International Journal of Engineering Trends and Technology
ISSN: 2231-5381/ https://doi.org/10.14445/22315381/IJETT-V70I5P240

Volume 70 Issue 5, 367-381, May 2022

Original Article

IndoXLNet: Pre-Trained Language Model for Bahasa Indonesia

Thiffany Pratama[1], Suharjito[2]

[1,2]*Computer Science Department, BINUS Online Learning, Bina Nusantara University, Jakarta, Indonesia.*

[1]thiffanyp@gmail.com

Received: 28 March 2022 Revised: 23 May 2022 Accepted: 30 May 2022 Published: 31 May 2022

Abstract *- BERT has been widely adopted to create pre-trained models in various languages, one of which is IndoBERT, a BERT-based pre-trained model for Bahasa Indonesia. However, BERT still has limitations, neglectingthe masked token's position and the difference between the pre-training and fine-tuning processes. XLNet has been proven to overcome the limitations of BERT by combining the autoregressive language model and autoencoding methods. Unfortunately, no one has developed a pre-trained XLNet model for Bahasa Indonesia. Therefore, this research aims to create a pre-trained XLNet model specifically for Bahasa Indonesia. This model can be used to solve Natural Language Processing problems in Bahasa Indonesia, such as sentiment analysis and named-entity recognition. The model is called IndoXLNet. IndoXLnet is trained using corpus datasets in Bahasa Indonesia to capture the context of the word representation in Bahasa Indonesia better than IndoBERT. It is proven that after testing various Natural Language Processing tasks on the IndoNLU benchmark, IndoXLNet's average F1-score performance increased against IndoBERT by 3.06% with an equivalent architecture.*

Keywords *- Bahasa Indonesia, BERT, Natural Language Processing, Pre-trained Model, XLNet.*

1. Introduction

Developing deep learning models in NLP (Natural Language Processing) has helped solve complex problems such as information extraction, semantic role labeling, part-of-speech tagging, and many other problems[1]. Deep learning models, especially pre-trained deep learning models, have been resulting good performances in solving various cases in the field of Natural Language Processing in English, such as BERT (Bidirectional Encoder Representations from Transformers)[2]and XLNet [3]. Both models can get excellent benchmark results when tested on several benchmark platforms such as GLUE (The General Language Understanding Evaluation). [4]and SQuAD (The Stanford Question Answering Dataset) [5].

BERT was developed using the MLM (Mask Language Model) method. MLM works by randomly masking the tokens in the input sequence, and then the model predicts the token. With this method, BERT can capture the context of the representation of words from both directions. With this ability, BERT can solve problems in the NLP field very well. It was proven when tested against the GLUE and SQuAD benchmark platforms. In this study, two variants of BERT were made, namely $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$.

BERT has been widely used to solve various specific problems in NLP. Detecting malware characteristics[6], extracting sentiment from financial news [7], and even clinical trial information extraction [8].

The successful application of BERT to NLP problems has led to the developmentof BERT-based pre-trained models for various languages. For example, there is IndoBERT [9], Flaubert[10], CamemBERT [11], Kr-BERT [12], Spanish-BERT [13], Arabic-BERT [14], and others. The development of a specific pre-trained model based on language is carried out because there are differences in sentence structure, word structure, and vocabulary.

The IndoBERT pre-trained model is a BERT-based pre-trained model trained using the Bahasa Indonesia corpus dataset. In this study, several variants of IndoBERT are generated, namely: $\text{IndoBERT}_{\text{LARGE}}$, $\text{IndoBERT}_{\text{BASE}}$, $\text{IndoBERT-lite}_{\text{LARGE}}$, and $\text{IndoBERT-lite}_{\text{BASE}}$. Each variant is divided into two training phases: phase 1 (p1) and phase 2 (p2). Corpus datasets in Bahasa Indonesia allow IndoBERT to capture the context of the representation of words in Bahasa Indonesia. It is proven by testing the model against the IndoNLU benchmark [9], which contains twelve testing datasets with different problem categories. The categories of problems include single sentence classification, single

sentence order marking, sentence pair classification, and sentence pair order labeling. This model produces good performance and can exceed the performance of the XLM-$_{LARGE}$ (Cross-lingual Language Model) model [15] on five of the six text classification tasks and four of the eight sequence labeling tasks [9].

Although it has been proven effective in solving various problems in the NLP field, BERT still has limitations in its ability to capture the context of the representation of words. The first limitation of BERT is that it ignores the position of masked tokens. This model predicts tokens masked randomly during the training process using the MLM method. As a result, the masked token will be independent of other tokens and automatically remove the context of the token's position against other tokens. The second limitation of BERT is the difference between the pre-training and fine-tuning processes. The artificial symbol [MASK] used in the BERT pre-training process does not appear in the fine-tuning process, causing a mismatch between the pre-training and fine-tuning processes [3].

XLNet model [3] was developed to overcome the limitations of BERT and has been proven to exceed the performance of BERT in document ranking & reading comprehension, text classification, and natural language understanding in English after being tested on several benchmark platforms such as GLUE, SQuAD, and RACE (The Reading Comprehension dataset from Examinations) [16]. In this research, two variants of XLNet were generated, namely XLNet$_{BASE}$, which is equivalent to BERT$_{BASE}$, and XLNet$_{LARGE}$, which is equivalent to BERT$_{LARGE}$.

The development of research on the pre-trained model for Indonesian is still minimal, even though Indonesian is one of the languages with the largest number of speakers. In this situation, IndoBERT has succeeded in becoming one of the most widely used Indonesian pre-trained models because of its ability to solve NLP problems in Bahasa Indonesia. However, this model inherits weaknesses from BERT itself, namely the neglect of the position of masked tokens and the difference between the pre-training and fine-tuning processes. IndoBERT's weakness can be overcome by creating a pre-trained model based on XLNet specifically for Indonesian. However, there has been no research to create an XLNet-based pre-trained model specifically for Indonesian. In addition, the statement that XLNet can exceed the performance of BERT and overcome the limitations of BERT can only be proven in the NLP problem in English. It is necessary to prove it in Bahasa Indonesia.

Based on these problems, this research aims to create a pre-trained model using the XLNet model architecture, specifically for Indonesian. The pre-trained model is from now on referred to as IndoXLNet. IndoXLNet is then evaluated by comparing its performance against the equivalent IndoBERT model on the Indonesian language testing datasets provided by IndoNLU.

Two variants of IndoXLNet are generated in this research, namely IndoXLNet-4B and IndoXLNet-4Bplus. The IndoXLNet-4B and IndoXLNet-4Bplus hyperparameters are set based on XLNetBASE so that they are equivalent to the IndoBERT-base-phase 1 (p1) variant. The difference between IndoXLNet-4B and IndoXLNet-4Bplus lies in the dataset used to train the model. IndoXLNet-4B was trained using the same dataset used by IndoBERT-base-p1 consisting of Indo4B [9], whereas IndoXLNet-4Bplus was trained using Indo4B and web crawling data.

IndoXLNet-4B and IndoXLNet-4Bplus were evaluated using the testing dataset provided by IndoNLU. The comparison is conducted between the performance of IndoXLNet-4B against IndoBERT-base-p1 to measure the effectiveness of the model between the two. Meanwhile, IndoXLNet-4Bplus is compared against the performance of IndoXLNet-4B to see the effect of adding more training datasets to IndoXLNet's performance.

