RobBERT: a Dutch RoBERTa-based Language Model

Pieter Delobelle[1] and **Thomas Winters**[1] and **Bettina Berendt**[1,2]
[1] Department of Computer Science, KU Leuven
[2] Faculty of Electrical Engineering and Computer Science, TU Berlin
firstname.lastname@kuleuven.be

Abstract

Pre-trained language models have been dominating the field of natural language processing in recent years, and have led to significant performance gains for various complex natural language tasks. One of the most prominent pre-trained language models is BERT, which was released as an English as well as a multilingual version. Although multilingual BERT performs well on many tasks, recent studies show that BERT models trained on a single language significantly outperform the multilingual version. Training a Dutch BERT model thus has a lot of potential for a wide range of Dutch NLP tasks. While previous approaches have used earlier implementations of BERT to train a Dutch version of BERT, we used RoBERTa, a robustly optimized BERT approach, to train a Dutch language model called RobBERT. We measured its performance on various tasks as well as the importance of the fine-tuning dataset size. We also evaluated the importance of language-specific tokenizers and the model's fairness. We found that RobBERT improves state-of-the-art results for various tasks, and especially significantly outperforms other models when dealing with smaller datasets. These results indicate that it is a powerful pre-trained model for a large variety of Dutch language tasks. The pre-trained and fine-tuned models are publicly available to support further downstream Dutch NLP applications.

1 Introduction

The advent of neural networks in natural language processing (NLP) has significantly improved state-of-the-art results within the field. Initially, recurrent neural networks and long short-term memory networks dominated the field. Later, the transformer model caused a revolution in NLP by dropping the recurrent part and only keeping attention mechanisms (Vaswani et al., 2017). The transformer model led to other popular language models, e.g. GPT-2 (Radford et al., 2018, 2019). BERT (Devlin et al., 2019) improved over previous models and recurrent networks by allowing the system to learn from input text in a bidirectional way, rather than only from left-to-right or the other way around. This model was later re-implemented, critically evaluated and improved in the RoBERTa model (Liu et al., 2019).

These large-scale attention-based models provide the advantage of being able to solve NLP tasks by having a common, expensive pre-training phase, followed by a smaller fine-tuning phase. The pre-training happens in an unsupervised way by providing large corpora of text in the desired language. The second phase only needs a relatively small annotated dataset for fine-tuning to outperform previous popular approaches in one of a large number of possible language tasks.

While language models are usually trained on English data, some multilingual models also exist. These are usually trained on a large quantity of text in different languages. For example, Multilingual-BERT is trained on a collection of corpora in 104 different languages (Devlin et al., 2019), and generalizes language components well across languages (Pires et al., 2019). However, models trained on data from one specific language usually improve the performance of multilingual models for this particular language (Martin et al., 2019; de Vries et al., 2019). Training a RoBERTa model (Liu et al., 2019) on a Dutch dataset thus also potentially increases performances for many downstream Dutch NLP tasks. In this paper, we introduce RobBERT[1], a Dutch RoBERTa-based pre-trained language model, and critically evaluate its performance on various language tasks against

[1]The model named itself RobBERT when it was prompted with *"Ik heet <mask>BERT."* (*"My name is <mask>BERT."*), which we found quite a suitable name.

other Dutch languages models. We also propose several new tasks for testing the model's zeroshot ability, evaluate its performance on smaller datasets, and for measuring the importance of a language-specific tokenizer. Finally, we provide an extensive fairness evaluation using recent techniques and a new translated dataset.

2 Related Work

Transformer models have been successfully used for a wide range of language tasks. Initially, transformers were introduced for use in machine translation, where they efficiently improved the state-of-the-art (Vaswani et al., 2017). This cornerstone was used in BERT, a transformer model obtaining state-of-the-art results for eleven natural language processing tasks, such as question answering and natural language inference (Devlin et al., 2019). BERT is pre-trained with large corpora of text using two unsupervised tasks. The first task is called masked language modeling (MLM), making the model guess which word is masked in certain position in the text. The second task is next sentence prediction, in which the model has to predict if two sentences occur subsequent in the corpus, or randomly sampled from the corpus. These tasks allow the model to create internal representations about a language, which could thereafter be reused for different language tasks. This architecture has been shown to be a general language model that could be fine-tuned with little data in a relatively efficient way for a very distinct range of tasks and still outperform previous architectures (Devlin et al., 2019).

Transformer models are also capable of generating contextualized word embeddings (Peters et al., 2018). Traditional word embeddings, e.g. word2vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), lack the capibility of differentiating words based on context (e.g. *"a stick"* versus *"let's stick to"*). Transformer models, like BERT, on the other hand automatically incorporate the context a word occurs into its embedding.

