Automatic Indonesian Hoax News Detection Using BERT

Reza Fuad Rachmadi
Department of Computer Engineering
Faculty of Intelligent Electrical
and Informatics Technology
Sepuluh Nopember
Institute of Technology
Surabaya, Indonesia 60111
fuad@te.its.ac.id

Mauridhi Hery Purnomo
Department of Computer Engineering
Faculty of Intelligent Electrical
and Informatics Technology
Sepuluh Nopember
Institute of Technology
Surabaya, Indonesia 60111
hery@ee.its.ac.id

Aufa Nabil Amiri
Department of Computer Engineering
Faculty of Intelligent Electrical
and Informatics Technology
Sepuluh Nopember
Institute of Technology
Surabaya, Indonesia 60111
aufa.17072@mhs.its.ac.id

***Abstrak*—Fake news or called hoax, is one of the things that still plaguing Indonesia. Even more so, with the rise of the social media, a fake news can spread wider and faster than ever before. Worse, Indonesian people have quite a high tendencies to share fake news. That is why, we are in a dire need of a method to detect fake news. This research is using BERT to automatically classifiy whether a news is a hoax or not. From a raw text, we applied a tokenization process before inputting the text to the BERT. Next, the pooled output of the BERT is being used as the input for Linear Regression, a tested-and-true method for classifying task. The output of the Linear Regression is then being used as a way to determine whether a news is a hoax or not. The purpose of this research is to create a machine learning model to help the people to determine whether a text is a fake news or not. The result of this research is a model to classify a hoax text with 89% in accuracy.**

***Kata kunci*—BERT, Hoax, Fake News Classification, Linear Regression**

I. Introduction

News is a report or a factual story, designed to be the fastest, has a good way of describing problems, and is just by nature to all problem in which it is choose to be published [1]. News also has a very important role in the public, not only because it is a good way to attain a new information, but also to broaden one knowledge.

Hoax or fake news is a way or method to try to deceive people so they believed something that is can't be considered correct and those incorrect things is more often than not is something only a mad-man would believe [2]. Not only reading a fake news will cost you your knowledge, hoax can have many other effects, ranging from the loss of reputation, money, up to even death threat.

Based on the data that we got from the Ministry of Communication and Informatics, there are a total of 5156 hoaxes that have been found only from a short range of August 2018 to March 2020. From January 2020 to March 2020, there are as many as 959 fake news that have been found [3]. Still based on the very same source, at Juny 2020, there are dozens of new hoaxes have been discovered every single day [4].

Nowadays, there are high chance everyone has a few social media account rather than those that are not. This in turn, has quite an effect on the spreading of the fake news, with those hoaxes spreading far and wide with speed never seen before. Based on the survey conducted by Khan and Idris, there are more than half of Indonesian people has a high tendency to share news links without feeling the need to do any validation of said news beforehand [5]. Another survey with similar topic conducted by Kunto with 480 respondence at East Java, shows that around 30% of the total of the responder has a tendency to share a fake news from mild to severe [6]. From those studies, it is safe to assume that Indonesian people in general, has a high tendency to share fake news through their social media accounts.

Neural networks is one of the many branches of machine learning study in which it is applying neurons, just like those that is usually found in human brain structure. Those neurons is used by neural network to process data which in turn resulting in an output. One of the newest things in neural network branch is a method called Bi-Directional Encoder Representations from Transformers or BERT in short. BERT is a method to get a context from a raw text in which it is inputted.

There are many previous works on this automatic hoax detection topic that have been done by other researchers in the past. Aggarway et al. has done an extensive research to see the difference between BERT, XGBoost and LSTM to classify fake news from english sources. Based on that research, turn out BERT has quite an edge to detect hoaxes compared to the other two method [7]. Another researcher under the name Bahad et al. has done another research to see which one is better between CNN, RNN, uni-directional LSTM RNN and bi-directional LSTM RNN when used also as classifier tool to detect fake news. The result shows that LSTM coupled with attention-span, whether it is a uni-directional or bi-directional one, has quite a high accuracy compared to the other method like CNN or RNN [8]. From either of those two researches, it can be concluded that if an algorithm is able to "remember" or know the context of the text, it will most likely has a higher accuracy if compared to the other non-"remember" approach.