2. Related Works

Contextual language models have continued to progress rapidly since the development of ELMo (Embeddings from Language Models) [17] and GPT (Generative Pre-Training) [18], which have shown excellent results for solving problems in the natural language processing field. ELMo can capture the context of the representation of words in depth by using two layers of film (Bidirectional Language Model). By using this method, ELMo can model the use of words with complex characters in terms of semantics and syntax. The ELMo was tested with six natural language processing tasks, including answering questions, textual entailment, and sentiment analysis. The results of the ELMo test prove that ELMo can outperform all previous best models for each of these tasks. Unlike ELMo, GPT was developed by utilizing the Transformers architecture to generate pre-trained models. GPT was tested with twelve natural language processing tasks. The GPT test results show that this model outperforms other models using discriminatory training methods on five of the six inference language tasks, all questions answering questions, one out of two classification tasks, and two out of three semantic similarity tasks. Both ELMo and GPT use the pre-training method to capture the context of the representation of a sequence. GPT uses a unidirectional language model method, where the context of the representation captured by the model is only carried out in one direction. It is not optimal for completing tasks that require linguistic context at the sentence level.

Research on the BERT model [2] was conducted to overcome the limitations of unidirectional language models such as GPT. BERT was developed using the MLM method

to capture the context of sequence representation from two directions (forward and backward). In addition, BERT also utilizes the multi-head attention layer on the Transformer architecture, resulting in a faster training phase. BERT was tested with eleven natural language processing tasks and outperformed ELMo and GPT on all of these tasks. The natural language processing tasks given include question answering, natural language inference, sentence similarity, and sentiment analysis tasks. The test results prove that BERT can capture the context of the representation of words in sequences better than ELMo and GPT. However, using the MLM method on BERT results in several limitations: ignoring the position of the masked token and differences between the pre-training and fine-tuning process.

Research on the challenges of NLP for Bahasa Indonesia [19] was conducted to highlight the challenges of NLP in Bahasa Indonesia. This study describes four common challenges to developing an NLP model for Bahasa Indonesia. The lack of dataset sources in Bahasa Indonesia, the diversity of regional languages and different dialects, orthographic variations in regional languages, and social challenges are the challenges mentioned in this study. Better documentation of regional languages with different dialects is needed. Adding regional language metadata and adding metadata and register of different language styles can enrich the dataset sources used to conduct NLP research for Bahasa Indonesia.

Research on the IndoBERT model [9]was carried out to create a pre-trained model used as a baseline model for the IndoNLU benchmark. IndoBERT is a pre-trained model using BERT architecture specifically for Bahasa Indonesia. IndoBERT uses the indo4B dataset of approximately four billion words and 250 million sentences in Bahasa Indonesia. Several variants of IndoBERT were generated, including $\text{IndoBERT}_{\text{LARGE}}$, $\text{IndoBERT}_{\text{BASE}}$, $\text{IndoBERT-lite}_{\text{LARGE}}$, and $\text{IndoBERT-lite}_{\text{BASE}}$. Each variant is divided into two training phases: phase 1 (p1) and phase 2 (p2). In line with the success achieved by BERT, IndoBERT's performance can exceed the performance of the $\text{XLM-R}_{\text{LARGE}}$ model on five of the six text classification tasks and four of the eight sequence labeling tasks contained in the IndoNLU benchmark.

IndoBERT is widely adopted as a basic pre-trained model to solve NLP problems in Bahasa Indonesia. Research on sentiment analysis to detect hoaxes about Covid-19 [20], research on creating web-based applications to detect clickbait in news headlines [21], and research on the detection of hate speech on Twitter with Bahasa Indonesia tweets [22], these studies adopted IndoBERT as their base model.

Research on the XLNet [3] was carried out to create a pre-trained model that can overcome the limitations in BERT. XLNet was developed by modeling all possible input sequence factorization order permutations. The model can capture the context of the representation of words in both directions (backward and forward) and does not ignore the position of the input sequence tokens. Instead of using the MLM method, XLNet uses the autoregressive language model method in the pre-training process. There is no difference in the process between pre-training and fine-tuning. XLNet is tested over twenty different natural language processing tasks, including question answering, natural language inference, sentence similarity, and sentiment analysis tasks. XLNet's performance was proven to exceed the performance of BERT on all of these tasks when tested on GLUE [4], SQuAD [5], and RACE [16].

Research [23] strengthens the statement that XLNet's performance is better than BERT. This study made performance comparisons between several models, including BERT and XLNet. This study compares the performance of each model to perform an emotion classification task on the ISEAR (International Survey on Emotion Antecedents and Reactions) dataset [24]. The comparison results in this study indicate that the XLNet model obtains a better F1-score than the BERT model on all emotion labels.

Research on the IndoXLNet is carried out for two main reasons, XLNet's performance is better than BERT, and there is no pre-trained model with XLNet architecture for Bahasa Indonesia. In contrast to IndoBERT, which uses the BERT architecture, IndoXLNet uses the XLNet architecture to eliminate the limitations of the IndoBERT model.

Based on similar studies, a comparison table is created between these research containing the evaluation results and the dataset used in Table 1. The testing datasets shown in the table are only similar datasets tested by other research. The score shown in the table for IndoBERT is the average score for the IndoBERT-base-p1 variant and does not include the CASA dataset.

Table 1. Related works summary

Author	Proposed Model	Evaluation Result		Dataset Pre-training
		Dataset Testing	Score	
[17]	ELMo	SNLI	88.7 ± 0.17 (accuracy)	1B Word Benchmark.
[18]	GPT	GLUE	72.8 (average score)	BooksCorpus (800M words)
		SNLI	89.9 (accuracy)	
		RACE	59.0 (accuracy)	
[2]	BERT	GLUE	82.1 (average score)	BooksCorpus (800M words) dataset English Wikipedia (2,500M words)
		SQuAD v1.1	93.2 (F1-Score)	
		SQuAD v2.0	83.1 (F1-Score)	
[9]	IndoBERT	IndoNLU	77.469 (average score)	Indo4B
[3]	XLNet	GLUE	90.54 (average score)	BooksCorpus (800M words) dataset English Wikipedia (2,500M words) Giga5 (16GB text) dataset ClueWeb 2012-B dataset Common Crawl dataset
		SQuAD v1.1	95.08 (F1-Score)	
		SQuAD v2.0	90.69 (F1-Score)	

3. Research Method

The schematic framework for this research is described as shown in Fig. 1. In this research, the development of IndoXLNet begins by identifying the data requirements that will be used as material to train the model. After the data needs are known, research activities are continued by collecting relevant data as material for training the model. The corpus dataset collected is the Bahasa Indonesia corpus dataset. After the corpus dataset is collected, research activity is then followed by identifying the characteristics of the corpus dataset. Identification is required to preprocess the corpus dataset. From the results of identifying the characteristics of the corpus dataset, a text cleaning method is determined according to the characteristics of each corpus dataset. The research activity was then followed by the tokenization process of the cleaned corpus dataset.

Furthermore, the corpus dataset trains the pre-trained IndoXLNet models (IndoXLNet-4B and IndoXLNet-4Bplus). The model will be evaluated by comparing the performance generated by IndoXLNet-4B against the performance generated by one of the variants of the IndoBERT model, which is equivalent to IndoXLNet, namely IndoBERT-base-p1. The tasks used for the model evaluation process include singlesentence classification, singlesentence sequencetagging, sentencepair classification, and sentencepair sequence labeling.

