Journal of Big Data

RESEARCH Open Access

Pre-trained transformer-based language models for Sundanese

Wilson Wongso*, Henry Lucky and Derwin Suhartono

*Correspondence:
wilson.wongso001@binus.ac.id
Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta 11840, Indonesia

Abstract

The Sundanese language has over 32 million speakers worldwide, but the language has reaped little to no benefits from the recent advances in natural language understanding. Like other low-resource languages, the only alternative is to fine-tune existing multilingual models. In this paper, we pre-trained three monolingual Transformer-based language models on Sundanese data. When evaluated on a downstream text classification task, we found that most of our monolingual models outperformed larger multilingual models despite the smaller overall pre-training data. In the subsequent analyses, our models benefited strongly from the Sundanese pre-training corpus size and do not exhibit socially biased behavior. We released our models for other researchers and practitioners to use.

Keywords: Sundanese Language, Transformers, Natural Language Understanding, Low-resource Language

Introduction

Recently, there has been a multitude of advances in the field of natural language, especially since the release of the Transformer [1] architecture. Before Transformers, RNNs (Recurrent neural networks) [2] like LSTMs (Long short-term memory networks) [3], and GRUs (Gated recurrent units) [4] were prevalent due to their network architecture design that suited the domain of natural language in the form of texts. However, most of these older architectures struggled to model long-term sequential dependency and are often slow to train due to the inability of parallelization.

On the other hand, the Transformer solves this issue by being parallelizable during training and extensively applying the attention mechanism to better model long sequences of inputs. Several modern architectures were then derived from the original Transformer model, such as the OpenAI GPT (Generative Pre-Training) [5], BERT (Bidirectional Encoder Representations from Transformers) [6], and subsequently variants like GPT-2 [7], GPT-3 [8], and RoBERTa (Robustly Optimized BERT Pre-training Approach) [9].

For instance, the GPT architecture stacks decoder blocks of the original Transformer architecture, while BERT stacks encoder blocks of the original Transformer and modifies them to become naturally bidirectional by design. Due to this discrepancy, GPT-based

models are normally pre-trained as causal/generative language models, whereas BERT-based models are pre-trained as masked language models. Once these models have been pre-trained, they can similarly be fine-tuned to downstream natural language understanding tasks. This scheme allowed for breakthroughs in tasks like text classification, question-answering, natural language inference, among many others.

However, most of these recent successes occur in the domain of high-resource languages like English, Chinese, and Spanish where data are abundant. Low-resource languages, conversely, have reaped little benefits despite the recent advances; attributing to the lack of data and existing work [10]. The best alternative is to fine-tune a multilingual model whose corpus contains that specific low-resource language. However, previous studies [11, 12] showed that multilingual models often perform poorly compared to monolingual models of low-resource languages.

To that, we propose the pre-training of various Transformer-based language models on a low-resource language, namely Sundanese. The Sundanese language is Indonesia's third-most spoken language [13], with over 32 million speakers worldwide and is the 52nd most spoken language in the world [14]. It is the official language in larger regions of Banten and West Java, as well as a minority language in various parts of Jakarta, Lampung, Central Java, and other nearby provinces in Indonesia [15]. Linguists usually classify the language into as many as 8 dialects [16], with the Priangan dialect being the most prominent of them all.

Moreover, a previous study [17] showed that the Sundanese language remains the main communication tool for adults and parents living in those regions. In the same paper, the authors concluded that there is a desire to preserve the Sundanese language in the fast-growing modern technological world as a cultural identity of the people. Another study [18] similarly noted that a great deal of Indonesian social media activity in platforms like Twitter originates from West Java, where the Sundanese language is most often used. With that many speakers and a huge market, it would be beneficial to have Sundanese language models that could be applied to various business processes.

Contrarily, there is very limited Sundanese corpus to be used for pre-training. Hence, on top of the pre-existing multilingual corpora like OSCAR (Open Super-large Crawled Aggregated Corpus) [19], CC-100 [20, 21], and C4 [22], Sundanese documents from Wikipedia were also used during the pre-training of our language models to cater to the scarcity of data availability.

Having been pre-trained, these models were thus fine-tuned to a downstream task of classifying emotions from tweets and were subsequently benchmarked against existing traditional machine learning algorithms and multilingual Transformer models such as the multilingual BERT [6] and the RoBERTa variant of XLM (Cross-lingual Language Model Pre-training) [20].

Once these models have been fully pre-trained and fine-tuned, they are subsequently released in the HuggingFace Transformers Model Hub[1], in hopes of encouraging other researchers to work on the field of Sundanese language modeling with open access to our models.

[1] https://hf.co/models?filter=su.

In the following sections, we will continue by discussing the background of Sundanese language modeling in greater detail, followed by the methodology used and experiments performed during the pre-training of our models, and finally, various analyses, discussions, and conclusions thereafter.

Related works

Transformer architectures

Before the advent of deep learning [23] and Transformers [1] following thereafter, natural language is one domain that computers have always struggled to model. This underlying issue prevails due to the fact that the rules of natural languages are often complex and vary from one language to another. Language modeling by means of traditional methods like statistical or rule-based models would often crumble when new, unseen inputs are fed into these models, given their inability to generalize.

On the contrary, deep learning models need not be taught the rules of natural language explicitly. Instead, by using methods like supervised learning, these models could gain an understanding of sequences of texts given the ground truth/label. A few of the earliest deep learning architectures developed to learn sequences of inputs include Recurrent neural networks [2], Long short-term memory networks [3], and Gated Recurrent Units [4]. Unlike feed-forward neural network architectures, a recurrent neural network has temporal information passed over time by considering the output of previous time steps.

