Computational Linguistics in the Netherlands Journal 13 (2024) 193-203 Submitted 02/2024; Published 03/2024

RobBERT-2023: Keeping Dutch Language Models Up-To-Date at a Lower Cost Thanks to Model Conversion

Pieter Delobelle* pieter.delobelle@kuleuven.be
François Remy** francois.remy@ugent.be
(equal contributions)

*Department of Computer Science; Leuven.AI, KU Leuven, Belgium
**Internet and Data science Lab, Ghent University, Belgium

Abstract

Pre-training large transformer-based language models on gigantic corpora and later repurposing them as base models for finetuning on downstream tasks has proven instrumental to the recent advances in computational linguistics. However, the prohibitively high cost associated with pre-training often hampers the regular updates of base models to incorporate the latest linguistic developments. To address this issue, we present an innovative approach for efficiently producing more powerful and up-to-date versions of RobBERT, our series of cutting-edge Dutch language models, by leveraging existing language models designed for high-resource languages. Unlike the prior versions of RobBERT, which relied on the training methodology of RoBERTa but required a fresh weight initialization, our two RobBERT-2023 models (base and large) are entirely initialized using the RoBERTa-family of models. To initialize an embedding table tailored to the newly devised Dutch tokenizer, we rely on a token translation strategy introduced by Remy et al. (2023). Along with our RobBERT-2023 release, we deliver a freshly pre-trained Dutch tokenizer using the latest version of the Dutch OSCAR corpus. This corpus incorporates new high-frequency terms, such as those related to the COVID-19 pandemic, cryptocurrencies, and the ongoing energy crisis, while mitigating the inclusion of previously over-represented terms from adult-oriented content. To assess the value of RobBERT-2023, we evaluate its performance using the same benchmarks employed for the state-of-the-art RobBERT-2022 model, as well as the newly-released Dutch Model Benchmark. Our experimental results demonstrate that RobBERT-2023 not only surpasses its predecessor in various aspects but also achieves these enhancements at a significantly reduced training cost. This work represents a significant step forward in keeping Dutch language models up-to-date and demonstrates the potential of model conversion techniques for reducing the environmental footprint of NLP research.

1. Introduction

Transformer-based Language Models have become a key component of the modern NLP pipeline, and their size has ballooned to unprecedented levels in the recent years (Zhao et al. 2023), at least when it comes to high-resource languages such as English or Chinese. Auto-regressive models such as Falcon (Almazrouei et al. 2023, 40 billion parameters) or Llama2 (Touvron et al. 2023, 70 billion parameters) lead the way in terms of parameter count, but even bi-directional transformer models used for classification and retrieval tasks have become more parameter-heavy as time passed, with examples such as Roberta-Large (Liu et al. 2019, 345 million parameters), Megatron-BERT (NVIDIA, Applied Deep Learning Research team 2023, 345 million parameters), or DeBERTa-Large (He et al. 2023, 435 million parameters).

This evolution has not been equally followed by language models destined to lower-resource languages, such as Dutch, where even the largest BERT-type language models to this day remained capped at 117M parameters (Delobelle et al. 2022). Multiple reasons explain this fact but both the

prohibitive cost of their training and the difficulty to assemble a corpus large enough to train such large models from scratch are certainly part of the answer.

On the other hand, multilingual language models have offered a very welcome solution to many researchers around the world, by enabling the pooling of the training costs and data of such language models across a multitude of languages. While this has permitted a large number of languages to benefit from the increase of model capacity brought by larger language models, this came at the cost of ineffective tokenization (drastically increasing the inference speed of such models, as their computation costs grows quadratically to the sequence length) despite also requiring the use of large vocabularies (increasing the memory consumption of such models, and imposing restrictions to their usage on edge devices), as well as causing a range of negative effects due to language interference (Limisiewicz et al. 2023).