The attention mechanism in transformer encoder models also allows for better resolution of coreferences between words (Joshi et al., 2019a). For example, in the sentence *"The trophy doesnt fit in the suitcase because its too big."*, the word *"it"* would refer to the the suitcase instead of the trophy if the last word was changed to *"small"* (Levesque et al., 2012). Being able to resolve these coreferences is for example important for translation, as dependent words might change form, e.g. due to word gender.

While BERT has been shown to be a powerful language model, it also received scrutiny on its training and pre-processing. The authors of RoBERTa (Liu et al., 2019) showed that while the NSP pre-training task made the model perform better, it was not due to its intended reason, as it might merely predict relatedness between corpus sentences rather than subsequent sentences. That Devlin et al. (2019) trained a better model when using NSP than without NSP is likely due to the model learning long-range dependencies that were longer than when just using single sentences. As such, the RoBERTa model uses only the MLM task, and uses multiple full sentences in every input. Other researchers later improved the NSP task by instead making the model predict for two subsequent sentences if they occur in the given or flipped order in the corpus (Lan et al., 2019).

Devlin et al. (2019) also presented a multilingual model (mBERT) with the same architecture as BERT, but trained on Wikipedia corpora in 104 languages. Unfortunately, the quality of these multilingual embeddings is considered worse than their monolingual counterparts, as Rönnqvist et al. (2019) illustrated for German and English models in a generative setting. The monolingual French CamemBERT model (Martin et al., 2019) also outperformed mBERT on all tasks. Brandsen et al. (2019) also outperformed mBERT on several Dutch tasks using their Dutch BERT-based language model, called BERT-NL, trained on the small SoNaR corpus (Oostdijk et al., 2013a). More recently, de Vries et al. (2019) also showed similar results for Dutch using their BERTje model, outperforming multilingual BERT in a wide range of tasks, such as sentiment analysis and part-of-speech tagging by pre-training on multiple corpora. Since both these works are concurrent with ours, we compare our results with BERTje and BERT-NL in this paper.

3 Pre-training RobBERT

We pre-trained RobBERT using the RoBERTa training regime. We trained two different versions, one where only the pre-training corpus was replaced with a Dutch corpus *(RobBERT v1)* and one where both the corpus and the tokenizer were replaced with Dutch versions *(RobBERT v2)*. These

two versions allow to evaluate the importance of having a language-specific tokenizer.

3.1 Data

We pre-trained our model on the Dutch section of the OSCAR corpus, a large multilingual corpus which was obtained by language classification in the Common Crawl corpus (Ortiz Suárez et al., 2019). This Dutch corpus is 39GB large, with 6.6 billion words spread over 126 million lines of text, where each line could contain multiple sentences. This corpus is thus much larger than the corpora used for similar Dutch BERT models, as BERTje used a 12GB corpus, and BERT-NL used the SoNaR-500 corpus (about 2.2GB). (de Vries et al., 2019; Brandsen et al., 2019).

3.2 Tokenizer

For RobBERT v2, we changed the default byte pair encoding (BPE) tokenizer of RoBERTa to a Dutch tokenizer. The vocabulary of the Dutch tokenizer was constructed using the Dutch section of the OSCAR corpus (Ortiz Suárez et al., 2019) with the same byte-level BPE algorithm as RoBERTa (Liu et al., 2019). This tokenizer gradually builds its vocabulary by replacing the most common consecutive tokens with a new, merged token. We limited the vocabulary to 40k words, which is 10k words less than RobBERT v1, due to additional tokens including non-negligible number of Unicode tokens that are not used in Dutch. These are likely caused due to misclassified sentences during the creation of the OSCAR corpus (Ortiz Suárez et al., 2019).

3.3 Training

RobBERT shares its architecture with RoBERTa's base model, which itself is a replication and improvement over BERT (Liu et al., 2019). Like BERT, it's architecture consists of 12 self-attention layers with 12 heads (Devlin et al., 2019) with 117M trainable parameters. One difference with the original BERT model is due to the different pre-training task specified by RoBERTa, using only the MLM task and not the NSP task. During pre-training, it thus only predicts which words are masked in certain positions of given sentences. The training process uses the Adam optimizer (Kingma and Ba, 2017) with polynomial decay of the learning rate $l_r = 10^{-6}$ and a ramp-up period of 1000 iterations, with hyperparameters $\beta_1 = 0.9$ and RoBERTa's default $\beta_2 = 0.98$. Additionally, a weight decay of 0.1 and a small dropout of 0.1 helps prevent the model from overfitting (Srivastava et al., 2014).

RobBERT was trained on a computing cluster with 4 Nvidia P100 GPUs per node, where the number of nodes was dynamically adjusted while keeping a fixed batch size of 8192 sentences. At most 20 nodes were used (i.e. 80 GPUs), and the median was 5 nodes. By using gradient accumulation, the batch size could be set independently of the number of GPUs available, in order to maximally utilize the cluster. Using the Fairseq library (Ott et al., 2019), the model trained for two epochs, which equals over 16k batches in total, which took about three days on the computing cluster. In between training jobs on the computing cluster, 2 Nvidia 1080 Ti's also covered some parameter updates for RobBERT v2.