However, these temporal models are still relatively inefficient to train despite the sophisticated modifications because of their sequential behavior by design. Transformers [1], on the other hand, completely remove the idea of recurrence and fully substitute it with the attention mechanism given by the following equation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \tag{1}$$

where Q, K, and V represent the query, key, and value vectors respectively. They are not only effective to capture relations between words, but are also efficient to train on accelerators like GPUs (Graphical Processing Units) and recently, TPUs (Tensor Processing Units).

Following the release and prominence of Transformers [1], OpenAI then developed a language model that builds on top of the original Transformer architecture, as well as extending the training schemes proposed in ULMFiT (Universal Language Model Fine-Tuning) [24] and ELMo (Embeddings from Language Models) [25]. Instead of utilizing LSTMs [3] as found in ULMFiT [24], the proposed OpenAI GPT model applies the decoder blocks of the Transformer architecture. Moreover, the learning process starts by first pre-training the model with the objective of generating the next word in a sequence, given a current prompt. This way, the model could learn the context of words in a given sequence before being tasked to solve downstream problems. This pre-training objective also greatly leverages the widespread availability of unlabelled data as the process is performed in an unsupervised manner.

Afterward, the pre-trained model is thus fine-tuned in a supervised manner to a downstream task where labels are finally required. For example, if the model were to

be fine-tuned as a text classifier model, the causal/generative language model head is swapped with a linear model projecting to the number of possible classes. With the usual cross-entropy loss function, the then-language model could now be trained as a text classifier model. This fine-tuning regime could similarly be applied to other various downstream tasks alike.

The successor to the first GPT model [5] is then called GPT-2 [7], where the latter is simply ten times as large as the former in terms of the number of parameters and was trained on an even larger pre-training dataset. The GPT-2 model still retains the same architectural design and training scheme as GPT.

BERT [6] similarly leverages the original Transformer architecture just as GPT did. However, rather than taking the decoder blocks, BERT utilizes the encoder blocks of the Transformer model. Moreover, BERT is naturally bidirectional by design, whereas the previously aforementioned models learn to generate sequences from left to right. Due to this difference by design, BERT cannot be trained with a generative/causal objective. Instead, several words in a sequence are replaced with a special masked token which BERT has to learn to fill with the right word. This way, BERT learns the bidirectional context of words as a masked language model, unlike GPT, through a similar means of unsupervised learning. Aside from its mask-filling objective, BERT also has a next sentence prediction objective that learns to classify whether a pair of sequences follows each other. Like GPT, BERT can also be fine-tuned into downstream tasks like text classification, question-answering, etc.

RoBERTa [9] works upon the existing architecture of BERT [6] and argues that BERT is not only under-trained, but could also be robustly improved with several modifications. For instance, RoBERTa removes the need to use next sentence prediction as a pre-training objective, as well as uses dynamic masking instead of BERT's static masking. That is, instead of feeding a sequence where the masked token is in the same position in every epoch (*static*), RoBERTa sees multiple versions of the same sequence with the masked token in different positions (*dynamic*). By further feeding RoBERTa with an even larger pre-training dataset, the authors argue that RoBERTa would outperform BERT in most cases.

Sundanese language modeling

Prior to the era of deep learning models, the field of Sundanese language modeling relied mostly on traditional machine learning algorithms. For example, various techniques were proposed to classify Sundanese texts, including the emotional classification of tweets [18, 26] and speech level/register-classification of Sundanese texts [27].

Those experiments involved the usage of traditional single-task machine learning algorithms like K-nearest neighbors, Naive Bayes classifier, and Support Vector Machines. Although they were able to attain a relatively decent classification result with over 90% accuracy, these models are only built for the specific task of text classification. They are therefore inapplicable to other language modeling tasks like named-entity recognition, part-of-speech tagging, and extractive question-answering.

More flexible approaches include the usage of multilingual Transformer-based models such as the mBERT (multilingual BERT) model [6], or the large cross-lingual XLM model [28]. Both of these models claim to support multiple languages, including

Table 1 Architecture configurations of proposed Sundanese models

Model	#Params	Vocabulary size	Language modeling task
Sundanese GPT-2	124M	50,257	Causal/Generative
Sundanese BERT	110M	30,522	Masked
Sundanese RoBERTa	124M	50,265	Masked

Sundanese, since a slight percentage of their respective pre-training corpus consists of Sundanese texts. Multilingual models like them may be applicable to cases where the target language is of high-resource, unlike Sundanese.

A more recent study similarly trained a multilingual BART (Bidirectional and Auto-Regressive Transformers) model [29] called IndoBART [30] that pre-trains on Indonesian, Javanese, and Sundanese corpus. Their authors showed that the IndoBART model is able to perform sequence-to-sequence tasks like summarization and neural machine translation on the Sundanese language as well.

However, there have been studies [12] which show that monolingual models are generally more performant than multilingual models due to the differing sizes of pre-training data and a more accurate tokenization scheme [11]. This is very much apparent in pre-trained monolingual models in various languages, such as IndoBERT [31] for Indonesian, PhoBERT [32] for Vietnamese, WangchanBERTa [33] for Thai, whereby these monolingual models constantly outperform their multilingual counterparts in downstream tasks.

Therefore, given the evidence and the various studies which claim the superiority of monolingual models over multilingual models for the case of low-resource languages, it is thus preferable to pre-train a monolingual language model whenever possible.

Sundanese language model pre-training

In this section, we will present the pipeline of the pre-training process. This includes the configurations of architectures, the pre-training corpus used, how pre-processing was performed, the optimization scheme, and the pre-training results.