In the past, several model conversion techniques were devised to work around this problem (Artetxe et al. 2020, de Vries et al. 2021, Garcia et al. 2021, Gogoulou et al. 2022). These techniques all leverage monolingual models developed for high-resource languages such as English (and known as the source language) as a base building block for the creation of models tailored for another language (known as the target language). To this day, strategies of this type had never managed to yield a performance level equivalent to the one of models trained for scratch for the target language, limiting the appeal of the proposed model conversion techniques.

However, the Tik-to-Tok technique recently introduced by Remy et al. (2023) has demonstrated that an effective model conversation was achievable, using a token matching strategy generalizing over a word translation dictionary. In their paper, the authors converted both large and small English RoBERTa-based models into potent Dutch models, showcasing even stronger performance than equivalently sized models trained from scratch in Dutch.

However, to enable fair comparison with the state-of-the-art approaches, the converted models from the Tik-to-Tok publication relied on the same tokenizer and training corpus as pre-existing Dutch models. In this paper, we introduce two new RoBERTa-based Dutch models (large and base) trained on the brand new Oscar-2023-01 corpus (Abadji et al. 2022), using a newly-trained Dutch tokenizer which can take better into consideration the lexical and semantic changes that have happened in the recent years as a result of macro-economic and political changes, while simultaneously ensuring a fairer and less biased token vocabulary, thanks to the new data filtering efforts that have been put in place by the Oscar-2023 team (Jansen et al. 2022).

2. Related work

BERT-based models (Devlin et al. 2019) perform exceptionally well for natural language classification and regression tasks, such as sentiment analysis, natural language inference, question-answering and more (Liu et al. 2019). These models are pre-trained using a Masked Language Modeling (MLM) objective on a large unlabeled training dataset, and then finetuned on different downstream tasks using smaller datasets. In the MLM task, the model has to predict the correct tokens for masked and swapped tokens in text from the original pre-training corpus. The original BERT model also incorporated a second training objective, Next Sentence Prediction (NSP). However, this has been shown to not improve performance liu2019roberta. Consequentally, this NSP objective was not used to train the RoBERTa model and we also disgard it.

BERT's tokenizer splits input sentences into a sequence of tokens according to a vocabulary. These tokens can correspond to words for frequent words, but are often subwords as well. A popular technique is byte-pair encodings (BPE), which chooses the most frequent symbol pairs to merge into larger tokens until a desired vocabulary size is reached (Sennrich et al. 2016, Liu et al. 2019).

The original BERT model was trained on English text and a later model, called Multilingual BERT (mBERT) was trained on a collection of over 100 languages (Devlin et al. 2019). While multilingual models generally perform quite well, monolingual models tend to outperform them for monolingual language tasks of that particular language (Martin et al. 2019, Delobelle et al. 2020). For

Dutch, the state-of-the-art model is the RobBERT model, which was released in the first month of 2020 delobelle2020robbert. The model was trained using the RoBERTa architecture (Liu et al. 2019) on the Dutch OSCAR corpus from 2019 (Ortiz Suárez et al. 2019). In parallel, both BERTje and BERT-NL were developed and are trained on a more formal but smaller dataset using the BERT architecture (de Vries et al. 2019, Brandsen et al. 2019). Due to RobBERT using a larger dataset and optimized architecture, it outperforms the other BERT models on most tasks.

3. Methodology

In this article, we present the approach using which we produced the new RobBERT-2023 models. We start by presenting the methodology used to produce our new high-quality Dutch tokenizer, and continue by exposing the training details of our two models (base and large).

3.1 Training the new tokenizer

We train a new Dutch tokenizer consisting of 50k tokens on a randomly-selected subset of the Dutch portion of the Oscar 2023-01 corpus. To compute these 50k tokens, we reused the same BPE algorithm which has previously been used to create the vocabulary of RoBERTa as well as the previous versions of RobBERT.

It can be noted that RobBERT-2023 features a slightly larger vocabulary than RobBERT-2022, which only featured 42k tokens. The slightly-increased number of tokens was chosen to match with the vocabulary size used by the tokenizer of the English RoBERTa model, from which we intend to initiate the conversion.