4 Evaluation

We evaluated RobBERT on multiple downstream Dutch language tasks. For testing text classification, we evaluate on sentiment analysis and on demonstrative and relative pronoun prediction. The latter task helps evaluating the zero-shot prediction abilities, i.e. using only the pre-trained model without any fine-tuning. Both classification tasks are also used to measure how well RobBERT performs on smaller datasets, by only using subsets of the data. For testing RobBERT's token tagging capabilities, we used both part-of-speech (POS) tagging and named entity recognition (NER) tasks.

4.1 Sentiment Analysis

We replicated the high-level sentiment analysis task used to evaluate BERT-NL (Brandsen et al., 2019) and BERTje (de Vries et al., 2019) to be able to compare our methods. This task uses a dataset called Dutch Book Reviews dataset (DBRD), in which book reviews from `hebban.nl` are labeled as positive or negative (van der Burgh and Verberne, 2019). Although the dataset contains 118,516 reviews, only 22,252 of these reviews are actually labeled as positive or negative, which are split in a 90% train and 10% test datasets. This dataset was released in a paper analysing the performance of an ULMFiT model (Universal Language Model Fine-tuning for Text Classification model) (van der Burgh and Verberne, 2019).

We fine-tuned RobBERT on the first 10,000

Table 1: Results of RobBERT fine-tuned on several downstream classification tasks, compared to the state of the art models for the tasks. For accuracy, we also report the 95% confidence intervals. *(Results annotated with * from van der Burgh and Verberne (2019), ** from de Vries et al. (2019), *** from Brandsen et al. (2019), **** from Allein et al. (2020))*

	10k		Full dataset	
Task + model	ACC (95% CI) [%]	F1 [%]	ACC (95% CI) [%]	F1 [%]
Sentiment Analysis (DBRD)				
van der Burgh and Verberne (2019)	—	—	93.8*	—
BERTje (de Vries et al., 2019)	—	—	93.0**	—
BERT-NL (Brandsen et al., 2019)	—	—	—	84.0***
RobBERT v1	86.730 (85.32, 88.14)	86.729	94.422 (93.47,95.38)	94.422
RobBERT v2	**94.379** (93.42, 95.33)	**94.378**	**95.144** (94.25,96.04)	**95.144**
Die/Dat (Europarl)				
Baseline (Allein et al., 2020)	—	—	75.03****	—
mBERT (Devlin et al., 2019)	92.157 (92.06, 92.25)	90.898	98.285 (98.24,98.33)	98.033
BERTje (de Vries et al., 2019)	93.096 (92.84, 93.36)	91.279	98.268 (98.22,98.31)	98.014
RobBERT v1	97.006 (96.95, 97.07)	96.571	98.406 (98.36, 98.45)	98.169
RobBERT v2	**97.816** (97.76, 97.87)	**97.514**	**99.232** (99.20, 99.26)	**99.121**

training examples as well as on the full dataset. While the ULMFiT model is first fine-tuned using the unlabeled reviews before training the classifier (van der Burgh and Verberne, 2019), it is unclear whether the other BERT models utilized the unlabeled reviews for further pre-training (Sun et al., 2019) or only used the labeled data for fine-tuning the pre-trained model. We did the latter, meaning further improvement is possible by additionally pre-training on unlabeled in-domain sequences. Another unknown is how these models dealt with reviews that were longer than the maximum number of tokens, as the average book review length is 547 tokens, with 40% of the documents being longer than our model could handle. For our experiments, we only gave the last tokens of a review as input, as we found the training performance to be better, likely due to containing a summarizing comments. We trained our model for 2000 iterations with a batch size of 128 and a warm-up of 500 iterations, reaching a learning rate of 10^{-5}. The training took approx. 2 hours on 2 Nvidea 1080 Ti GPUs, the best-performing RobBERT v2 model was selected based on a validation accuracy of 0.994. We see that RobBERT outperforms the other BERT models. Both versions of RobBERT also outperform the state-of-the-art ULMFiT model, although the difference is only statistically significant for RobBERT v2.

4.2 Die/Dat Disambiguation

Since BERT models perform well on coreference resolution tasks (Joshi et al., 2019b), we propose to evaluate RobBERT on the recently introduced *"die/dat disambiguation"* task (Allein et al., 2020), as a novel way to evaluate the zero-shot ability of Dutch BERT models. In Dutch, depending on the sentence, both *"die"* and *"dat"* can be either demonstrative or relative pronouns; in addition they can also be used in a subordinating conjunction, i.e. to introduce a clause. The use of either of these words depends on the gender of the word it refers to. Allein et al. (2020) presented multiple models trained on the Europarl (Koehn, 2005) and SoNaR corpora (Oostdijk et al., 2013b), achieving an accuracy of 75.03% on Europarl to 84.56% on SoNaR.