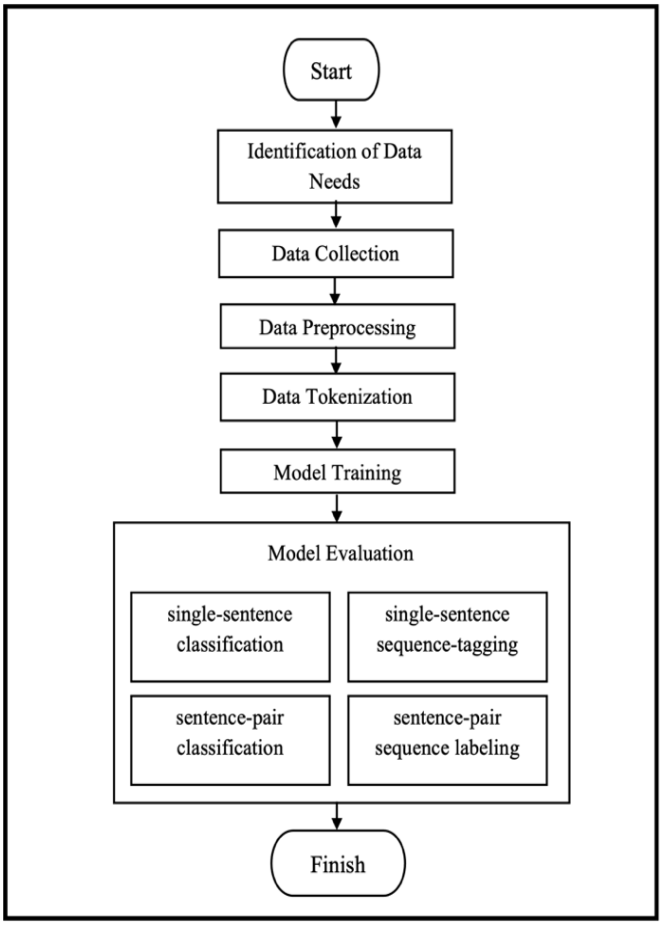

Fig. 1 Research method

3.1. Proposed Model

A pre-trained model specifically for Bahasa Indonesia using the XLNet architecture is proposed. This model is called IndoXLNet. IndoXLNet is a pre-trained model that

can be used to solve NLP problems in Bahasa Indonesia by modifying the output layer of the model.

Compared to BERT, XLNet gives better results in solving NLP tasks, according to research conducted in [3]. XLNet is a generalized autoregressive (AR), whereas BERT is an auto-encoding (AE) language, model. XLNet takes advantage of the AE model and reduces the cons of the AR model by modeling all possible permutations of the input sequence factorization order.

In this study, two variants of the IndoXLNet model are generated. The first variant is called IndoXLNet-4B, whereas the second is called IndoXLNet-4Bplus. IndoXLNet-4B is used to compare the performance of IndoXLNet against IndoBERT so that the hyperparameters are set based on XLNetBASE and the dataset used by IndoXLNet-4B is the same as the dataset used by IndoBERT-base-p1 to produce an equal comparison between them. IndoXLNet-4Bplus is generated to see the effect of adding more datasets to IndoXLNet. The dataset used by IndoXLNet-4Bplus is the result of adding to the dataset used by IndoXLNet-4B and web data crawling. The hyperparameters used by IndoXLNet-4Bplus are the same as those of IndoXLNet-4B, so the comparison between IndoXLNet-4B and IndoXLNet-4Bplus is caused only by adding datasets.

IndoXLNet (IndoXLNet-4B and IndoXLNet-4Bplus) were trained to use Bahasa Indonesia datasets to capture contexts of representation of words in Bahasa Indonesia. One of the differences between the pre-trained $XLNet_{BASE}$ and IndoXLNet models lies in the data source used as model training materials. The $XLNet_{BASE}$ pre-trained model uses various English datasets in the pre-training stage, whereas the IndoXLNet datasets use Bahasa Indonesia, a mixture of formal and informal languages. In the pre-training process, the dataset used to create the $XLNet_{BASE}$ pre-trained model consists of an English corpus, including BooksCorpus and English Wikipedia. Meanwhile, the dataset used to create IndoXLNet-4B, and IndoXLNet-4Bplus consists of Bahasa Indonesia corpus, including Indo4B for IndoXLNet-4B and Indo4B with web crawl data for IndoXLNet-4Bplus. The dataset differences between IndoXLNet and $XLNet_{BASE}$ are described in Fig. 2.

IndoXLNet (IndoXLNet-4B and IndoXLNet-4Bplus) adopted the hyperparameter used by $XLNet_{BASE}$[3] to obtain an equal comparison between IndoXLNet-4B and IndoBERT-base-p1. However, not all hyperparameters on IndoXLNet are the same as hyperparameters on $XLNet_{BASE}$. It is because of the memory limitation on the 8-V2 TPU on the Google Colab Pro+ platform. These differences include the max sequence length, batch size, learning rate, and steps. The hyperparameter differences between IndoXLNet and XLNet are explained in Fig. 2.

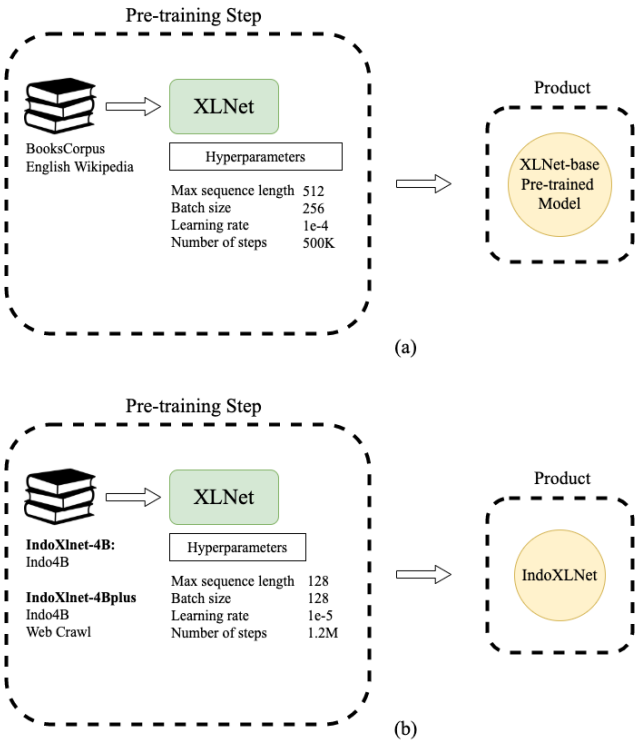

Fig. 2 Difference between IndoXLNet and XLNet; (a) XLNet (b) IndoXLNet

3.2. Identification of Data Needs

IndoXLNet is a pre-trained model of XLNet specifically for Bahasa Indonesia, so the dataset corpus needed is in Bahasa Indonesia. A large collection of dataset corpus is required to create a pre-trained model so that the model can capture contextual information in Bahasa Indonesia texts.

A particular format is required for the corpus dataset to be used as a data source for the pre-trained model. Each sentence in the corpus dataset is separated by the character "\n" or a new line. A <eop> token separate each paragraph in the corpus dataset, and each input document in the corpus dataset is separated by one blank line.

3.3. Data Collection

There are two primary dataset sources used to train the two IndoXLNet variants. The first dataset is Indo4B and the second is web crawl data from the Leipzig Corpora Collection [25]. One of the Indo4B datasets is Twitter data, but because of the provisions of Twitter, IndoNLU cannot publish the Twitter dataset. The new Twitter data is collected to replace Twitter data from the Indo4B dataset to overcome this.

The Indo4B dataset contains approximately three and a half billion words and 250 million sentences containing formal and colloquial language texts. The Indo4B dataset is

divided into twelve sub-datasets. Eight sub-datasets contain formal language, and the rest contain a mixture of formal and colloquial. The web crawl data of the Leipzig Corpora Collection includes fifteen sub-datasets, four of them contain formal language, and the rest contain colloquial.