Unlike the previous versions of the Oscar corpus which were language-tagged at the sentence, this new version is tagged at the document level. This means that the corpus likely contains a few sentences in many languages across the globe, as web pages might frequently be partly multilingual.

This proved challenging as the BPE algorithm is a merge-based protocol, which starts by assigning one token to each character in the training set (i.e. letters, numbers, punctuation signs, but also ideograms), resulting in thousands of tokens being occupied by Chinese and Japanese ideograms, most of which appear only once or a very limited amount of times in the training corpus, and of which the model is therefore unlikely to be able to make sense of, wasting a precious entry of the embedding matrix.

To counteract this problem, an extensive corpus cleaning was first implemented, in order to remove characters not belonging to the Latin scripts or emoji Unicode planes, in order to reserve a maximum of number of tokens for Dutch words. We also performed full character composition normalization (NFKC), to reduce variation in the encoding of accents and other foreign-language diacritics. We provide a copy of our normalization script along with our models on HuggingFace.

To assess the potential impact of our new tokenizer, we provide below a few cherry-picked examples of newly-added and newly-removed tokens, by comparing our new tokenizer to the RobBERT-2022 tokenizer. The goal of this short list is to illustrate several of the patterns we noticed by comparing both vocabularies with each other, including the addition of new tokens related to the Covid-19 epidemic (e.g., *coronavirus*, *coronamaatregelen*, *coronatijd*, etc...), the energy crisis (e.g., *duurzamere*, *energieneutraal*, *energieprijzen*, etc...), the rise in prominence of crypto-currencies (e.g., *crypto*, *cryptomunten*, *betaalmiddel*, etc...), the increase in attention to privacy matters following GDPR (e.g., *geanonimiseerd*, *locatiegegevens*, *privacywetgeving*, etc...), or even the rise to prominence of new social networks (e.g. *influencer*, *Meta*, *TikTok*, etc..).

In the list of tokens which are no longer listed in the new vocabulary, we can find a few technologies whose relevance fell even further in the recent years (e.g., *desktops*, *modem*, *telefoongids*, etc...). We can also note the effect of the new quality filtering implemented by the Oscar team, principally visible in the removal of pornographic and/or illegal content from the crawled pages, resulting in the disappearance of many tokens related to adult sites (e.g., *naaktfotos*, *pornoster*, *seksvideos*, etc...).

In summary, we believe the new RobBERT-2023 tokenizer is a significant improvement over the previous versions of our Dutch tokenizer, and it enables a significantly more efficient processing of modern texts written in Dutch.

3.2 Training the new RobBERT-2023 base model

With the goal of providing a new and up-to-date version of the RobBERT model, we found ourselves well-aligned with the Online Language Modelling group (Thrush and Oblokulov 2022, henceforth OLM) aiming to regularly produce new Transformer models from English based on new C4 CommonCrawl versions (Dodge et al. 2021), to maximally ensure that models they release are aware of recent facts and events.

We therefore converted their "olm-base" model using the Tik-to-Tok strategy to use our new Dutch tokenizer. Following the strategy outlined in the Tik-to-Tok paper, we first initialized the new embedding table using an Englo-Dutch FastText model and the original model weights. We then finetuned the resulting embeddings on the Oscar 2023-01 corpus for 150 thousand steps, and further finetuned the entire model for 50 thousand steps. These 50 thousand steps were further divided into 30 thousand steps with learning rate 5e-5, weight decay 0.01, and only the two top and bottom layers unfrozen, and 20 thousand steps with learning rate 1e-5, no weight decay, and all the weights unfrozen.

We performed this using a single NVIDIA 2080 GPU for 9 days (5 for the 150k embedding steps, and 4 days for the finetuning phases), using as few as 20% of the compute required for training the original RobBERT model.