For this task, we use the Dutch Europarl corpus (Koehn, 2005), with the first 1.3M sequences (`head`) for training and last 399k (`tail`) as test set. Every sequence containing *"die"* or *"dat"* creates an example for every occurrence of either word by masking the occurrence. For the test set, this resulted in about 289k masked sentences.

BERT-like models can solve this task using two different approaches. Since the task is about predicting words, their default MLM task can be used to guess which of the two words is more probable in a particular masked position. This allows the comparison of zero-shot BERT models, i.e. without any fine-tuning on the training data (Table 2).

The second approach uses the masked sentences to create two versions by filling the mask with either "die" and "dat", separate them using the `[SEP]` token and making the model predict which of the two sentences is correct. This fine-tuning was performed using 4 Nvidia GTX 1080 Ti GPUs, taking 30 minutes for 13 epochs on 10k sequences and about 24 hours for 3 epochs on the full dataset. We did no hyperparameter tuning, as the initial hyperparameters ($l_r = 10^{-5}, \epsilon = 10^{-9}$, warm-up of 250 steps, batch size of 32 (10k) or 128 (full dataset), dropout of 0.1) were satisfactory.

To measure RobBERTs performance on smaller datasets, we trained the model twice for both the sentiment analysis task and the *die/dat* disambiguation task, once with a subset of 10k utterances, and once with the full training dataset.

Table 2: Performance of predicting *die/dat* as most likely candidate for a mask using zero-shot BERT models (i.e. without fine-tuning) as well as a majority class predictor (ZeroR), tested on the 288,799 test set sentences

Model	**Accuracy [%]**
ZeroR (majority class)	66.70
mBERT (Devlin et al., 2019)	90.21
BERTje (de Vries et al., 2019)	94.94
RobBERT v1	98.03
RobBERT v2	**98.75**

RobBERT outperforms previous models as well as other BERT models both with as well as without fine-tuning (see Table 1 and Table 2). It is also able to reach similar performance using less data. The fact that both for the fine-tuned and the zero-shot setting, RobBERT outperforms other BERT models is also an indication that the base model has internalised more knowledge about Dutch than the others, likely due to the improved pre-training regime and using a larger corpus. We can also see that having a Dutch tokenizer strongly helps reduce the error rate for this task, halving the error rate when fine-tuned on the full dataset. The reason the BERT-based models outperform the previous RNN-based approach is likely the encoders ability to better deal with coreference resolution (Joshi et al., 2019a), and by extension deciding which word the *"die"* or *"dat"* belongs to. The fact that RobBERT strongly outperforms the other BERT models on subsets of the data indicates that it is a suitable candidate for Dutch tasks that only have limited data available.

4.3 Part-of-speech Tagging

Part-of-speech (POS) tagging involves labeling tokens rather than labeling sequences. For this, we used a different head with an classification output for each token, all activated by a softmax function. When a word consists of multiple tokens, the first token is used for the the label of the word.

We perform the same POS fine-tuning regimes as RoBERTa (Liu et al., 2019) to evaluate RobBERT's performance. When fine-tuning, we employ a linearly decaying learning rate with a warm-up for 6% of the total optimisation steps (Liu et al., 2019). For all the encoder-based models in our evaluation, we also perform a limited hyperparameter search on the development set with learning rate $l_r \in \{10^{-5}, 2 \cdot 10^{-5}, 3 \cdot 10^{e-5}, 10^{-4}\}$ and batch size $\in \{16, 32, 48\}$, which is also based on RoBERTa's fine-tuning.

Table 3: POS tagging on Lassy UD. For accuracy, we also report the 95% confidence intervals.

Task + model	ACC (95% CI) [%]
Frog (Bosch et al., 2007)	91.7 (91.2, 92.2)
mBERT (Devlin et al., 2019)	**96.5** (96.2, 96.9)
BERTje (de Vries et al., 2019)	96.3 (96.0, 96.7)
RobBERT v1	96.4 (96.0, 96.7)
RobBERT v2	96.4 (96.0, 96.7)

To evaluate the POS-performance, we used the Universal Dependencies (UD) version of the Lassy dataset (Van Noord et al., 2013), containing 17 different POS tags. We compared its performance with Frog, a popular memory-based Dutch POS tagging approach, and with other BERT models. Surprisingly, multilingual BERT marginally outperformed both Dutch BERT models, although not statistically significantly, with both RobBERT models in second place with an almost equal accuracy. The higher performance of multilingual BERT could be indicative that it benefits from transferable language structures from other languages helping it to perform well for POS tagging. Alternatively, this could signal a limit of the UD Lassy dataset, or at least for the performance of BERT-like models on this dataset.