But, if we are talking about Indonesia news detection state nowadays, there are not that many researcher has been doing that topic. There has been a research, done by Prasetijo et al. , that try to use SVM and SGD to detect Indonesian hoax news and resulting in a model with the accuracy of 85% [9]. Another research by Rahutomo et al. on the same topic but using naive bayes as the method, has been succesfully attain 80% accuracy on the same task [10].

The purpose of this research is to develop a model to automatically detect Indonesian hoax news by using BERT. The reason being is that by using BERT, hopefully, there will be an increase in efficiency and accuracy of Indonesian hoax

news detection.

II. Design and Implementation

This research is explaining about the implementation of one of the branch of deep learning studies with the aim to automatically detect Indonesian hoax news by leveraging BERT method. This detection method is trained by using a combination of dataset from https://data.mendeley.com/datasets/p3hfgr5j3m/1 and dataset that we made ourself for this paper alone by using web crawling technology. Picture 1 is the outline of this research in a nutshell.

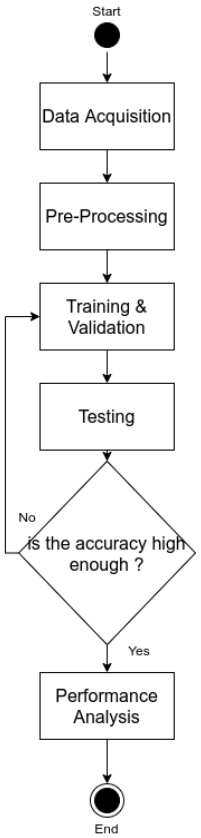

Fig. 1. This research method in a nutshell.

A. Material and Tools Specification

The dataset that is being used in this research is a dataset originated from https://data.mendeley.com/datasets/p3hfgr5j3m/1 coupled with our own made dataset in which we have create it using web crawling technology. Both of these dataset combined, is resulting in total of 1621 data with the exact details can be seen at table II. Meanwhile, table I is the starting point of our dataset which we gotten from https://data.mendeley.com/datasets/p3hfgr5j3m/1 alone.

Each of these dataset is containing the content of the news along with its label which can be either "Valid" or "Hoaks". We took the news from accredited and verified news sources for the valid news, while on the other side, we took all of the hoax news mostly from https://turnbackhoax.id, a website that contains the list of user reported hoax news from many sources.

B. Data Acquisition

Because the dataset that we get from https://data.mendeley.com/datasets/p3hfgr5j3m/1 feel severely lacking for our purpose because it only consist of 600 data, and because there are

TABLE I
Total of news from data.mendeley.com

Label	Total Data
Hoaks	228
Valid	372
Total	**600**

TABLE II
Total of training dataset

Label	Total Data
Hoaks	885
Valid	736
Total	**1621**

no web crawling which outputting its result into a convenient CSV file from Indonesian news sites, we took on our hand a task to create a webcrawling program to take news content from many Indonesian news sites, those sites included but not limited to liputan6.com, detik.com, tempo.com and others. As all of those sites is rightfully accredited and verified by the government, it is used for our valid news dataset. Our hoax news site however, only has one source from turnbackhoax.id, this is mainly because said site has quite an active forum behind it in which lots of people can report their finding of hoax text, seen and checked by lots of other people, before lastly, will be uploaded to the turnbackhoax.id site. But, the biggest factor in choosing that site compare to others is mainly because turnbackhoax.id wrote the original hoaxes text in their website, this coupled with the fact that their website has some kind of structure into it has shorten our task significantly. For this research, the webcrawling process has took news from varied dates, ranging from April 2018 as the oldest to April 2021.