3.3 Training the new RobBERT-2023 large model

In contrast to previous RobBERT release, we decided to provide this year a new "large" model in addition to the base model usually proposed. To achieve this, we decided to convert the Roberta-Large model using the same strategy as above.

This time, we used an NVIDIA V100 GPU for 6 days. This is significantly less than what would have been required to train such a language model from scratch, but more importantly this also used much less data. Indeed, this training run did not even need all the available Dutch data in Oscar 2023-01 to achieve convergence (50% of the corpus was not used during training). This means that training an even larger model to convergence would still be possible, even though we leave this unexplored in this paper.

4. Evaluation

We evaluate our new models using the newly released Dutch Model Benchmark (de Vries et al. 2023, henceforth DUMB). This benchmark evaluates the performance of masked-language models on 9 downstream tasks which aim to cover the entire range of NLP tasks for which bi-directional models such as RobBERT are employed in the community. These tasks range from token-level classifications (e.g., part-of-speech-tagging, named-entity-recognition) to document-level classification tasks (e.g., sentiment analysis, hate speech detection) but also includes word-pair (e.g. word sense disambiguation, pronoun resolution) and sentence-level classification tasks (e.g., semantic textual similarity, natural language inference).

This benchmark attempts to provide the highest possible scores for all the models that are evaluated on its tasks, by performing an extensive hyper-parameter search for each of these tasks, and training optimized models for all of them independently.

In the end, the results of these evalustions are provided both in terms of absolute scores for each of the tasks, as well as a relative error reduction over the BERTje baseline (reported in Table 1).

Table 1: **Task scores and Relative Error Reduction (RER) scores per model**. Models are grouped by pre-train language and model size. Bold values indicate highest (or not significantly different, $p \geq 0.05$) scores per task. Gray values are significantly ($p < 0.05$) below baseline. Updated results with newer models can be found on `dumbench.nl`.

		Word				Word Pair				Sentence Pair				Document					
	Avg	**POS**		**NER**		**WSD**		**PR**		**CR**		**NLI**		**SA**		**ALD**		**QA**	
Model	RER	RER	Acc.	RER	F1	RER	Acc.	RER	Acc.	RER	Acc.	RER	Acc.	RER	Acc.	RER	F1	RER	F1
BERTje	0	0	97.8	0	86.1	0	65.9	**0**	**65.8**	0	62.0	0	85.2	0	93.3	0	58.8	0	70.3
RobBERT$_{V1}$	-16.3	12.5	98.1	-19.4	83.5	-15.3	60.6	-24.0	57.6	-14.7	56.4	-12.7	83.3	-58.2	89.4	4.8	60.8	-19.4	64.6
RobBERT$_{V2}$	1.6	16.2	98.2	4.1	86.7	-5.3	64.1	**0.1**	**65.8**	-10.2	58.1	-3.8	84.6	-0.5	93.2	12.0	63.7	2.2	71.0
RobBERT$_{2022}$	3.6	17.3	98.2	7.6	87.2	-6.4	63.7	**-1.8**	**65.2**	-10.1	58.2	3.1	85.6	4.0	93.5	18.9	66.6	-0.2	70.3
RobBERT$_{2023\ base}$	3.6	17.3	98.2	7.6	87.2	-6.4	63.7	**-1.8**	**65.2**	-10.1	58.2	3.1	85.6	4.0	93.5	18.9	66.6	-0.2	70.3
RobBERT$_{2023\ large}$	**18.6**	15.9	98.1	19.0	88.8	4.0	67.2	**-0.9**	**65.5**	**47.1**	**79.9**	**27.7**	**89.3**	**21.9**	**94.7**	16.5	65.6	16.2	75.1
mBERT$_{cased}$	-5.8	6.2	97.9	9.2	87.4	7.7	68.5	-11.0	62.0	-18.4	55.0	-6.2	84.3	-41.7	90.5	-4.5	56.9	6.9	72.4
XLM-R$_{base}$	-0.3	13.9	98.1	10.8	87.6	1.9	66.5	-16.2	60.2	-26.8	51.8	2.0	85.5	-3.6	93.0	3.4	60.2	12.3	74.0
mDeBERTaV3$_{base}$	12.8	18.2	98.2	17.2	88.5	10.8	69.6	-20.8	58.7	19.7	69.5	25.2	88.9	3.3	93.5	12.4	63.9	29.2	79.0
XLM-R$_{large}$	14.4	**26.5**	**98.4**	**29.7**	**90.3**	**21.3**	**73.1**	-15.8	60.4	-25.8	52.2	24.4	88.8	13.2	94.2	**19.0**	**66.6**	37.2	81.4
BERT$_{base}$	-42.8	-19.8	97.4	-30.8	81.9	-22.4	58.2	-18.7	59.4	-28.0	51.4	-19.2	82.3	-203.9	79.6	-16.1	52.2	-26.2	62.5
RoBERTa$_{base}$	-25.6	-6.5	97.7	-27.3	82.3	-14.0	61.1	-20.4	58.8	-24.1	52.8	-19.7	82.3	-99.9	86.6	-16.0	52.2	-2.1	69.7
DeBERTaV3$_{base}$	-1.6	6.5	97.9	1.7	86.4	-4.2	64.4	-25.3	57.1	-20.5	54.2	8.6	86.5	-14.6	92.3	3.5	60.2	29.7	79.1
BERT$_{large}$	-35.1	-12.0	97.5	-25.9	82.5	-25.4	57.2	-29.3	55.8	-31.2	50.2	-15.4	82.9	-158.7	82.6	-7.8	55.6	-10.4	67.2
RoBERTa$_{large}$	-14.1	6.4	97.9	-12.3	84.4	-19.8	59.1	-23.3	57.8	-26.1	52.1	-8.5	83.9	-63.8	89.0	1.2	59.3	19.7	76.2
DeBERTaV3$_{large}$	15.7	17.9	98.2	10.9	87.6	12.7	70.2	-14.4	60.9	35.4	75.4	24.1	88.7	-6.4	92.8	12.5	64.0	**48.4**	**84.7**