IndoXLNet-4B uses the same dataset as IndoBERT-base-p1, namely the Indo4B dataset (Table 2) as the pre-training dataset. Meanwhile,

IndoXLNet-4Bplus uses the Indo4B dataset (Table 2) plus the web crawl data (Table 3) as the pre-training dataset.

Table 2. Dataset used to train IndoXLNet-4B

Dataset	Number of Words	Number of Sentences	Size	Style	Source
Crawl Twitter	186.568.621	12.446.966	1.04 GB	Mixed	Twitter
OSCAR	2.279.761.186	148.698.472	14.9 GB	Mixed	IndoNLU
CoNLLu Common Crawl	905.920.488	77.715.412	6.1 GB	Mixed	
OpenSubtitles	105.061.204	25.255.662	664.8 MB	Mixed	
Wikipedia Dump	76.263.857	4.768.444	528.1 MB	Formal	
Wikipedia CoNLLu	62.373.352	4.461.162	423.2 MB	Formal	
OPUS JW300	8.002.490	586.911	52 MB	Formal	
Tempo	5.899.252	391.591	40.8 MB	Formal	
Kompas	3.671.715	220.555	25.5 MB	Formal	
BPPT	500.032	25.943	3.5 MB	Formal	
Parallel Corpus	510.396	35.174	3.4 MB	Formal	
TALPCo	8.795	1.392	56.1 KB	Formal	
Frog Storytelling	1.545	177	10.1 KB	Mixed	

Table 3. Additional dataset to train IndoXLNet-4Bplus

Dataset	Number of Words	Number of Sentences	Size	Style	Source
ind_newscrawl-tufs6_2012_3M	50.458.616	3.000.000	358.1 MB	Formal	Leipzig Corpora Collection
ind_web-tufs3_2015_3M	49.687.294	3.000.000	354.5 MB	Mixed	
ind_newscrawl-tufs5_2011_3M	48.885.244	3.000.000	343.9 MB	Formal	
ind_web-tufs13_2012_3M	48.181.957	3.000.000	342.7 MB	Mixed	
ind_mixed_2013_1M	18.445.026	1.000.000	114.6 MB	Mixed	
ind_mixed_2012_1M	17.331.393	1.000.000	105 MB	Mixed	
ind-id_web_2015_1M	16.555.175	1.000.000	117.3 MB	Mixed	
ind_web-tufs2_2013_1M	16.346.171	1.000.000	115.9 MB	Mixed	
ind-id_web_2013_1M	16.344.359	1.000.000	115.1 MB	Mixed	
ind-id_web-public_2017_1M	16.210.551	1.000.000	117.5 MB	Mixed	
ind-id_web_2017_1M	15.882.357	1.000.000	113.2 MB	Mixed	
ind_mixed-tufs4_2012_1M	15.758.902	1.000.000	112.5 MB	Mixed	
ind_wikipedia_2021_1M	15.703.915	1.000.000	110.9 MB	Formal	
ind_news_2020_1M	15.469.717	1.000.000	110.1 MB	Formal	
ind_news_2019_1M	15.253.181	1.000.000	107.1 MB	Formal	

3.4. Data Preprocessing

IndoXLNet is a pre-trained model of XLNet specifically for Bahasa Indonesia, so the dataset corpus needed is in Bahasa Indonesia. A large collection of dataset corpus is required to create a pre-trained model so that the model can capture contextual information in Bahasa Indonesia texts.

The datasets were analyzed by taking samples from each sub-dataset. Dataset analysis is needed to find patterns of discrepancies in each sub-dataset text. After finding the patterns, a text cleaning method is determined for each sub-dataset.

Sentences with a word count of less than four and not part of a single document are eliminated from the corpus. Numbers in parentheses that indicate the year or order of verses in the scriptures are also eliminated from the corpus. The sentence structure in several sub-datasets is adjusted to the needs of the XLNet training data specifications by separating documents using blank lines.

Due to the limitations of the Google Colab Pro+ runtime and the memory on the TPU, the corpus dataset is split into several parts.

3.5. Data Tokenization

The cleaned dataset is then tokenized using the Sentencepiece library. Several parameters need to be set to tokenize data using this library. The data tokenization parameter is the same as the setup on $XLNet_{BASE}$[3]. The detail of the parameters in Table 4. All datasets broken down are then converted into .tfrecords using the Sentencepiece library.

Table 4 Sentencepiece parameter

Parameter	Value
vocab_size	32000
character_coverage	0.99995
model_type	unigram
control_symbols	<cls>,<sep>,<pad>,<mask>, <eod>
user_defined_symbols	<eop>,.,(,),",-,–,£,€
shuffle_input_sentence	True
input_sentence_size	10.000.000

3.6. Model Training

The two IndoXLNet variants are trained using different datasets, but each dataset must be in the form of .tfrecords. The training for the two IndoXLNet variants was carried out on the Google Colab Pro+ platform by utilizing the 8-V2 TPU.

The pre-training hyperparameters used to generate IndoXLNet-4B and IndoXLNet-4Bplus are set to $XLNet_{BASE}$[3]. It aims to obtain an equal comparison between the two variants of IndoXLNet and IndoBERT-base-p1. However, not all hyperparameters are created equal due to memory limitations on the platform used. The different hyperparameters are the max sequence length, batch size, learning rate, and steps. Details of the IndoXLNet and IndoXLNet-4Bplus pre-training hyperparameters are listed in Table 5.

Table 5. Hyperparameters for IndoXLNet-4B and IndoXLNet-4Bplus[3]

Hyperparameter	Value
train_batch_size	128
seq_len	128
reuse_len	64
mem_len	384
perm_size	64
n_layer	12
d_model	768
d_embed	768
n_head	12
d_head	64
d_inner	3072
untie_r	True
mask_alpha	6
mask_beta	1
num_predict	85
uncased	True
ff_activation	gelu
learning_rate	1e-5
weight_decay	0.01
dropout	0.1
dropatt	0.1

There are several different hyperparameters between the two variants of the IndoXLNet model with IndoBERT-base-p1. The difference lies in the batch size, learning rate, steps, and vocab size. The difference in batch size, learning rate, and steps is due to limitations on the platform used to train the two IndoXLNet variants. The difference in these three hyperparameters is not related to the model architecture but to the model training process on the platform used. Meanwhile, the difference in vocab size in the two variants of IndoXLNet and IndoBERT-base-p1 is because the IndoXLNet hyperparameter is set based on XLNetBASE. The hyperparameter differences between the two IndoXLNet and IndoBERT-base-p1 variants are listed in Table 6.

Table 6. Hyperparameter differences between IndoXLNet and IndoBERT-base-p1

Hyperparameter	IndoXLNet	IndoBERT-base-p1
train_batch_size	128	256
learning_rate	1e-5	2e-5
steps	1.2M	1M
vocab_size	32 000	30.522

3.7. Model Evaluation

Performance evaluations of the two IndoXLNet variants will be carried out on four natural language processing problems, including singlesentence classification, singlesentence sequencetagging, sentencepair classification, and sentencepair sequence labeling. The performance obtained by IndoXLNet-4B is compared against the performance obtained by IndoBERT-base-p1 [9], whereas the performance obtained by the IndoXLNet-4Bplus is compared against the IndoXLNet-4B.

Datasets used to evaluate classification tasks are EmoT, SmSA, HoASA, and WReTE. Meanwhile, datasets used to evaluate sequence labeling tasks are POSP, BaPOS, TermA, KEPS, NERGrit, NERP, and FacQA. The evaluation metric used is the F1-score for all given tasks. Details of the datasets used for the model evaluation process are described in Table 7.