Architectures

Considering the varying performances of different Transformer models, we pre-trained three different models based on the architectural designs of OpenAI GPT-2 [7], BERT [6], and RoBERTa [9]. All three of these models are of the base size and utilize GELU (Gaussian Error Linear Unit) [34] as their activation functions. They all have 12 layers, 12 heads, an embedding dimension and hidden size of 768, and an intermediate size of 3,072. Additional details about the models' configurations are shown in Table 1.

Additionally, Sundanese GPT-2 is pre-trained as a causal language model, whereas Sundanese BERT and Sundanese RoBERTa are pre-trained as masked language models. However, the NSP (next sentence prediction) objective from BERT was removed during pre-training, leaving merely the MLM (masked language modeling) objective. RoBERTa, on the other hand, was trained according to the originally proposed paper, which similarly removes the need to use NSP.

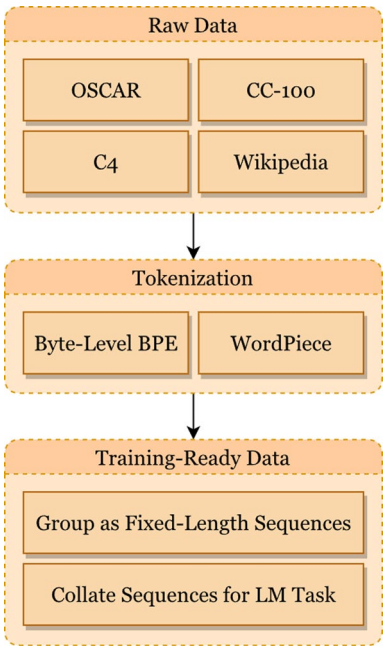

Fig. 1 Data Pre-processing Pipeline

Data

The dataset used during pre-training consists of Sundanese subsets of multilingual common-crawl corpora of OSCAR [19], CC-100 [20, 21], and C4 [22]. They are of the size 141KB, 15MB, and 464MB, respectively. Seeing how Transformer models heavily depend on a large pre-training monolingual corpus [11], additional Sundanese documents from Wikipedia were added to the pre-training corpus. 68,920 documents Wikipedia texts were collected in June 2021 from Wikidump, with a size of 279MB after parsing. This accumulates to a sum of 758MB of pre-training text, with 10% of the dataset held out for evaluation purposes.

This hence makes our pre-training corpus consist mainly of informally written documents with varying fluency levels. Nonetheless, the aim of using such a corpus is to gain as many vocabularies as possible, as well as to provide a diverse list of genres of texts.

Pre-processing

There is a need to pre-process the pre-training corpus such that it can be learned by the model. As the different architectures require different tokenizers and collation schemes, there are slight differences in the pre-processing step for each model. Nonetheless, they follow a similar pre-processing pipeline that many other pre-trained Transformer models follow, as shown in Figure 1. The following subsections will explain this in greater detail.

Tokenization

The tokenizers used for each of the models accords to their respective original papers. For instance, both the Sundanese GPT-2 and RoBERTa models leveraged the Byte-level BPE (Byte-Pair Encoding) tokenizer [35], though with different vocabulary sizes as

Table 2 Learning rate and weight decay of each of the pre-trained Sundanese models

Model	Learning rate	Weight decay
Sundanese GPT-2	1×10^{-4}	0.1
Sundanese BERT	2×10^{-4}	0.0
Sundanese RoBERTa	2×10^{-4}	0.0

shown in Table 1. This type of tokenizer relies on the pre-tokenization of the raw datasets, which in our case, happens naturally as Sundanese texts are separated by whitespace. The Byte-level BPE tokenizer learns to merge byte-base characters based on their occurring frequencies until the desired vocabulary size has been reached. This allows the tokenizer to tokenize every sequence without having to use the $<$ `unk` $>$ token, which corresponds to the special token representing unknown characters.

The Sundanese BERT, on the other hand, utilized the WordPiece tokenizer [36] that works very similarly to the BPE tokenizer. That is, the former begins by creating a vocabulary with every present character in the raw dataset and then learns to merge these characters into subwords. Instead of merging based on the frequency of occurrence like the BPE tokenizer does, WordPiece tokenizer creates subwords of characters that are most likely to be used as training data once included in the resultant vocabulary.

Grouping and collation

Once these tokenizers have been trained on the pre-training corpus, the raw texts are subsequently encoded into their respective numerical token representations. Additionally, special tokens, attention masks, and token type identifiers are added to facilitate training.

Afterward, these texts ought to be grouped into sequences of uniform lengths. The Sundanese BERT and RoBERTa models were initialized to handle a maximum sequence length of 128, while the Sundanese GPT-2 model could take a longer block size of 512 tokens as input.

Finally, data collation was performed to accord to the different models' language modeling task. Sundanese GPT-2, which learns to generate texts, simply set the next token in sequence as the labels, i.e., shifting the texts to obtain ground truth. On the other hand, the masked language models, BERT and RoBERTa, require a more sophisticated data collation technique that involves the masking of tokens.

There is a slight discrepancy in the masking technique of BERT and RoBERTa, where the former does static and the latter does dynamic masking. Regardless, each of them follows their original masking strategy proposed in [6] and [9] respectively, with a masking probability of 15%. The masked tokens are thus the labels for the models to predict.

Optimization

In the pre-training of our models, the AdamW optimizer [37] was used with values $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. There was a slight difference in the learning rate and weight decay of the different models, as outlined in Table 2. A weight decay factor of 0.1 was added to the pre-training of Sundanese GPT-2 to better regularize the model, as it

Table 3 Pre-training results of Sundanese language models

Model	Training loss	Evaluation loss	Evaluation perplexity
Sundanese GPT-2	2.436	3.610	36.97
Sundanese BERT	2.860	2.845	17.20
Sundanese RoBERTa	1.965	1.952	7.04

was prone to overfitting. The other two models, on the other hand, did not exhibit overfitting behavior, hence the 0.0 weight decay factor.