Averaged over all tasks, our new RobBERT-large model outperforms all other models currently included in the benchmark (17 of them, as of the time of publication).

These models range from other monolingual Dutch models such as BERTje (de Vries et al. 2019) and previous iterations of RobBERT (Delobelle et al. 2020), to large multilingual models such as XLMR (Conneau et al. 2020), and even include large monolingual English model such as DeBERTAv3 (He et al. 2023), which compensate their lack of Dutch knowledge by their significanctly increased parameter count, which enables some of them to learn enough Duch vocabulary during finetuning to perform well on some of the evaluated tasks.

In the paper presenting the Dutch Model Benchmark, de Vries et al. (2023) perform a theoretical computation of the expected gain of training from scratch a large monolingual Dutch model based on the Roberta architecture, similar to the one presented in this paper, and estimate such a model would obtain an error-reduction rate of 13.4 while our model manages to achieve an even-more-impressive error reduction of 18.6, showing that converting large English models is not only a more efficient strategy to produce large language models for Dutch, but also that it achieves a performance level on par or even better than the theoretical performance of such a model trained from scratch.

In the next sections, we go over each of the tasks, and analyze the performance of our models compared to the other models in the list.

4.1 Document-level classification

When it comes to document classification tasks, RobBERT-2023-large seems to establish a solid new state of the art, with more than 15% error reduction on the question answering and sentiment analysis tasks. It performs competitively with the large multilingual models, while being smaller than these models. Meanwhile, the RobBERT-2023-base model also performs better than the pre-existing Dutch model (8% error reduction on average) but it remains largely overshadowed by the larger models. Based on these results, it appears that RobBERT-2023 models should be considered as the new default choice for researchers which want to perform document classification tasks in Dutch.