Table 7. Dataset Testing used by IndoXLNet [9]

Dataset	Task Description	Number of Labels	Number of Class	Domain	Style
Single Sentence Classification Tasks					
EmoT	emotionclassification	1	5	tweets	colloquial
SmSA	sentimentanalysis	1	3	general	colloquial
HoASA	aspect-basedsentimentanalysis	10	4	hotel	colloquial
Sentence Pair Classification Tasks					
WReTE	textualentailment	1	2	wiki	formal
Single Sentence Sequence Labeling Tasks					
POSP	part-of-speechtagging	1	26	news	formal
BaPOS	part-of-speechtagging	1	41	news	formal
TermA	spanextraction	1	5	hotel	colloquial
KEPS	spanextraction	1	3	banking	colloquial
NERGri	namedentityrecognition	1	7	wiki	formal
NERP	namedentityrecognition	1	11	news	formal
Sentence Pair Sequence Labeling Tasks					
FacQA	spanextraction	1	3	news	formal

For all testing datasets, several experiments are conducted to get the best performance of IndoXLNet-4B. Learning rate and optimizer are the fine-tuning hyperparameters that are tuned. The epoch of the fine-tuning process is decided using an early stop on the F1-score. Seed 42 is used in overall testing datasets to randomize the dataset order and the weight initialization during fine-tuning. The hyperparameters resulting best performance for IndoXLNet-4B were then used to fine-tune the IndoXLNet-4Bplus.

3.7.1. Single Sentence Classification Task

Three datasets will evaluate model performance for this category: EmoT, SmSA, and HoASA. The hyperparameter variations tested for this category are listed in Table 8.

Table 8. Hyperparameters single sentence classification

Hyperparameter	Value
Learning Rate	5e-6, 1e-5, 5e-5, 1e-4, 1e-2
Optimizer	Adam, AdamW, RMSprop, SGD

More[9] is a series of classification datasets to determine the labels of emotions in each input sequence. There are five labels on the target variable, including anger, fear, happiness, love, and sadness.

SmSA[9] is a series of sentiment analysis datasets at the sentence level. This task aims to determine labels on review sentences collected from various sources. There are three labels on the target variable, including positive, neutral, and negative

HoASA[9] is a series of datasets for an aspect-based sentiment analysis task containing hotel reviews. The dataset has ten review aspects, including ac, hot water, smell, general, cleanliness, linen, service, sunrise_meal, tv, and wifi. There are four labels for each aspect: positive, negative, neutral, and positive-negative.

3.7.2. Sentence Pair Classification Task

IndoXLNet will be tested on the **WReTe**[9] dataset, the sentence pair classification task category. The model will be tested to determine the involvement between pairs of sentences. Two labels state if the meaning in the second sentence can be found in the first sentence or not, and the label is denoted by "Entail or Paraphrase" and "Not Entail."

The hyperparameter variations tested for this category are listed in Table 9.

Table 9. Hyperparameters sentence pair classification

Hyperparameter	Value
Learning Rate	5e-6, 1e-5, 5e-5, 5e-3, 1e-2
Optimizer	Adam, AdamW, RMSprop, SGD

3.7.3. Single Sentence Sequence Labeling

Six datasets will evaluate model performance for this category: POSP, BaPOS, TermA, KEPS, NERGrit, and NERP. The hyperparameter variations tested for this category are listed in Table 10.

Table 10. Hyperparameters single sentence sequence labeling

Hyperparameter	Value
Learning Rate	1e-6, 5e-6, 1e-5, 5e-5, 1e-4, 1e-2, 5e-2, 1e-1
Optimizer	Adam, AdamW, RMSprop, SGD

POSP[9] is a series of datasets that fall into the part-of-speech tagging category. There are 26 POS tag labels in this dataset. The objective of this task is to label the word classes of the input sequence sentences.

BaPOS[9] is a series of datasets that fall into the part-of-speech tagging category. There are 23 POS tag labels in this dataset. The objective of this task is to label the word classes of the input sequence sentences.

TermA[9] is a dataset for POS tagging tasks. The objective of this task is to label the word range of the input sequence sentence. There are two types of tags in this dataset: aspect and sentiment.

KEPS[9] is a series of datasets for the keyphrase extraction task. Phrases that contain significant meaning are considered keyphrases, and an input sequence may have one or more keyphrases with different locations.

NERGrit[9] is a set of datasets for named-entity-recognition tasks. There are three entity tags, including PERSON, PLACE, and ORGANIZATION.

NERP[9] is a series of datasets for named-entity-recognition tasks. There are five entity tags, including PER (name of person), LOC (name of location), IND (name of product or label), EVT (name of the event), and FNB (name of food and beverage).

3.7.4. Sentence Pair Sequence Labeling Task

IndoXLNet will be tested on the **FacQA**[9], the sentence pair sequence labeling task category. The model will be tested to find the answers contained in a quote to the given question. The hyperparameter variations tested for this category are listed in Table 11.

Table 11. Hyperparameters sentence pair classification

Hyperparameter	Value
Learning Rate	5e-6, 1e-5, 5e-5, 1e-2
Optimizer	Adam, AdamW, RMSprop, SGD

4. Result and Discussion

This section describes the performance results of the two IndoXLNet variants tested with eleven different datasets contained in the IndoNLU benchmark. The IndoBERT model variant used as a comparison is the IndoBERT-base-p1 model. Details of the performance comparison between IndoBERT-base-p1, IndoXLNet-4B, and IndoXLNet-4Bplus are shown in Table 12.

Table 12. Performance comparison between IndoBERT-base-p1 against IndoXLNet for classification

Model	Classification				
	EmoT	**SmSA**	**HoASA**	**WReTE**	**AVG**
IndoBERT-base-p1	**75.480**	87.730	92.070	78.550	83.458
IndoXLNet-4B	75.026	**91.082**	**92.933**	83.167	**85.298**
IndoXLNet-4Bplus	74.876	89.651	91.826	**83.623**	84.994

Table 13. Performance comparison between IndoBERT-base-p1 against IndoXLNet for classification for sequence labeling

Model	*Sequence Labeling*							
	POSP	**BaPOS**	**TermA**	**KEPS**	**NERGrit**	**NERP**	**FacQA**	**AVG**
IndoBERT-base-p1	95.260	87.090	90.730	70.360	69.870	75.520	53.450	77.470
IndoXLNet-4B	95.772	**89.930**	91.381	**71.498**	74.511	**77.283**	**64.960**	**80.245**
IndoXLNet-4Bplus	**96.841**	89.718	**91.386**	68,657	**75.158**	76.997	62.114	80.124

Table 12 shows that the testing datasets used to test the IndoXLNet-4B and IndoXLNet-4Bplus models in the classification category include EmoT, SmSA, HoASA, and WreTe. The average F1-score obtained by IndoXLNet-4B is higher than the average F1-score IndoBERT-base-p1 and IndoXLNet-4Bplus. It means that the performance of IndoXLNet-4B is better than the performance of IndoBERT-base-p1 in solving NLP problems for classification tasks. In addition, for classification tasks, adding more datasets in the pre-training process does not have a good impact on IndoXLNet-4B because the average F1-score obtained by IndoXLNet-4Bplus is not better than IndoXLNet-4B. Thus, we conclude that the pre-trained model with the XLNet architecture (IndoXLNet-4B) can solve NLP problems in Bahasa Indonesia better than the pre-trained model with the BERT architecture (IndoBERT-base-p1) for the classification tasks. However, the addition of a pre-training dataset to the model with the XLNet architecture (IndoXLNet-4Bplus) is not proven to improve the model's performance in general for the classification task category.