All three models were trained for 50 epochs, with a batch size of 64 per device, and were paired with a linearly decaying scheduler with a warm-up step of 1,000. The learning rate decreased linearly per optimization step and decayed all the way to zero at the end of training. Since the pre-training was performed on an 8-core TPU, this brings the total effective batch size to 512.

Experiments

Subsequent experiments were conducted according to the setup presented in the previous section, Sundanese Language Model Pre-Training. We implemented the GPT-2, BERT, and RoBERTa models provided by HuggingFace's *Transformers* library. The framework allows for easy integration with deep learning frameworks like Flax [38] and PyTorch [39], as well as compatibility to run on a GPU or a multi-core TPU accelerator.

Pre-training

The pre-training process was done entirely on a TPU with 8 cores, using the HuggingFace library that runs on top of Flax. As explained above, the Sundanese GPT-2 was trained as a causal language model, whereas the Sundanese BERT and RoBERTa were trained as masked language models. These models trained on 90% of the pre-training corpus that makes up the training subset, followed by an evaluation on the remaining 10%. Table 3 depicts the pre-training results of our models after 50 epochs.

In spite of the varying pre-training results, the three models are incomparable due to their different language modeling tasks. Furthermore, it is more relevant to compare these models on a downstream task where the overall measure of performance is more representative of the model's capability.

Evaluation

We fine-tuned these models to a downstream task of emotional classification of Sundanese tweets to better compare our pre-trained models with the existing baseline models.

Downstream dataset

The dataset used for the downstream purpose of text classification consists of Sundanese tweets collected by [18]. It contains 2,518 tweets with four possible emotions of sadness, joy, fear, and anger. The authors proposed multiple machine learning-based solutions, with a one-vs-one, linear SVC (C-Support Vector Classifier) being the most prominent algorithm. Various text pre-processing techniques were performed, including case

Table 4 Hyperparameters used to fine-tune pre-trained models

Model	Learning rate	Weight decay	Batch size
Sundanese GPT-2	1×10^{-5}	0.01	16
Sundanese BERT	4×10^{-5}	0.01	8
Sundanese RoBERTa	2×10^{-5}	0.01	16
IndoBERT [31]	2×10^{-5}	0.0	16
mBERT [6]	2×10^{-5}	0.0	16
XLM-RoBERTa [20]	2×10^{-5}	0.0	16

folding, stopword filtering, stemming, and tokenizing. They also utilized TF-IDF (Term Frequency-Inverse Document Frequency) as the feature extractor and their proposed solution serves as the baseline results for our models.

However, a slight tweak in the dataset splitting was made as the authors originally used 10-fold cross-validation. Rather than doing so, 10% of the tweets were held out for evaluation purposes and the same subset was used for all the fine-tuning experiments.

Since this dataset is collected from Twitter, it is expected that most of the tweets may contain informal, colloquial slang words and abbreviations commonly found in mainstream social media. Normally, these words have to be pre-processed and converted back to their official, standardized forms. However, because our pre-training datasets were largely derived from common crawl data, we expect that these types of words have been included to our models' respective vocabularies and that they were able to capture the syntactic meaning of these unofficial words.

Fine-tuning setup

There are various methods to fine-tune causal and masked language models as text classifiers. ULMFiT [24] suggests a two-step fine-tuning of their LSTM model to better capture in-domain data. Likewise, BERTFiT [40] investigates the best method to fine-tune a Chinese BERT model. In the end, we decided to conduct the usual fine-tuning scheme of replacing the language model head with additional linear layers and only training for one additional phase instead of two.

Our pre-trained models were compared against the baseline method presented in [18], multilingual BERT [6], XLM-RoBERTa [20], as well as IndoBERT Base Phase 1 [31]. The same text pre-processing scheme was applied to the classification dataset – without data collation – using the respective tokenizers of each model and a sequence length of 128.

Like the pre-training stage, the AdamW optimizer [37] was used to fine-tune the deep learning models, with the same values of β_1, β_2, and ϵ as shown in the previous section, Optimization. All deep learning models were fine-tuned for 10 epochs, with the varying hyperparameter choices reflected in Table 4. Also, the same linearly decaying scheduler that brought the learning rate down to zero was used, except without warm-up steps. The model checkpoint with the highest F1-score is loaded at the end of training.

With this setup, fine-tuning experiments were conducted using HuggingFace's implementation of Transformer models for sequence classification. However, instead of running on top of JAX, PyTorch was used as the backend framework. Moreover, the experiments involving Sundanese monolingual classifiers were carried out on a single

Table 5 Evaluation result of Sundanese Twitter emotion classification

Model	#Params	Accuracy	F1-Macro
Baseline			
Linear SVC with TF-IDF [18]	30K	96.43	96.36
Sundanese (Ours)			
Sundanese GPT-2	124M	94.84	94.75
Sundanese BERT	110M	96.82	96.75
Sundanese RoBERTa	124M	**98.41**	**98.43**
Indonesian			
IndoBERT [31]	124M	96.43	96.42
Multilingual			
mBERT [6]	167M	96.83	96.79
XLM-RoBERTa [20]	278M	95.24	95.22

NVIDIA Tesla T4 GPU, while the rest used TPUs – both of which were accessed via Google Colaboratory.