4.2 Sentence-level classification

RobBERT-2023-large performs exceptionally well on the two sentence classification tasks evaluated in DUMB: causal reasoning, and natural language interference. It surpasses its closest competitor for causal reasoning (DeBERTaV3-large) by 4.5% in absolute accuracy, an error reduction of 12%. It also surpasses its closest contender for natural language inference by 1% in absolute accuracy, an error reduction of 2.5%. RobBERT-2023-base performs best among the base-sized model, but remains distant from the larger models. When it comes to sentence-level tasks, RobBERT-2023 models show themselves to be the new state-of-the-art when it comes to Dutch, with a significant advantage over current baselines.

4.3 Token-level classification

In word and word pair tasks, RobBERT-2023 Large outperforms again all known Dutch models, but it does not perform better than large multilingual models. This is especially true for the tasks of Named Entity Recognition (where we posit that the large multilingual corpus is a strength, increasing the set of known entities by a significant factor) and Word Sense Disambiguation (where aligning embeddings across many languages is obviously an advantage, as words do not share the same homonyms across languages). The base model seems to perform pretty similarly to the large model on these tasks, to the exception of pronoun resolution. We do not have a strong hypothesis to justify this result. Overall, this seems to indicate that, when it comes to token classification, smaller models perform well while benefiting from a faster and cheaper inference. When performance is critical, token-level classification tasks should however remain solved using large multilingual models.

4.4 Benefits of updating a language model

Finally, an important aspect of our work consists in the fact it enables to produce up-to-date Dutch language models at a low cost and effort. To show that these updated models provide benefits in practice, we turn to the evaluation of RobBERT-2022 (Delobelle et al. 2022), our first attempt at keeping large language models up-to-date.

Delobelle et al. (2022) created a model based on RobBERT (Delobelle et al. 2020), which was originally trained in 2019. This model extended the tokenizer by introducing new token types based on the difference of a new tokenizer trained on OSCAR-2022 and the original RobBERT tokenizer. They argued that the benefit was that tokens like 'COVID-19' could be represented in single tokens with corresponding trained embeddings.

However, extending a BPE vocabulary is not possible without recreating the exact same merges on the same dataset, thus defeating the goal of adding novel tokens to account for language shifts. We create a new BPE vocabulary, as described in § 3.1 and intialize our model wih Tik-to-Tok (Remy et al. 2023) to address this limitation.

To evaluate the effectiveness of training a new language model, we test our model on Vaccin-Chat (Buhmann et al. 2022), see Table 2. This task became available after the release of $\text{RobBERT}_{\text{v2}}$, as the COVID-19 pandemic did not happen yet and was consequentely not present in the original corpus. We observe that both the base model and the large model have a reasonable improvement over $\text{RobBERT}_{\text{v2}}$, illustrating the benefit of an updated tokenizer with pretraining on an up-to-date corpus.

5. Limitations

5.1 Conversion-induced English biases

The initialization of our Dutch models with an English model is likely to have had an impact on the range of type of linguistic idioms understood by our RobBERT-2023 models. English-only idioms

Table 2: **Results on VaccinChat** (Buhmann et al. 2022), a task on FAQ about COVID-19 vaccines. We compare our RobBERT-2023 models with $\text{RobBERT}_{\text{v2}}$ as a baseline, since this is the most competivive model trained before the COVID-19 crisis. Results without F_1 score are reported by Buhmann et al. (2022).

Model	**ACC** ↑	$\mathbf{F_1}$ ↑
Domain-adapted models		
BERTje+	77.7%	—
CoNTACT+	77.9%	—
General-purpose models		
BERTje	74.7%	—
$\text{RobBERT}_{\text{v2}}$	74.9%	77.2%
RobBERT_{2022}	76.3%	79.3%
$\text{RobBERT}_{\text{2023 base}}$	77.5% ↑ 3%	80.2% ↑ 4%
$\text{RobBERT}_{\text{2023 large}}$	**79.4%** ↑ 6%	**82.5%** ↑ 7%

might now be considered as valid Dutch idioms by our model as a result of its initialization, while some Dutch idioms might have been difficult to learn due to lack of similar patterns in English. We leave as a future work the investigation of this possible contamination.