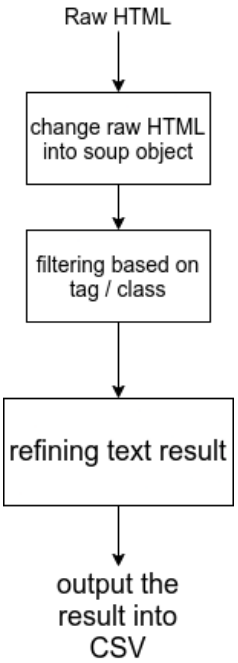

Fig. 2. Garis besar alur program *web crawl*.

Picture 2 is the outline flow of the webcrawling program. Starting with inputting raw HTML code into the program, changing said code into an easier-to-process objects, get the news text and do some post-cleaning on the text, lastly, create

TABLE III
DATASET SAMPLE

news	tagging
Wakil Gubernur DKI Jakarta Sandiaga Uno menargetkan pengerjaan tahap awal Stadion BMW dilakukan pada Oktober. Stadion ini diperuntukkan bagi klub Persija....	Valid
"Komisi II bersama KPU dan Bawaslu masih membahas ketentuan wajib cuti bagi petahana presiden yang maju Pilpres 2019. Mekanisme pengambilan.....	Valid
Jaksa penuntut Ulumum (JPU) pada Komisi Pemberantasan Korupsi (KPK) mencecar Pejabat Pembuat Komitmen (PPK) reguler pada Direktorat Perlindungan Sosial Korban Bencana Sosial Kemensos Victorious Saut Hamonangan Siahaan soal...	Valid
"Halo Kak! Aku Winda Dari Team Giveaway BAIM WONG Anda Memenangkan Hadiah Uang 100Jt dari kami info klik: https://wa.me/+6285796306857"	Hoax
"Apa yang terjadi dengan hewan dalam penelitian? Teknologi ini telah dicoba pada hewan, dan pada hewan penelitian yang dilakukan, semua hewan mati , tidak langsung dari suntikan...	Hoax
"Kadrun istilah dr PKI alias KOMUNIS ditujukan buat islam. Kl mau jd komunis pake aja istilah kadrun buat umat islam. Auto lsg Komunis"	Hoax

a .CSV file to store all of the obtained news text with the appropriate format.

```
...
<div class="detail__body␣itp_bodycontent_wrapper">
<div class="detail__body-text␣itp_bodycontent">

<strong>Jakarta</strong> - Koalisi <a href="https://
detik.com/tag/jokowi" target="_blank">Jokowi</a>
sedang menyusun visi-misi jagoannya. Setelah
menerima masukan dari <a href="https://detik.com/
tag/muhammadiyah" target="_blank"> Muhammadiyah</a>,
 ...
Dan kita pun membuka diri untuk menerima
masukan untuk penyempurnaan,"␣imbuhnya.<br><br><!--
s:parallaxindetail--><div␣class="clearfix"></div><style>
...
```

Listing 1. Penggalan Kode Sumber HTML detik.com.

Firstly, we need to determine tag or class of the HTML code for our first filter. If we look into listing 1 as a reference, we can see `detail__bodytext` class is the one that containing our desired news text. We filtered that class by inputting the class name into the appropriate parameter.

More often than not, our filtering result will contain some garbage or unrelated text resulting in the need to refine it further by post-clean it after the filter process. Usually, those text is writer or editorial notes, ad, or related news links which we don't need at all.

Finally, the last step is outputting all of the acquired news text as a .CSV file. There are no particular reason on the article of why we chose CSV file format compared to other famous Copy of file format aside from the CSV file format is easier to use in our training program and because it is an open-format that can be opened and edited if need be, by nearly any spreadsheet program.

As the general interface and improving user experience for our webcrawling software, we use a .json format file to configure what news sources that we want to get, how much is it, and when is it. All of those configuration will be processed by the program and the program will take the news in accordance with said configuration.