Table 13 shows that the testing datasets used to test the IndoXLNet-4B and IndoXLNet-4Bplus models in the sequence labeling tasks include POSP, BaPOS, TermA, KEPS, NERGrit, NERP, and FacQA. The average F1-score obtained by IndoXLNet-4B is higher than the average F1-score IndoBERT-base-p1 and IndoXLNet-4Bplus. It means that the performance of IndoXLNet-4B is better than the performance of IndoBERT-base-p1 in solving NLP problems for the sequence labeling tasks. Similar to the classification tasks, there was no increase in the average F1-score from IndoXLNet-4Bplus in sequence labeling. The addition of datasets in the pre-training process for sequence labeling does not positively impact IndoXLNet-4B. Thus, we conclude that the pre-trained model with the XLNet architecture (IndoXLNet-4B) can solve NLP problems for Bahasa Indonesia better than the pre-trained model with the BERT architecture (IndoBERT-base-p1) for the sequence labeling tasks. However, adding more pre-training datasets to the model with the XLNet architecture (IndoXLNet-4Bplus) is not proven to improve the overall model performance for the sequence labeling tasks.

Based on the model evaluation results in Table 12 and Table 13, the average F1-score performance of IndoXLNet-4B increased against IndoBERT-base-p1 by 3.06%, so that IndoXLNet can capture the context of the representation of words better than IndoBERT. Thus, this model could be relied upon to solve various sentiment analysis problems and named-entity recognition in Natural Language Processing, specifically for Bahasa Indonesia.

4.1. Classification

In the classification tasks, the average F1-score of IndoXLNet-4B exceeds the average F1-score of IndoBERT-base-p1. IndoXLNet-4B can outperform IndoBERT-base-p1 in all testing datasets except EmoT. In EmoT, IndoBERT-base-p1 is still above the performance of IndoXLNet-4B. We tried to analyze the performance of IndoXLNet-4B for EmoT in terms of label distribution and the number of datasets.

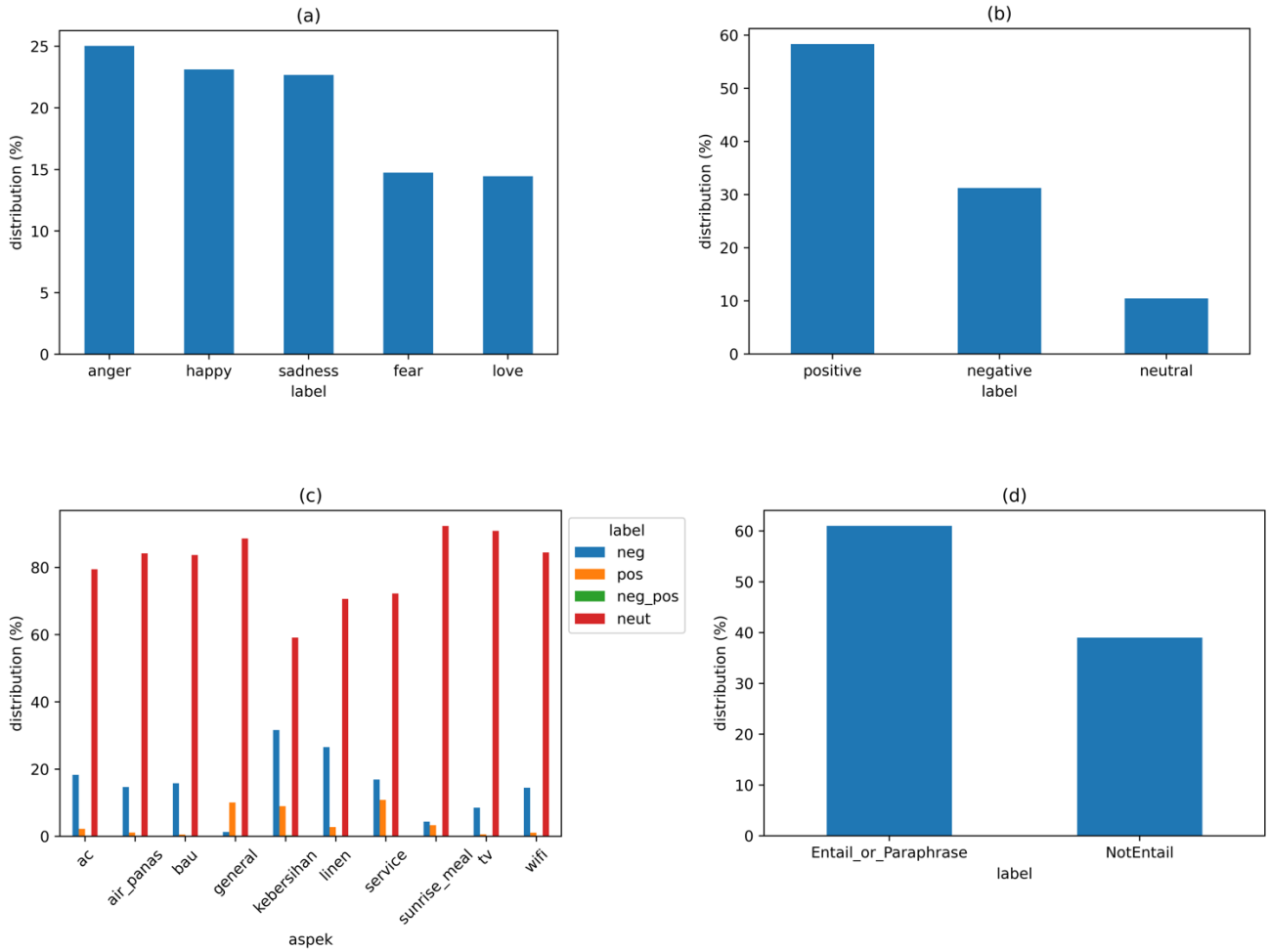

Fig. 3 Label distribution of eachclassification testing dataset; (a) EmoT; (b) SmSA; (c) HoASA; (d) WreTe

Fig. 3presents the label distribution for each classification dataset. Some of the labels on the testing dataset are in English, but the text on the testing dataset used for the model training process is in Bahasa, Indonesia. The use of labels in English does not affect the model training process because all labels will be converted into numeric codes during the training process. In addition, all the testing datasets used to evaluate the two variants of the IndoXLNet model were also used to evaluate IndoBERT [9], so that the testing datasets can be used to evaluate both IndoXLNet variants.

Fig. 3(a) shows the label distribution for the EmoT dataset. There are five different labels: anger, joy, sadness, fear, and love. The distribution of labels in the EmoT dataset is unbalanced because the fear and love labels are far below the other labels. Fig. 3 (b) shows the label distribution for the SmSA dataset. There are three different labels which include positive, negative, and neutral. The distribution of labels in the SmSA dataset is unbalanced because the proportions between the three labels are far apart and not evenly distributed. Fig. 3 (c) shows the distribution of labels for each aspect of the HoASA dataset. Unlike other classification datasets, the HoASA dataset has ten different aspects, and each aspect has its distribution of labels. Labels on each aspect include negative, positive, negative and positive, and neutral. The distribution of labels on each aspect of the HoASA dataset is unbalanced. The proportion of neutral labels on each aspect dominates the other labels. Fig. 3 (d) shows the label distribution for the WreTe dataset. There are only two labels on the WreTe dataset, which are entailed_or_paraphrase and not_entailed. The label distribution in the WreTe dataset is unbalanced because the proportions of the two labels are not evenly distributed.