5.2 Converted model types

One of the limitation of this work is that it only attempts the conversion of Roberta-based models from English to Dutch, while other model architectures have now been proposed, such as DeBERTAv3 (He et al. 2023), which could be more promising than Roberta as an initialization. While the Tik-to-Tok technique (Remy et al. 2023) employed in this paper does not preclude the conversion of other types of models, the training of DeBERTAv3 models require both a masked-language-model and a discriminative model to be trained jointly, making a conversion strategy more difficult as the conversion would not preserve well the joint training objective. We leave as future work the extension of this technique to novel types of model architectures.

6. Conclusion

In this article, we introduced two new state-of-the-art monolingual Dutch models, RobBERT-2023-base and RobBERT-2023-large. Based on monolingual English models and the Tik-to-Tok model conversion strategy, these two models achieve substantially better performance than existing monolingual Dutch, multilingual and large monolingual English models on a wide array of Dutch NLP tasks, as evidenced by their score on the DUMB benchmark. Thanks to their entirely new and revised tokenizer, these models are the most efficient models available to encode and process Dutch text to date. We hope that these new models will continue the now well-established tradition of RobBERT models and keep propelling it to new heights in the future, thanks to the possibilities introduced by a faster-than-ever training of high-quality models.

Acknowledgements

We thank Matthieu Meeus and Anthony Rathé for the productive environment that kickstarted this project.

Pieter Delobelle received funding from the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" programme and was supported by the Research Foundation - Flanders (FWO) under EOS No. 30992574 (VeriLearn). He received a grant from "Interne Fondsen KU Leuven/Internal Funds KU Leuven". François Remy thanks the Flemish Innovation Agency (VLAIO) for the funding of his research, and the infrastructure team of IDLab for access to the compute node required for enabling this work.

References

Abadji, Julien, Pedro Ortiz Suarez, Laurent Romary, and Benoît Sagot (2022), Towards a cleaner document-oriented multilingual crawled corpus, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, European Language Resources Association, Marseille, France, pp. 4344–4355. https://aclanthology.org/2022.lrec-1.463.

Almazrouei, Ebtesam, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo (2023), Falcon-40B: an open large language model with state-of-the-art performance.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre (2020), Translation artifacts in cross-lingual transfer learning, *in* Webber, Bonnie, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics, Online, pp. 7674–7684. https://aclanthology.org/2020.emnlp-main.618.

Brandsen, Alex, Anne Dirkson, Suzan Verberne, Maya Sappelli, Dung Manh Chu, and Kimberly Stoutjesdijk (2019), BERT-NL a set of language models pre-trained on the Dutch SoNaR corpus. http://textdata.nl.

Buhmann, Jeska, Maxime De Bruyn, Ehsan Lotfi, and Walter Daelemans (2022), Domain- and task-adaptation for VaccinChatNL, a Dutch COVID-19 FAQ answering corpus and classification model, *Proceedings of the 29th International Conference on Computational Linguistics*, International Committee on Computational Linguistics, Gyeongju, Republic of Korea, pp. 3539–3549. https://aclanthology.org/2022.coling-1.312.

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov (2020), Unsupervised cross-lingual representation learning at scale, *in* Jurafsky, Dan, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Online, pp. 8440–8451. https://aclanthology.org/2020.acl-main.747.

de Vries, Wietse, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim (2019), BERTje: A Dutch BERT model, *arXiv preprint arXiv:1912.09582*.

de Vries, Wietse, Martijn Bartelds, Malvina Nissim, and Martijn Wieling (2021), Adapting monolingual models: Data can be scarce when language similarity is high, *in* Zong, Chengqing, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, Association for Computational Linguistics, Online, pp. 4901–4907. https://aclanthology.org/2021.findings-acl.433.

de Vries, Wietse, Martijn Wieling, and Malvina Nissim (2023), Dumb: A benchmark for smart evaluation of dutch models, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Singapore.