C. Preprocessing

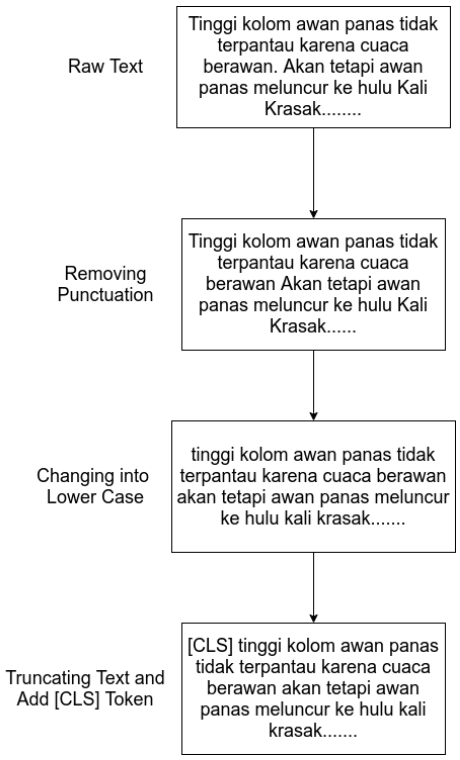

Fig. 3. Preprocessing Method

In this particular process, data need to be prepared first thing before being processed into BERT. This data preprocessing method is consisted of removing any punctuation in the text, change all of the capital letter into its lower letter and truncate any texs that is longer than the capacitites of word that BERT could process at once that is at 512 words or token. There are a few options on how to truncate the text, for example we can take the first 512 words and delete the rest, we can take the last 512 words, or we can combine from both the start of the text and the end portion of the text with some ratio. Last step of this preprocessing is to add a special `[CLS]` token. Picture 3 will explain the same thing but with a better clarity.

Other than that, we will also divide the dataset into 3 portionss with details stated below :

- 70% Training, 10% Validation, 20% Test

1) Training
   This set is used with BERT as an input when it is in its training phase so we can get an optimized model for our task.
2) Validation
   This set is used right after BERT finished its training phase. Used to determined whether our created model has appropriate weight for our task or if our model still need to be trained again. This portion is also used to determine whether our model is overfitting or underfitting which is a bad thing.
3) Test
   This set is used as an accuracy test after both the validation and the training phase is finished. The resulting accuracy of this set is the one that we consider as our result.

To make it clearer, check table IV. We can see based on this table that the divition of the dataset is already appropriate.

TABLE IV
DATASET PORTIONING DETAILS

Bagian	Hoaks	Valid	Total Data
Training	647	519	1166
Validasi	85	78	163
Pegujian	153	139	292
Total			**1621**

D. The Architecture of BERT

BERT is one of the latest machine learning implementation at this time especially for Natural Language Processing (NLP) task. It is based on the Transformer implementation that is based on a previous research by Vaswani et al. [11]. BERT has successfully achieved a higher accuracy than ever before in understanding the context of a raw text if compared to other transformer implementation.

One of the distinct feature of BERT is in the way it is pre-trained. There are 2 steps for pretraining BERT. The first is by doing a Masked Language Model (MLM) in which BERT will be given masked text A and some words B that can be the correct word for the masked text or not as an input, and it will need to predict whether the word B is the correct word for the masked part in text A. This way, BERT will be able to "learn" the relationship between words. The next steps for pretraining BERT is to do some Next Sentence Prediction (NSP) task. The inputted text of this task is 2 sentence, sentence A and sentence B and BERT task is to predict whether these 2 sentences will form a complete paragraph or not. By doing NSP tasks, BERT should be able to get the relationship between sentences easier.

In this research, we decided to use fine-tuning approach, what this mean is that we use a pre-trained BERT model rather than create our own BERT model from scratch, however, we still need to connect the last layer of BERT into a classification layer. In this case, we chose Linear Reggression as the classification layer. For greater detail, figure 4 is the architecture of BERT that will be used throughout this research.