We also evaluated the models on several smaller subsets of the training data, to illustrate how much data is needed to achieve acceptable results. For all models, the same hyperparameters obtained for

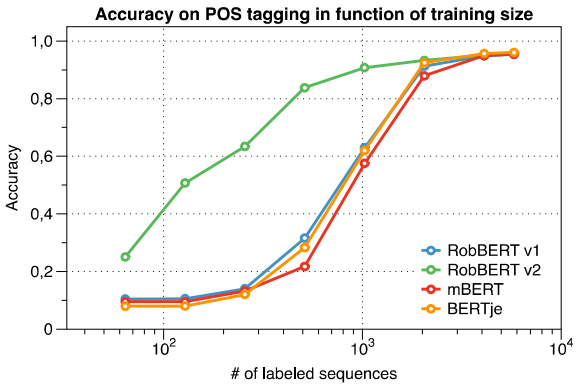

Figure 1: POS tagging accuracy on the test set for different sizes of training sets.

Table 3 are used for all subsets, under the assumption that using a subset of the training data also works well under the same hyperparameters. The hyperparameters which yielded the results of RobBERT v2 are $l_r = 10^{-4}$, batch size of 16 and dropout of 0.1. The separate development set was used to select the best-performing model after each epoch based , which had a cross-entropy loss of 0.172 on the development set. While all BERT models perform similarly after seeing all instances of the UD Lassy dataset, there is a clear difference when using smaller training sets (Figure 1). RobBERT v2 outperforms all other models when using only 1,000 data points or less, again showing that it is more capable of dealing with smaller datasets.

4.4 Named Entity Recognition

Named entity recognition (NER) is the task of labeling named entities in a sentence. It is thus a token-level task, just like POS-tagging, meaning we can use the same setup and hyperparameter tuning as described in Subsection 4.3. We use the CoNLL-2002 dataset and evaluation script[2], which use a four value BIO labeling, namely for organisations, locations, people and miscellaneous (Tjong Kim Sang, 2002). The hyperparameters yielding the results for RobBERT v2 are $l_r = 3 \cdot 10^{-5}$, batch size of 32 and dropout of 0.1. The separate development set was used to select the best-performing model after each epoch. As the F_1 score is generally used for evaluation of this task, we opted to use this metric instead of cross-entropy loss for selecting the best-performing model, which had an F_1 score of 0.8769 on the development set. We compared the F_1 scores on the NER task in Table 4.

[2]Retrieved from https://www.clips.uantwerpen.be/conll2002/ner/

Table 4: NER for various models, F_1 score calculated with the CoNLL 2002 evaluation script, except for † which used the Seqeval Python library, * from Wu and Dredze (2019), ** from Brandsen et al. (2019), *** from de Vries et al. (2019).

Task + model	F_1 score [%]
Frog (Bosch et al., 2007)	57.31
mBERT (Devlin et al., 2019)	84.19
mBERT (Wu and Dredze, 2019)	**90.94***
BERT-NL (Brandsen et al., 2019)	89.7†**
BERTje (de Vries et al., 2019)	88.3***
RobBERT v1	87.53
RobBERT v2	89.08

We can see that (Wu and Dredze, 2019) outperforms all other BERT models using a multilingual BERT model with an F_1 score of 90.94. When we used the token labeling fine-tuning regime described earlier on multilingual BERT, we were only able to get to an F_1 score of 84.19 using multilingual BERT, thus being outperformed by the Dutch BERT models. One possibility is that the authors used a more optimal fine-tuning regime, or that they trained their model longer.

5 RobBERT and Fairness

As language models are trained on large corpora, this poses a risk that minorities and protected groups are ill-represented, e.g. by encoding stereotypes (Bolukbasi et al., 2016; Zhao et al., 2019; Gonen and Goldberg, 2019). In word embeddings, these studies often rely on analogies (Bolukbasi et al., 2016; Caliskan et al., 2017) or embedding analysis (Gonen and Goldberg, 2019). These approaches are not directly transferable to BERT models, since the sentence the word occurs in influences its embedding.

Efforts to generalize these approaches often rely on templates (May et al., 2019; Kurita et al., 2019). These can be intentionally neutral ("<mask> *is a word*") or they might resemble an analogy in textual form ("<mask> *is a zookeeper.*"). One can then perform an association test between possible values for the <mask> slot, similar to a word embedding association test (Caliskan et al., 2017).

