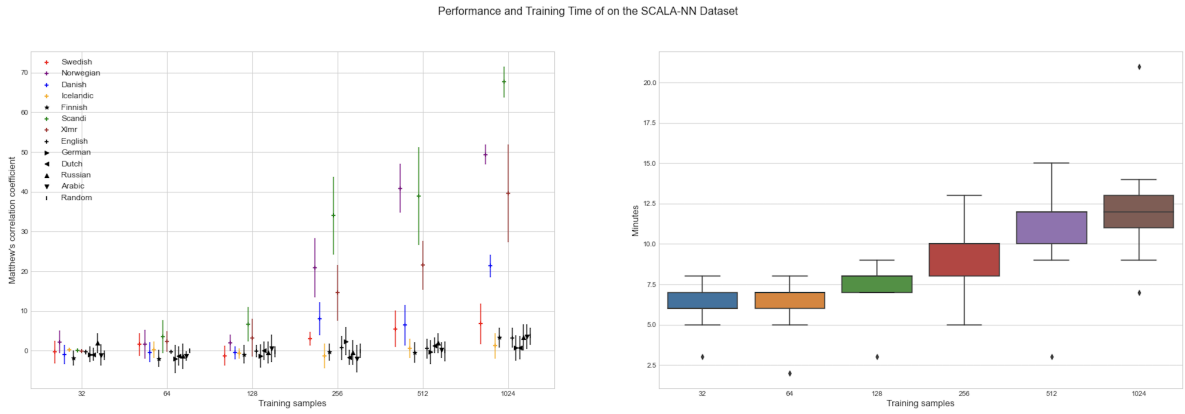

Figure 10: The results from the training data size experiment for the `ScaLA-NN` dataset.