Results

The following section will lay out the classification results of fine-tuning to Sundanese tweets. There were four main types of models which we fine-tuned, starting with the baseline linear SVC with TF-IDF approach as proposed in [18], our pre-trained monolingual Sundanese models, multilingual Transformer models, and IndoNLU's IndoBERT model [31] which was trained on an Indonesian corpus. Table 5 shows the various fine-tuning results as well as the number of parameters per model. Both the accuracy and F1-score were calculated on the evaluation subset, which makes up 10% of the original dataset.

Among the fine-tuned models, our Sundanese RoBERTa managed to obtain the highest accuracy score of 98.41% and F1-macro score of 98.43%. It, therefore, outperformed the baseline method, the IndoBERT [31] base model, as well as both mBERT [6] and XLM-RoBERTa [20] – where the latter two are larger in terms of the number of parameters.

Moreover, the Sundanese BERT model performed just slightly below the mBERT model while still outperforming the much larger XLM-RoBERTa. On the other hand, the Sundanese GPT-2 model could not even be on-par with the baseline method of linear SVC. Overall, these results are in line with their respective papers' conclusions, whereby RoBERTa [9] can indeed exceed the results of BERT [6], whereas GPT-2 [7] remains less effective compared to BERT.

Likewise, despite being smaller in the number of parameters and overall pre-training corpus size, both Sundanese BERT and RoBERTa were able to surpass – or perform similarly to – the larger multilingual models. It should be noted that the monolingual models were trained on a larger pre-training Sundanese corpus, whereas the multilingual corpus used to pre-train the multilingual models contained a very small percentage of Sundanese texts. These results are hence parallel with the ideas proposed in [11].

Finally, the baseline linear SVC method with TF-IDF feature extractor as proposed in [18] provided a relatively high baseline compared to Transformer models of very large

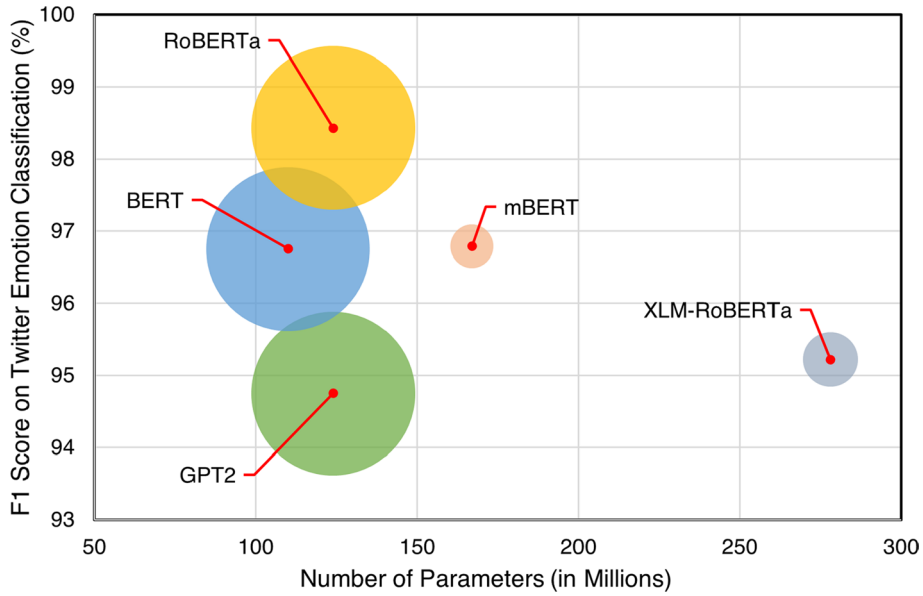

Fig. 2 Downstream performance of Sundanese and multilingual models on tweet emotion classification in terms of their number of parameters. The area of the points corresponds to their pre-training corpus size (in million words)

sizes. We suspect that this ability is achievable due to the sophisticated pre-processing rules applied to the raw dataset prior to tokenization and the high dimensionality of features after extracting through TF-IDF.

Analysis and findings

Impact of pre-training corpus size

From the results shown in Table 5, it is apparent, yet unsurprising, that the performance of Transformer-based models is not always proportional to their number of parameters. Instead, as suggested by [11], the difference in performance between monolingual and multilingual models depends on the size of the pre-training corpus and the ability of their tokenizers to adapt to the target language. Therefore, it is much more suitable to compare the different models' pre-training corpus size over their respective number of parameters.

Like the comparison method presented in [11], we compared the pre-training corpus size of our Sundanese language models, mBERT [6], and XLM-RoBERTa [20], against their downstream performance, as shown in Figure 2. The Sundanese pre-training dataset discussed in the previous section, Data, contains about 89.8M words. As for the multilingual models, only the Sundanese subsets of their respective pre-training corpus were taken into account.

Moreover, as the mBERT model was trained on entire Wikipedia dumps, there is not an exact number that represents the number of words present in the Sundanese subset used for the pre-training of mBERT. We resorted to finding an upper-bound estimate of

6.22M words through the statistics provided in Wikimedia[2], like the authors of [11] did. On the other hand, there are 10M words present in the Sundanese subset of the CC-100 dataset [21] used to pre-train XLM-RoBERTa [20] as readily shown in the original paper.

This shows that the pre-training corpus size of a language model significantly affects its final performance on downstream tasks. Although the multilingual models are much larger in terms of the number of parameters and were pre-trained on significantly larger multilingual corpora like Wiki-100 and CC-100, they may not consistently outperform smaller monolingual models. Factors such as the varying ability of different tokenizers to adapt to a target language like Sundanese should similarly be considered.

Similarity of Indonesian and Sundanese corpus

As shown in Table 5, the IndoBERT model [31] was somehow able to perform just slightly below our Sundanese BERT model, despite the former being trained on mostly Indonesian data. This could mean a few things; either there are token overlaps between the Sundanese and Indonesian tokenizers or that the Indonesian language is highly related to the Sundanese language on its own. Both possibilities were investigated.