In this section, we discuss two distinct potential origins of representational harm (Blodgett et al., 2020) a language model could exhibit, and evaluate these on RobBERT v2. The two discussed behaviours are (i) stereotyping of gender roles in

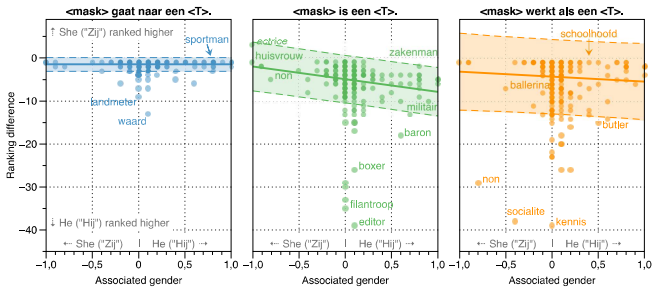

Figure 2: Ranking difference between gendered pronouns for various professions. Three templates were used to evaluate, were <T> is replaced by a profession. In the leftmost template, the pronoun and profession refer to different entities.

occupations and (ii) unequal predictive power for texts written by men and women. These exemplifications highlight how language models risk affecting the experience of the end user, or replicating and reinforcing stereotypes.

5.1 Gender Stereotyping

To assess how gender stereotypes of professions are present, we performed a template-based association test similar to Kurita et al. (2019) and the *semantically unbleached* templates of May et al. (2019). We used RobBERT's LM head—trained during pre-training with the MLM task—to fill in the <mask> slot for each template, in the same manner as the zero-shot task described in Subsection 4.2. These templates have a second slot, which is used to iterate over the professions.

For this list of professions and the gender projection on the *he-she* axis, we base us on the work by Bolukbasi et al. (2016), who crowdsourced the associated gender for various professions. Ideally, we would use a similarly crowdsourced Dutch dataset. However, since this does not yet exist, we opted for manually translating these English professions using the guidelines established by the European Parliament for gender neutral professions (Dimitrios Papadimoulis, 2018), meaning that we opted for the inclusive form for neutral professions in English that do not have a neutral counterpart, but an inclusive binary male variant and a female variant with explicit gender (e.g. for lawyer: using *"advocaat"* and not *"advocate"*). In the rare case that an inclusive or neutral form translated to an exclusive binary form, we excluded this profession.

We evaluated three templates on RobBERT, including one control template without co-referent entities ("<mask> *goes to a* <T>") (Figure 2). For the control template, there should be no and indeed is no correlation between ranking difference for both pronouns and the associated gender of a profession. Interestingly, none of the instances has a positive ranking difference, meaning the language model always ranks the male pronoun as more likely.

When the profession and <mask> slot refer to the same entity, the general assessment of the language model correlates with the associated gender. But again, RobBERT estimates that the male pronoun is more likely in almost all cases, even when these professions have a gendered suffix. Curiously, actress ("*actrice*") is the only word where this is not the case. Since RobBERT estimates the male pronoun to be more likely even in the control template, we suspect that the effect is due to more coverage of men in the training corpus.

5.2 Unequal Predictive Performance

Unfairness is particularly problematic if it leads to unequal predictive performance. This problem has been demonstrated for decision support systems, including recidivism prediction (Angwin et al., 2016) and public employment services (Allhutter et al., 2020). Such predictions can be downstream tasks of language understanding; for example when job resums are processed (Van Hautte et al., 2020).

To review fairness in downstream tasks, we evaluated the sentiment analysis task on DBRD, a dataset with scraped book reviews. Although this task in itself may have low impact for end users, it still serves as an illustrative example of how fine-tuned models can behave unfairly.

To investigate whether such bias might result for our fine-tuned model, we analyzed its outcome for different values of a sensitive attribute (in this case gender), as is commonly done in fair machine learning research (Zemel et al., 2013; Hardt et al., 2016; Delobelle et al., 2020). To this end, we augmented the held-out test set of DBRD with gender as a sensitive attribute for each review[3]. Values were obtained from the reviews' author profiles with a self-reported binary gender ('*man*' or '*vrouw*') (64%). The remaining 36% of reviews did not report author gender, and they were discarded for this evaluation. Of the remain-

[3]We make this augmentation of DBRD available under CC-by-NC-SA at https://people.cs.kuleuven.be/~pieter.delobelle/data.html.

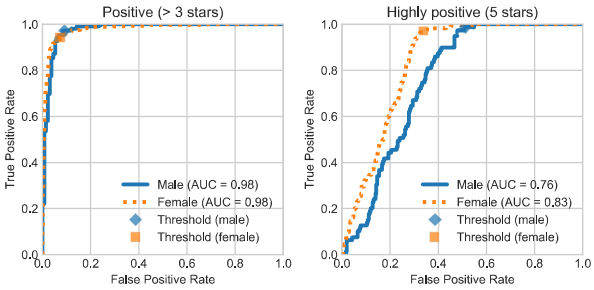

Figure 3: ROC of the fine-tuned model to predict positive reviews for male and female reviewers

ing, gender-labelled, reviews, 76% were written by women. Thus, the dataset is unbalanced.