Unbalanced label distribution is found in EmoT and other datasets for classification tasks, as shown in Fig. 3. In addition, inTable 7, the number of the EmoT dataset is higher than the HoASA and WreTe datasets. Still, the performance of IndoXLNet-4B can outperform the performance of IndoBERT-base-p1 on HoASA and WreTe. It means that the cause of IndoXLNet-4B's performance could not exceed IndoBERT-base-p1 for EmoT is not related

to the distribution of labels and the number of datasets but rather the character IndoXLNet-4B itself.

When compared to IndoXLNet-4B, IndoXLNet-4Bplus provides the best performance when tested using the WReTe testing dataset, but its performance is lower for other testing datasets other than WReTe. It means that the addition of the pre-training dataset does not positively impact IndoXLNet's performance for classification tasks.

4.1.1. Single Sentence Classification

The performance of IndoXLNet-4B exceeds the performance of IndoBERT-base-p1 on the SmSA and HoASA testing datasets but not with the EmoT testing dataset. In addition, IndoXLNet-4Bplus cannot outperform the performance of IndoXLNet-4B. It means that IndoXLNet-4B, in general, can outperform IndoBERT-base-p1 for the Single Sentence Classification category. However, adding a pre-training dataset does not positively impact IndoXLNet's performance for the Single Sentence Classification task.

The experimental results for single sentence classification are shown in Table 14. Based on the experiments conducted for this category, if using AdamW, Adam, or RMSprop as the optimizer, the best F1-scores are obtained when setting the learning rate at 5e-6 or 1e-5. However, using these optimizers for HoASA, the best F1-scores obtained are always at a higher learning rate, at 5e-5.

The best performance of IndoXLNet-4B is obtained by setting the hyperparameter, as shown inTable 15. IndoXLNet-4Bplus also uses this parameter.

Table 14IndoXLNet-4B experimentalresults for single sentence classification

Parameter		Dataset		
Optimizer	**Learning Rate**	**EmoT**	**SmSA**	**HoASA**
AdamW	5e-6	72.855	88.576	89.373
	1e-5	72.907	89.148	91.956
	5e-5	70.813	87.159	***92.933***
	1e-4	68.975	82.210	89.899
Adam	5e-6	72.855	89.593	89.799
	1e-5	72.907	89.783	91.879
	5e-5	70.978	86.901	92.468
	1e-4	65.109	82.784	90.690
RMSprop	5e-6	***75.026***	*	91.245
	1e-5	70.004	86.521	92.605
	5e-5	*	83.528	91.190
SGD	1e-2	72.200	***91.082***	92.068

*Not Performed

Table 15. IndoXLNet hyperparameter for single-sentence classification

Parameter	Dataset		
	EmoT	**SmSA**	**HoASA**
Learning Rate	5e-6	1e-2	5e-5
Optimizer	RMSprop	SGD	Adam

4.1.2. Sentence Pair Classification

IndoXLNet-4B's performance exceeds the performance of IndoBERT-base-p1 in the Sentence Pair Classification (WreTe dataset) task category. In addition, IndoXLNet-4Bplus produces higher performance compared to IndoXLNet-4B. It means that the performance of IndoXLNet can outperform the performance of IndoBERT-base-p1 in this task category. Also, the addition of the pre-training dataset has a good impact on the performance of the IndoXLNet model, specifically for tasks in this category.

The experimental results for sentence pair classification are shown in Table 16. Based on the experiments conducted for this category, the optimizer and learning rate setupdo not show a pattern for the results obtained.

The best performance of IndoXLNet-4B is obtained by setting the hyperparameter, as shown in Table 17. IndoXLNet-4Bplus also uses this parameter.

Table 16. IndoXLNet-4B experimental results for sentence pair classification

Parameter		Dataset
Optimizer	**Learning Rate**	**WreTe**
AdamW	5e-6	76.767
	1e-5	***83.167***
	5e-5	79.255
Adam	5e-6	77.072
	1e-5	79.647
	5e-5	79.255
RMSprop	5e-6	79.647
SGD	5e-3	79.506
	1e-2	81.663

Table 17. IndoXLNet hyperparameter for sentence pair classification

Parameter	Dataset
	WreTe
Learning Rate	1e-5
Optimizer	AdamW

4.2. Sequence Labeling

In the sequence labeling task's average F1-score, IndoXLNet-4B can exceed the average F1-score IndoBERT-base-p1, as shown in Table 13. IndoXLNet-4B

can exceed the performance of IndoBERT-base-p1 on all testing datasets in the sequence labeling category. The average F1-score obtained by IndoXLNet-4B is 80,245, while the average F1-score obtained by IndoBERT-base-p1 is 77,470. In addition, the average F1-score of IndoXLNet-4Bplus (80,124) can also exceed the average F1-score of IndoBERT-base-p1. It confirms that the pre-trained model with XLNet architecture can solve NLP problems in Bahasa Indonesia better than the pre-trained model with the BERT architecture for the sequence labeling category.

The performance of IndoXLNet-4Bplus can only exceed IndoXLNet-4B when tested on POSP, TermA, and NERGrit, which means adding a dataset to the pre-training process is not proven to improve IndoXLNet's performance in general for the sequence labeling category.

4.2.1.Single Sentence Sequence Labeling

IndoXLNet-4B and IndoXLNet-4Bplus exceed the performance of IndoBERT-base-p1 on all testing datasets in this category. It means that the performance of IndoXLNet can outperform the performance of IndoBERT-base-p1 in this task category.

The experimental results for single sentence sequence labeling are shown in Table 18. Based on the experiments conducted for this category, the optimizer and learning rate setup donot show a pattern for the results obtained.

The best performance of IndoXLNet-4B is obtained by setting the hyperparameter, as shown in Table 19. IndoXLNet-4Bplus also uses this parameter.

Table 18. IndoXLNet-4B experimental results for single sentence sequence labeling

Parameter		Dataset					
Optimizer	**Learning Rate**	**BaPOS**	**POSP**	**NERGrit**	**NERP**	**TermA**	**KEPS**
AdamW	1e-6	*	*	64.123	*	88.887	68.293
	5e-6	84.389	95.669	71.975	75.002	90.700	68.268
	1e-5	86.374	95.710	72.335	76.957	***91.381***	69.180
	5e-5	86.401	*	71.623	75.765	90.668	70.013
	1e-4	*	*	68.544	73.557	89.895	69.772
Adam	5e-6	86.200	95.809	71.301	77.003	*	69.452
	1e-5	85.965	*	72.334	76.396	90.919	70.014
	5e-5	85.253	*	69.652	74.795	89.817	70.190
	1e-4	84.088	*	66.004	72.053	89.725	*
RMSprop	5e-6	82.497	***95.772***	73.066	76.206	91.172	70.202
	1e-5	86.284	95.715	***74.511***	75.657	90.882	***71.498***
	5e-5	***89.930***	*	71.972	72.000	*	*
SGD	1e-2	75.584	95.739	*	***77.283***	90.744	*
	5e-2	*	*	*	*	*	70.530
	1e-1	*	*	*	*	*	70.583

*Not Performed

Table 19. IndoXLNet hyperparameter for single-sentence sequence labeling

Dataset	Parameter	
	Learning Rate	**Optimizer**
BaPOS	5e-5	RMSprop
POSP	5e-6	RMSprop
NERGrit	1e-5	RMSprop
NERP	1e-2	SGD
TermA	1e-5	AdamW
KEPS	1e-5	RMSprop

4.2.2.Sentence Pair Sequence Labeling

IndoXLNet-4B and IndoXLNet-4Bplus exceed IndoBERT-base-p1 for this category. It means that the performance of IndoXLNet can outperform the performance of IndoBERT-base-p1 in this task category.