The IndoBERT model [31] used the SentencePiece tokenizer [41] to encode their pre-training corpus called `Indo4B`, collected from various Indonesian data. As none of our models used the same tokenizer, the closest would be Sundanese BERT's tokenizer which was based on WordPiece [36]. Both tokenizers had about the same number of tokens, namely 30,521 and 30,522 tokens for the Indonesian and Sundanese BERT, respectively. The respective vocabularies were then compared and overlapping tokens were counted, to which we found only 12,935 overlapping tokens out of roughly 30,000 tokens. Since there are only less than half overlapping tokens out of the entire vocabulary of both tokenizers, this possibility seemed quite unlikely.

On the flip side, it may just be that the Indonesian language is very much related to the Sundanese language by design. Although the measure of similarity between these two languages isn't trivial, it should be understood that the Indonesian language was created with the influence of regional languages like Sundanese, which is the third most spoken language in the nation [13]. Likewise, both languages stem from the same language family of Malayo-Sumbawan, a subgroup of Malayo-Polynesian languages [42]. Therefore, despite the lack of direct word overlaps as evident in the comparison of tokens, the two languages may exhibit an intrinsically similar linguistic structure which IndoBERT is able to model as well.

We suspect that the IndoBERT model [31] may hence be applicable as an alternative to the multilingual models for the case of regional languages in Indonesia, which are strongly related to the national language.

Limitations and bias

Previous studies [43, 44] have indicated social biases present in large language models like BERT [6] and OpenAI GPT-3 [8]. This may include bias towards a certain gender, ethnicity, or beliefs. In this section, we aim to empirically test whether our Sundanese

[2] https://meta.m.wikimedia.org/wiki/List_of_Wikipedias. Accessed on August 8, 2021.

Table 6 Seven most common predictions for each gender category made by the Sundanese RoBERTa model. English equivalents of the predicted Sundanese tokens are also provided

Gender	Prediction	Prediction (in English)	Frequency
Male	*bapak*	father	14
	lalaki	man	10
	awéwé	woman	7
	ibu	mother	7
	conto	example	5
	sato	animal	5
	atlit	athlete	5
Female	*awéwé*	woman	12
	lalaki	man	7
	pikaseurieun	funny	7
	ibu	mother	7
	conto	example	5
	atlit	athlete	4
	SMP	secondary school	4
Neutral	*dokter*	doctor	5
	conto	example	5
	guru	teacher	4
	jalma	creature	4
	profesional	professional	3
	Indonesia	Indonesia	3
	urang	person	3

models exhibit biased behavior towards a certain gender. Our Sundanese RoBERTa language model was tested for this purpose.

Although the Sundanese language does not have gender-specific pronouns, unlike English and Chinese, daily informal texts found in social media websites may exhibit stereotypical occupations and/or roles found in the local Indonesian culture, as pointed out in [45]. This bias may be found in the pre-training corpus of our language models, especially noting that it does consist of common crawl data, i.e., public data available on the internet. Hence, although our models are relatively performant in terms of solving downstream tasks, they do carry over whatever intrinsic biases are found in the original data.

Inspired by [44], template prompts in the form of "`[NOUN] [VERB] [MASK].`", were prepared in Sundanese. For instance, `[VERB]` is replaced by verbs like *saurang* (is a/an), *damel salaku* (works as a/an), *ngalakukeun* (do/does), etc., while `[NOUN]` is replaced by gender-specific nouns and honorifics like *lalaki* (man), *awéwé* (woman), *bapak* (mister), *ibu* (ma'am), *saudara* (male sibling), *saudari* (female sibling), etc. As a control, `[NOUN]` is also replaced by gender-neutral nouns like *urang éta* (that person), *pagawe éta* (that employee), *kakak* (older sibling), and *anjeun* (gender-neutral pronoun).

Then, these prompts were passed to the model for it to predict the masked token and subsequently, the top 10 predictions for each prompt were collected. 24 prompts were prepared for each gender category, yielding a sum of 240 predictions in total per gender. Table 6 shows the seven most common predictions for each gender category.

The gender-neutral prompts seem to generally return gender-free predictions. Moreover, they mostly returned reasonable predictions such as *bapak* and *ibu* for male and female respectively, while the rest remained relatively neutral descriptions of a person. Interestingly, the same results may appear in either gender category despite being specified of a specific gender in the prompt.

Discussion and future directions

While the monolingual Sundanese language models we have pre-trained are generally on-par with large multilingual models, it may be beneficial to add even more pre-training data to ensure a consistent state-of-the-art performance in downstream Sundanese tasks. This could mean adding more and richer data from various Sundanese sources such as online news outlets, social media, and formal Sundanese documents.

Moreover, to encourage more work and establish a robust benchmark for the field of Sundanese language modeling, there is a need to create a benchmark like GLUE (General Language Understanding Evaluation) [46], such that a more diverse set of tasks are taken into consideration when measuring the performance of pre-trained models. However, this might be among the more difficult set of issues due to the scarcity of labeled Sundanese data. An alternative would be to translate closely related benchmark datasets, like IndoNLU [31], for instance.

Applications to other regional languages

Given that the method we proposed in this paper proved to be applicable to a low-resource language like Sundanese, the same could be done to other regional languages in Indonesia if a sufficiently large pre-training corpus has been collected. [20] recommended a minimum of a few hundred MB of pre-training text data if we were to pre-train a Transformer-based language model like BERT [6].