We quantify the gender difference with two metrics: (i) Demographic Parity Ratio (DPR), which expresses a relative difference between predicted outcomes $\hat{y}$ conditioned on the sensitive attribute a (Dwork et al., 2012), following

$$\frac{P(\hat{y} \mid \neg a)}{P(\hat{y} \mid a)},$$

and (ii) Equal Opportunity (EO) Hardt et al. (2016), which in addition also conditions on the true outcome y, as a task-specific fairness measure (Dwork et al., 2012), following

$$P(\hat{y} \mid \neg a, y) - P(\hat{y} \mid a, y).$$

Hardt et al. (2016) also relate EO to the ROC curves to evaluate fairness when dealing with a binary predictor and a score function. To derive a binary predictor, we used 0 as a threshold value. Figure 3 shows the single resulting predictor, with the ROC curves split on the sensitive attribute, for each of the two rating levels (over 3 resp. 5 stars).

The results of Figure 3 show that there is small difference in opportunity, which is especially pronounced for the highly positive classifier. For positive reviews, the EO difference is 0.028 at the indicated threshold and DPR is 70.2%. The DPR would indicate an unfairness, as values below 80% are often considered unfair. However, this metric has received some criticism, and when including the true outcome in EO, the difference in probabilities is close to 0, which does not signal any unfairness. When taking into account the ROC curves (Figure 3), the EO score can be explained by the good predictive performance. When considering highly positive reviews, however, the differences become more pronounced and the model has better predictive performance for reviews written by women.

6 Code

The training and evaluation code of this paper as well as the RobBERT model and the fine-tuned models are publicly available for download at https://github.com/iPieter/RobBERT.

7 Limitations and Future Work

There are several potential improvements for creating a better pre-trained RobBERT-like model. First, since BERT-based models are still being actively researched, one could potentially improve the training regime using new unsupervised pre-training tasks when they are discovered, e.g. sentence order prediction (Lan et al., 2019). Second, while RobBERT is trained on lines that contain multiple sentences, it does not put subsequent lines of the corpus after each other due to the shuffled nature of the OSCAR corpus (Ortiz Suárez et al., 2019). This is unlike RoBERTa, which does put full sentences next to one another if they do not exceed the available sequence length, in order to learn the long-range dependencies between words that the original BERT learned using its controversial NSP task. Creating an unshuffled version of OSCAR might thus further improve the performance of the pre-trained model. Third, there might be some benefit to modifying the tokenizer to use morpheme-based tokens, as Dutch uses compound words. Fourth, one could improve model's fairness during pre-training. We illustrated how representational harm in downstream tasks can affect the end user's experience, like the unequal predictive performance for the DBRD task. Various methods have been proposed to mitigate unfair behaviour in AI models (Zemel et al., 2013; Delobelle et al., 2020). While we refrained from training fair pre-trained and fine-tuned models in this research, training such models could be an interesting contribution. In addition, with the increased attention on fairness in machine learning, a broader view of the impact on other protected groups due to large pre-trained language models is also called-for.

The RobBERT model itself can be used in new settings to help future research. First, RobBERT could be used in a model that uses a BERT-like transformer stack for the encoder and a generative model as a decoder (Raffel et al., 2019; Lewis

et al., 2019) Second, RobBERT can serve as the basis for a large number of Dutch language tasks that we did not examine in this paper. Given RobBERT's state-of-the-art performance on small as well as on large datasets, it could help advance results when fine-tuned on new datasets.

8 Conclusion

We introduced a new language model for Dutch based on RoBERTa, called RobBERT, and showed that it outperforms earlier approaches as well as other BERT-based language models for a several different Dutch language tasks. More specifically, we found that RobBERT significantly outperformed other BERT-like models when dealing with smaller datasets, making it a useful resource for a large range of application domains. We expect this model to serve as a base for fine-tuning on other tasks, and thus help foster new models that can advance results for Dutch language tasks.

Acknowledgements

Pieter Delobelle was supported by the Research Foundation - Flanders under EOS No. 30992574 and received funding from the Flemish Government under the Onderzoeksprogramma Artificile Intelligentie (AI) Vlaanderen programme. Thomas Winters is a fellow of the Research Foundation-Flanders (FWO-Vlaanderen). Most computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation - Flanders (FWO) and the Flemish Government department EWI. We are especially grateful to Luc De Raedt for his guidance as well as for providing the facilities to complete this project. We are thankful to Liesbeth Allein and her supervisors for inspiring us to use the *die/dat* task. We are also grateful to Ott et al. (2019); Paszke et al. (2019); Haghighi et al. (2018); Wolf et al. (2019) for their software packages.

References

Liesbeth Allein, Artuur Leeuwenberg, and Marie-Francine Moens. 2020. Binary and Multitask Classification Model for Dutch Anaphora Resolution: Die/Dat Prediction. *arXiv:2001.02943 [cs]*.