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2020), RobBERT: a Dutch RoBERTa-based Language Model, *Findings of the Association for Computational Linguistics: EMNLP 2020*, Association for Computational Linguistics, Online, pp. 3255–3265. https://www.aclweb.org/anthology/2020.findings-emnlp.292.

Delobelle, Pieter, Thomas Winters, and Bettina Berendt (2022), Robbert-2022: Updating a dutch language model to account for evolving language use.

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

Dodge, Jesse, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner (2021), Documenting large webtext corpora: A case study on the colossal clean crawled corpus, *in* Moens, Marie-Francine, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, pp. 1286–1305. https://aclanthology.org/2021.emnlp-main.98.

Garcia, Xavier, Noah Constant, Ankur Parikh, and Orhan Firat (2021), Towards continual learning for multilingual machine translation via vocabulary substitution, *in* Toutanova, Kristina, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics, Online, pp. 1184–1192. https://aclanthology.org/2021.naacl-main.93.

Gogoulou, Evangelia, Ariel Ekgren, Tim Isbister, and Magnus Sahlgren (2022), Cross-lingual transfer of monolingual models, *in* Calzolari, Nicoletta, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, European Language Resources Association, Marseille, France, pp. 948–955. https://aclanthology.org/2022.lrec-1.100.

He, Pengcheng, Jianfeng Gao, and Weizhu Chen (2023), Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing.

Jansen, Tim, Yangling Tong, Victoria Zevallos, and Pedro Ortiz Suarez (2022), Perplexed by Quality: A Perplexity-based Method for Adult and Harmful Content Detection in Multilingual Heterogeneous Web Data, *arXiv e-prints* p. arXiv:2212.10440.

Limisiewicz, Tomasz, Jiří Balhar, and David Mareček (2023), Tokenization impacts multilingual language modeling: Assessing vocabulary allocation and overlap across languages.

Liu, Yinhan, 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*. https://arxiv.org/abs/1907.11692.

Martin, Louis, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de La Clergerie, Djamé Seddah, and Benoît Sagot (2019), Camembert: a tasty french language model, *arXiv preprint arXiv:1911.03894*.

NVIDIA, Applied Deep Learning Research team (2023), The Megatron-BERT-Cased-345M model. https://huggingface.co/nvidia/megatron-bert-cased-345m.

Ortiz Suárez, Pedro Javier, Benoît Sagot, and Laurent Romary (2019), Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures, *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*. https://hal.inria.fr/hal-02148693.

Remy, François, Pieter Delobelle, Bettina Berendt, Kris Demuynck, and Thomas Demeester (2023), Tik-to-tok: Translating language models one token at a time: An embedding initialization strategy for efficient language adaptation.

Sennrich, Rico, Barry Haddow, and Alexandra Birch (2016), Neural machine translation of rare words with subword units, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Berlin, Germany, pp. 1715–1725. https://aclanthology.org/P16-1162.

Thrush, Tristan and Muhtasham Muhtasham Oblokulov (2022), Online language modelling training pipeline. https://huggingface.co/olm.

Touvron, Hugo, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom (2023), Llama 2: Open foundation and fine-tuned chat models.

Zhao, Wayne Xin, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen (2023), A survey of large language models.

Appendix A. Hyperparameter space

Table 3: The hyperparameter space used for fine-tuning.

Hyperparameter	**Value**
adam_epsilon	10^{-8}
fp16	False
gradient_accumulation_steps	$i \in \{2, 4, 8, 16\}$
learning_rate	$[10^{-6}, 10^{-4}]$
max_grad_norm	1.0
max_steps	-1
num_train_epochs	3
per_device_eval_batch_size	8
per_device_train_batch_size	8
max_sequence_length.	512
seed	1
warmup_steps	0
weight_decay	$[0, 0.1]$