The experimental results for single sentence sequence labeling are shown inTable 20. Based on the experiments conducted for this category, the optimizer and learning ratesetupdo not show a pattern for the results obtained.

The best performance of IndoXLNet-4B is obtained by setting the hyperparameter, as shown in Table 17 IndoXLNet-4Bplus also uses this parameter.

Table 20. IndoXLNet-4B experimental results for sentence pair sequence labeling

Parameter		Dataset
Optimizer	**Learning Rate**	**FacQA**
AdamW	5e-6	61.350
	1e-5	61.963
	5e-5	58.735
Adam	5e-6	59.434
	1e-5	59.784
	5e-5	*64.960*
RMSprop	5e-6	61.774
	1e-5	60.062
	5e-5	58.044
SGD	1e-2	55.000

Table 21. IndoXLNet hyperparameter for sentence-pair sequence labeling

Parameter	Dataset
	FacQA
Learning Rate	5e-5
Optimizer	Adam

5. Conclusion

This study created a pre-trained model using the XLNet architecture, which the author calls IndoXLNet. Two variants of IndoXLNet were made, namely IndoXLNet-4B and IndoXLNet-4Bplus. IndoXLNet-4B was trained using the Indo4B dataset, while IndoXLNet-4Bplus was trained using the Indo4B dataset plus web crawl data.

Based on the research we conducted, IndoXLNet-4B, in general, can exceed the performance of IndoBERT-base-p1, which is an equivalent model, in the testing datasets provided by IndoNLU in the category of classification and sequence labeling tasks.

IndoXLNet-4Bplus, when tested on the IndoNLU testing dataset, was able to produce a better performance on specific testing datasets, namely WreTe, POSP, TermA, and NERGrit. It proves that the addition of datasets during the pre-training process of the IndoXLNet model only positively impacted specific testing datasets and, in general, did not improve IndoXLNet's performance.

We suggest developing a similar model with the LARGE architectural variant, which refers to the $XLNet_{LARGE}$ model and uses the hyperparameters $XLNet_{LARGE}$[3] to produce a better pre-trained model to solve NLP problems in Bahasa Indonesia.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

References

[1] T. Young, D. Hazarika, S. Poria and E. Cambria, Recent Trends in Deep Learning Based Natural Language Processing, IEEE Computational Intelligence Magazine. 13(3) (2018) 55-75.

[2] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding, in Comparative Analyses of Bert, Roberta, Distilbert, and Xlnet for Text-Based Emotion Recognition. (2019).

[3] Z. Yang, Z. Dai, Y. Yang, J. G. Carbonell, R. Salakhutdinov and Q. V. Le, XLNet: Generalized Autoregressive Pretraining for Language Understanding, Advances in Neural Information Processing Systems. 32 (2019).

[4] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy and S. R. Bowman, GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding, in Proceedings of the EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP. (2018).

[5] P. Rajpurkar, J. Zhang, K. Lopyrev and P. Liang, SQuAD: 100,000+ Questions for Machine Comprehension of Text, in Proceedings of the Conference on Empirical Methods in Natural Language Processing. (2016).

[6] S. Shahid, T. Singh, Y. Sharma and K. Sharma, Devising Malware Characteristics using Transformers, International Journal of Engineering Trends and Technology. 68(5) (2020) 33-37.

[7] R. Olaniyan, D. Stamate and I. Pu, A Two-Step Optimised BERT-Based NLP Algorithm for Extracting Sentiment from Financial News, in IFIP International Conference on Artificial Intelligence Applications and Innovations. (2021).

[8] X. Liu, G. L. Hersch, I. Khalil and M. Devarakonda, Clinical Trial Information Extraction with Bert, in IEEE 9th International Conference on Healthcare Informatics (ICHI). (2021).

[9] B. Wilie, K. Vincentio, G. I. Winata, S. Cahyawijaya, X. Li, Z. Y. Lim, S. Soleman, R. Mahendra, P. Fung, S. Bahar and A. Purwarianti, IndoNLU: Benchmark and Resources for Evaluating Indonesian Natural Language Understanding, in Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing. (2020).

[10] H. Le, L. Vial, J. Frej, V. Segonne, M. Coavoux, B. Lecouteux and D. & Schwab, Flaubert: Unsupervised Language Model Pre-Training for French., arXiv preprint arXiv:1912.05372. (2019).

[11] L. Martin, B. Muller, P. J. O. Suárez, Y. Dupont, L. Romary, É. V. de La Clergerie, D. Seddah and B. Sagot, Camembert: A Tasty French Language Model, arXiv preprint arXiv:1911.03894. (2019).

[12] S. Lee, H. Jang, Y. Baik, S. Park and H. Shin, Kr-Bert: A Small-Scale Korean-Specific Language Model, arXiv preprint arXiv:2008.03979. (2020).

[13] J. Canete, G. Chaperon, R. Fuentes, J. H. Ho, H. Kang and J. Pérez, Spanish Pre-Trained Bert Model and Evaluation Data, Pml4dc at ICLR. 2020 (2020) 2020.

[14] K. Gaanoun and I. Benelallam, Arabic Dialect Identification: An Arabic-BERT Model with Data Augmentation and Ensembling Strategy, In Proceedings of the Fifth Arabic Natural Language Processing Workshop. (2020).

[15] A. Conneau and G. Lample, Cross-Lingual Language Model Pre-Training, Advances in Neural Information Processing Systems. 32 (2019).

[16] G. Lai, Q. Xie, H. Liu, Y. Yang and E. Hovy, RACE: Large-scale Reading Comprehension Dataset From Examinations, in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. (2017).

[17] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee and L. Zettlemoyer, Deep Contextualized Word Representations, in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 1 (2018).

[18] (2018). A. Radford, K. Narasimhan, T. Salimans and I. Sutskever, Improving Language Understanding by Generative Pre-Training. [Online]. Available: https://www.cs.ubc.ca/~amuham01/LING530/papers/radford2018improving.pdf.

[19] A. F. Aji, G. I. Winata, F. Koto, S. Cahyawijaya, A. Romadhony, R. Mahendra, K. Kurniawan, D. Moeljadi, R. E. Prasojo, T. Baldwin, J. H. Lau and S. Ruder, One Country, 700+ Languages: NLP Challenges for Underrepresented Languages and Dialects in Indonesia, arXiv preprint arXiv:2203.13357. (2022).

[20] L. H. Suadaa, I. Santoso and A. T. B. Panjaitan, Transfer Learning of Pre-trained Transformers for Covid-19 Hoax Detection in Indonesian Language, IJCCS Indonesian Journal of Computing and Cybernetics Systems. 15(3) (2021).

[21] M. N. Fakhruzzaman and S. W. Gunawan, Web-based Application for Detecting Indonesian Clickbait Headlines using IndoBERT, arXiv preprint arXiv:2102.10601. (2021).

[22] A. Marpaung, R. Rismala and H. Nurrahmi, Hate Speech Detection in Indonesian Twitter Texts using Bidirectional Gated Recurrent Unit, in 13th International Conference on Knowledge and Smart Technology (KST). (2021).

[23] A. F. Adoma, N.-M. Henry and W. Chen, Comparative Analyses of Bert, Roberta, Distilbert, and Xlnet for Text-Based Emotion Recognition, in 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP). (2020).

[24] K. R. Scherer and H. G. Wallbott, Evidence for Universality and Cultural Variation of Differential Emotion Response Patterning, Journal of Personality and Social Psychology. 66(2) (1994) 310.

[25] C. Biemann, G. Heyer, U. Quasthoff and M. Richter, The Leipzig Corpora Collection-Monolingual Corpora of Standard Size, in Proceedings of Corpus Linguistic. (2007).