Doris Allhutter, Florian Cech, Fabian Fischer, Gabriel Grill, and Astrid Mager. 2020. Algorithmic Profiling of Job Seekers in Austria: How Austerity Politics Are Made Effective. *Front. Big Data*, 3:5.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. *arXiv:2005.14050 [cs]*.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357.

Antal van den Bosch, Bertjan Busser, Sander Canisius, and Walter Daelemans. 2007. An efficient memory-based morphosyntactic tagger and parser for dutch. *LOT Occasional Series*, 7:191–206.

Alex Brandsen, 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.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Wietse de Vries, Andreas van Cranenburgh, Arianna Bisazza, Tommaso Caselli, Gertjan van Noord, and Malvina Nissim. 2019. BERTje: A Dutch BERT Model. *arXiv:1912.09582 [cs]*.

Pieter Delobelle, Paul Temple, Gilles Perrouin, Benoît Frénay, Patrick Heymans, and Bettina Berendt. 2020. Ethical Adversaries: Towards Mitigating Unfairness with Adversarial Machine Learning. *arXiv:2005.06852 [cs, stat]*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dimitrios Papadimoulis. 2018. Genderneutraal taalgebruik in het Europees Parlement. Technical report, European Parlement.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness Through Awareness. In *3rd Innovations in Theoretical Computer Science Conference*, pages 214–226. ACM.

Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *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), pages 609–614, Minneapolis, Minnesota. Association for Computational Linguistics.

Sepand Haghighi, Masoomeh Jasemi, Shaahin Hessabi, and Alireza Zolanvari. 2018. PyCM: Multiclass confusion matrix library in Python. *Journal of Open Source Software*, 3(25):729.

Moritz Hardt, Eric Price, ecprice, and Nati Srebro. 2016. Equality of Opportunity in Supervised Learning. In *NIPS*, pages 3315–3323. Curran Associates.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2019a. Spanbert: Improving pre-training by representing and predicting spans.

Mandar Joshi, Omer Levy, Daniel S Weld, and Luke Zettlemoyer. 2019b. Bert for coreference resolution: Baselines and analysis. *arXiv preprint arXiv:1908.09091*.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit*, volume 5, pages 79–86. [object Object].

Keita Kurita, Nidhi Vyas, Ayush Pareek, Alan W Black, and Yulia Tsvetkov. 2019. Measuring bias in contextualized word representations. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 166–172, Florence, Italy. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension.

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:1907.11692 [cs]*.

Louis Martin, 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:1911.03894 [cs]*.

Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On Measuring Social Biases in Sentence Encoders. *arXiv:1903.10561 [cs]*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Nelleke Oostdijk, Martin Reynaert, Véronique Hoste, and Ineke Schuurman. 2013a. The construction of a 500-million-word reference corpus of contemporary written dutch. In *Essential speech and language technology for Dutch*, pages 219–247. Springer.

Nelleke Oostdijk, Martin Reynaert, Véronique Hoste, and Ineke Schuurman. 2013b. The construction of a 500-million-word reference corpus of contemporary written Dutch. In *Essential Speech and Language Technology for Dutch: Results by the STEVIN-Programme*, chapter 13. Springer Verlag.

Pedro Javier Ortiz Suárez, Benoît Sagot, and Laurent Romary. 2019. Asynchronous Pipeline for Processing Huge Corpora on Medium to Low Resource Infrastructures. In *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*, Cardiff, United Kingdom.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. Fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv:1910.10683 [cs, stat]*.

Samuel Rönnqvist, Jenna Kanerva, Tapio Salakoski, and Filip Ginter. 2019. Is multilingual BERT fluent in language generation? In *Proceedings of the First NLPL Workshop on Deep Learning for Natural Language Processing*, pages 29–36, Turku, Finland. Linköping University Electronic Press.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.

Erik F. Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning - Volume 20*, COLING-02, page 14, USA. Association for Computational Linguistics.

Benjamin van der Burgh and Suzan Verberne. 2019. The merits of Universal Language Model Fine-tuning for Small Datasets – a case with Dutch book reviews. *arXiv:1910.00896 [cs]*.

Jeroen Van Hautte, Vincent Schelstraete, and Mikaël Wornoo. 2020. Leveraging the inherent hierarchy of vacancy titles for automated job ontology expansion. In *Proceedings of the 6th International Workshop on Computational Terminology*, pages 37–42, Marseille, France. European Language Resources Association.

Gertjan Van Noord, Gosse Bouma, Frank Van Eynde, Daniel De Kok, Jelmer Van der Linde, Ineke Schuurman, Erik Tjong Kim Sang, and Vincent Vandeghinste. 2013. *Large scale syntactic annotation of written Dutch: Lassy*, pages 147–164. Springer.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Shijie Wu and Mark Dredze. 2019. Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT. 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 833–844, Hong Kong, China. Association for Computational Linguistics.

Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. 2019. Gender Bias in Contextualized Word Embeddings. *arXiv:1904.03310 [cs]*.