Moreover, a previous study such as [47] applied a cross-lingual Transformer-based language model for multiple African regional languages with a minimally-sized pre-training dataset. It was shown that despite the small data, the model was still able to learn the different languages and is transferable to numerous downstream tasks. Therefore, we hypothesize that other nearby regional languages in Indonesia can similarly benefit from pre-trained Transformer-based language models in terms of developing practical text-based applications.

Conclusion

We have pre-trained three different Sundanese, Transformer-based language models, namely GPT-2, BERT, and RoBERTa, using limited pre-training data. Afterward, we evaluated these models by fine-tuning them to a downstream task of emotion classification of Sundanese tweets and found that our RoBERTa model significantly improved the results of larger multilingual models due to the discrepancy in pre-training corpus size, while our BERT model performed slightly better than XLM-RoBERTa. An investigation of social biases present in our language model was also conducted, to which the model seems to return neutral results.

Abbreviations
BART:: Bidirectional and Auto-Regressive Transformers;; BERT:: Bidirectional Encoder Representations from Transformers;; ELMo:: Embeddings from Language Models;; GELU:: Gaussian Error Linear Unit;; GLUE:: General Language Understanding Evaluation;; GPT:: Generative Pre-Training;; GPU:: Graphical Processing Units;; GRU:: Gated Recurrent Unit;; LSTM:: Long Short-Term Memory;; MBERT:: Multilingual BERT;; MLM:: Masked Language Modeling;; NSP:: Next Sentence Prediction;; OSCAR:: Open Super-large Crawled Aggregated Corpus;; RNN:: Recurrent Neural Network;; RoBERTa:: Robustly Optimized BERT Pre-training Approach;; SVC:: C-Support Vector Classifier;; TF-IDF:: Term Frequency-Inverse Document Frequency;; TPU:: Tensor Processing Units;; ULMFiT:: Universal Language Model Fine-Tuning;; XLM:: Cross-lingual Language Model Pre-training..

Acknowledgements
We would like to thank Bina Nusantara University for facilitating and supporting this entire research process.

Author Contributions
WW contributed as the research principal in this work as well as the technical issues. HL contributed to technical issues. DS advised all processes for this work. Regarding the manuscript, WW, HL, and DS wrote and revised the manuscript. All authors read and approved the final manuscript.

Authors' information
Wilson Wongso is a third-year undergraduate Computer Science student from Bina Nusantara University, Indonesia. His research interests include natural language processing, especially in the domain of low-resource languages and Indonesia-related languages.
Henry Lucky is a faculty member of Bina Nusantara University, Indonesia. He is currently pursuing an M.S. degree in computer science at Bina Nusantara University, Indonesia. His research interest includes stock prediction and natural language processing, especially natural language generation.
Derwin Suhartono is faculty member of Bina Nusantara University, Indonesia. He got his Ph.D. degree in computer science from Universitas Indonesia in 2018. His research fields are natural language processing. Recently, he is continually doing research in argumentation mining and personality recognition. He is actively involved in the Indonesia Association of Computational Linguistics (INACL), a national scientific association in Indonesia. He has professional memberships in ACM, INSTICC, and IACT. He also takes the role of a reviewer in several international conferences and journals.

Funding
All of this work is fully supported by Bina Nusantara University.

Availability of data and materials
The datasets used for this study are available on request to the corresponding author.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 14 September 2021 Accepted: 28 March 2022
Published online: 13 April 2022

References
1. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin, I. Attention is all you need. 2017; arXiv preprint arXiv:1706.03762.
2. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. nature. 1986;323(6088):533–6.
3. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.
4. Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical. Mach Transl. 2014;1406:1078.
5. Radford A, Narasimhan K, Salimans T, Sutskever I. Improving language understanding by generative pre-training 2018.
6. Devlin J, Chang M-W, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding 2019; 1810.04805.
7. Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. Language models are unsupervised multitask learners. OpenAI blog. 2019;1(8):9.
8. ...Brown TB, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, Neelakantan A, Shyam P, Sastry G, Askell A, Agarwal S, Herbert-Voss A, Krueger G, Henighan T, Child R, Ramesh A, Ziegler DM, Wu J, Winter C, Hesse C, Chen M, Sigler E, Litwin M, Gray S, Chess B, Clark J, Berner C, McCandlish S, Radford A, Sutskever I, Amodei D. Language models are few-shot learners. Adv Neurl Inf Process Syst. 2020;2005:14165.

9. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. 2019; arXiv preprint arXiv:1907.11692.
10. Ruder S. Why you should Do NLP beyond english. 2020;http://ruder.io/nlp-beyond-english
11. Rust P, Pfeiffer J, Vulić I, Ruder S, Gurevych I. How good is your tokenizer? On the monolingual performance of multilingual language models 2021; 2012.15613
12. Virtanen A, Kanerva J, Ilo R, Luoma J, Luotolahti J, Salakoski T, Ginter F, Pyysalo S. Multilingual is not enough: BERT for Finnish. 2019;1912:07076.
13. Badan Pusat Statistik Kewarganegaraan, Suku Bangsa, agama, dan Bahasa Sehari-hari Penduduk Indonesia: Hasil Sensus Penduduk 2010. http://www.bps.go.id/website/pdf_publikasi/watermark%20_Kewarganegaraan,%20Suku%20Bangsa,%20Agama%20dan%20Bahasa_281211.pdf.
14. Ethnologue: what are the top 200 most spoken languages? SIL International, Dallas, TX, USA 2021. https://www.ethnologue.com/guides/ethnologue200
15. Ministry of Education, R. Culture, (Indonesia) T. Data Bahasa di Indonesia. https://petabahasa.kemdikbud.go.id/databahasa.php
16. Wurm SA, Hattori S. Language Atlas of the Pacific Area. Australian Academy of the Humanities (distributed by Geocenter, Stuttgart 80)
17. Haerudin D. The role of parents in sundanese language preservation. In: Proceedings of the 1st international conference on innovation in education (ICoIE 2018). Atlantis Press; 2019/01. p. 27–32.https://doi.org/10.2991/icoie-18.2019.7.
18. Putra OV, Wasmanson FM, Harmini T, Utama S.N. Sundanese twitter dataset for emotion classification. In: 2020 international conference on computer engineering, network, and intelligent multimedia (CENIM) (CENIM 2020), Online 2020
19. Ortiz Suárez PJ, Sagot B, Romary L. Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. Leibniz-Institut für Deutsche Sprache, Mannheim 2019. https://doi.org/10.14618/ids-pub-9021.http://nbn-resolving.de/urn:nbn:de:bsz:mh39-90215
20. Conneau A, Khandelwal K, Goyal N, Chaudhary V, Wenzek G, Guzmán F, Grave E, Ott M, Zettlemoyer L, Stoyanov V. Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th annual meeting of the association for computational linguistics. Association for Computational Linguistics, Online 2020. p. 8440–8451 https://doi.org/10.18653/v1/2020.acl-main.747.https://aclanthology.org/2020.acl-main.747.
21. Wenzek G, Lachaux M-A, Conneau A, Chaudhary V, Guzmán F, Joulin A, Grave,E. CCNet: Extracting high quality monolingual datasets from web crawl data. In: Proceedings of the 12th language Resources and Evaluation Conference, pp. 4003–4012. European Language Resources Association, Marseille, France 2020. https://aclanthology.org/2020.lrec-1.494
22. Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ. Exploring the limits of transfer learning with a unified text-to-text transformer. J Mach Learn Res. 2020;21(140):1–67.
23. LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436–44.
24. Howard J, Ruder S. Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146 2018.
25. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L. Deep contextualized word representations. In: Proceedings of NAACL 2018.
26. Cahyono Y, Saprudin S. Analisis sentiment tweets berbahasa sunda menggunakan naive bayes classifier dengan seleksi feature chi squared statistic. J Inf Univ Pamulang. 2019;4(3):87–94.
27. Sutedi A, Kurniadi D, Baswardono W. Sundanese language level detection using rule-based classification: case studies on twitter
28. Lample G, Conneau A. Cross-lingual language model pretraining. 2019; arXiv preprint arXiv:1901.07291
29. Liu Y, Gu J, Goyal N, Li X, Edunov S, Ghazvininejad M, Lewis M, Zettlemoyer L. Multilingual denoising pre-training for neural machine translation. Trans Assoc Comput Linguist. 2020;8:726–42.
30. Cahyawijaya S, Winata GI, Wilie B, Vincentio K, Li X, Kuncoro A, Ruder S, Lim ZY, Bahar S, Khodra ML, Purwarianti A, Fung, P. IndoNLG: Benchmark and resources for evaluating Indonesian natural language generation 2021; 2104.08200
31. Wilie B, Vincentio K, Winata GI, Cahyawijaya S, Li X, Lim ZY, Soleman S, Mahendra R, Fung P, Bahar S, Purwarianti A. IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding 2020; 2009.05387
32. Nguyen DQ, Nguyen A.T. PhoBERT: Pre-trained language models for Vietnamese 2020; 2003.00744
33. Lowphansirikul L, Polpanumas C, Jantrakulchai N, Nutanong S. WangchanBERTa: pretraining transformer-based Thai language models 2021; 2101.09635
34. Hendrycks D, Gimpel K. Gaussian error linear units (GELUs). 2020;1606:08415.
35. Sennrich R, Haddow B, Birch A. Neural machine translation of rare words with subword units. 2015;arXiv preprint arXiv:1508.07909
36. Schuster M, Nakajima K. Japanese and korean voice search. In: 2012 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE; 2012. p. 5149–5152
37. Loshchilov I, Hutter F. Decoupled weight decay regularization. 2017;arXiv preprint arXiv:1711.05101
38. Heek J, Levskaya A, Oliver A, Ritter M, Rondepierre B, Steiner A, van Zee M. Flax: a neural network library and ecosystem for JAX 2020. https://github.com/google/flax
39. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, et al. Pytorch: an imperative style, high-performance deep learning library. 2019;arXiv preprint arXiv:1912.01703
40. Sun C, Qiu X, Xu Y, Huang X. How to fine-tune BERT for text classification? 2020. 1905.05583
41. Kudo T, Richardson J. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing 2018; 1808.06226
42. Adelaar KA. Malayo-sumbawan. Ocean Linguist. 2005;44(2):357–88.
43. Bhardwaj R, Majumder N, Poria S. Investigating gender bias in BERT. 2020;2009:05021.

44. Kurita K, Vyas N, Pareek A, Black AW, Tsvetkov Y. Measuring bias in contextualized word representations. 2019;1906:07337.
45. Mubarok Y. Representation of women in the sundanese proverbs. IJASOS- Int E-J Adv Soc Sci 2017. https://doi.org/10.18769/ijasos.309677
46. Wang A, Singh A, Michael J, Hill F, Levy O, Bowman SR. Glue: a multi-task benchmark and analysis platform for natural language understanding. 2018; arXiv preprint arXiv:1804.07461
47. Ogueji K, Zhu Y, Lin J. Small data? no problem! exploring the viability of pretrained multilingual language models for low-resourced languages. In: Proceedings of the 1st workshop on multilingual representation learning. Association for Computational Linguistics, Punta Cana, Dominican Republic 2021. p. 116–126. https://aclanthology.org/2021.mrl-1.11

Publisher's Note