E. Training and Validation

At this stage, the raw text that will be inputted into BERT has already going through its preprocessing phase and is now going into a process called Tokenizer. Tokenizer is a process to change words in a text into token according to its word embedding that is already obtained beforehand when BERT is still in its pretraining phase. Only after all of these process has done, BERT will start its training phase based on the tokenized and preprocessed data.

Not all of the output of the BERT is being used in this particular research, we only need the content of the `[CLS]` token that is filled with the pooled output of all the other tokens and layers. The content of the `[CLS]` token is then inputted into Linear Regresson method. This method is choosen as it is easy enough while still retain quite good accuracy. Figure 5 is the training method in a nutshell.

There are also some parameters that we can adjust only in this stage, namely batch, learning rate, and epoch. Batch is a parameter to adjust how much data is being processed at once per iteration, mind you that there are usually a few iteration per epoch. By adjusting the batch values, the higher it is, the faster the training process is but at the cost of the memory usage. Thanks> to BERT method having quite a large number of layers (718 layers, in general) it can be considered quite heavy, hence we are set the batch value at 10.

Epoch is how much training and validation will take place before the training phase is considered as final. This parameter is one of the most important parameters to adjust as it has a direct effect on the accuracy and the loss of our model. If our loss is too high but the accuracy is too low, it is a clear indication that our model is suffering from underfitting state, meanwhile, if our loss is too low while the accuracy is too high we still need to check if our model is actually good or if it is suffering from overfitting. As our goals in this research is only to process text that is considered to be easier compared to processing image or video, we only set the epoch value to 10.

Learning rate is how much hyperparameter is allowed to change while the model is still in training process, this in turn will change the weight of the layers while in the same process based on the feedback gotten from validation phase. We decided to use the recommended value of 0.00002 [12]

The validation process is used as a way to get the loss validation value that we can use as a comparator between the loss value that we are able to obtain from the training process and the loss validation value that we get from this process. If the loss validation value is getting higher but coupled with a loss training value that shows sign of going lower still, it is a surefire way to know that our model is suffering from overfitting. In another note, if both of our loss value in our model is quite high, then there is a high chance our model is underfitting. Both cases indicate that our model can be further optimized and requiring more training while adjusting the parameter.

F. Testing

After going through validation and training phase, lastly, we need to test our newly created model. Based on the result of this process, we should be able to conclude whether our model can be considered good enough for our use case, or still can be further adjusted by reconfiguring some of the parameters back at the training phase.

G. Performance Analysis

The last step right after testing is performance analysis on our tested model. This process is quite important to see how our model will fare in real world scenario after it is being implemented. There are a few metrics that we are using to do this process, all of these metrics is considered to be the industry standard in the world that is machine learning industry. Firstly, there are confustion matrix to categorize the prediction result based on the actual label in the dataset into 4 division. The division being True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). We also using Recall, Precision, and F1-Score as our performance metrics in this research.

III. EXPERIMENT

Containing in this section we will explaining the result of our test along with the analysis that we have been in accordance with the system design written at the previous section. Dataset that is being used is a combination of dataset

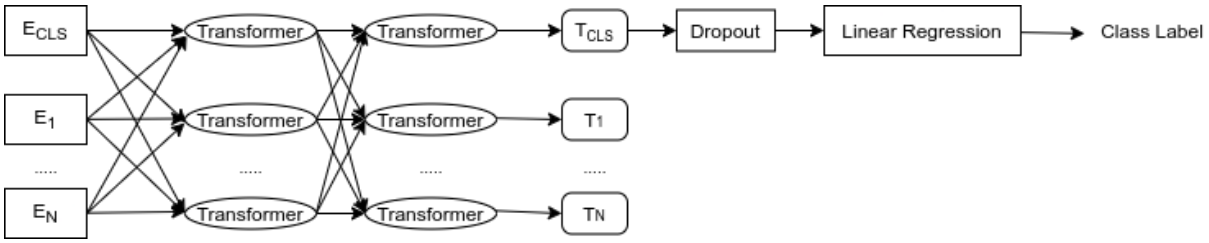

Fig. 4. The Architecture of BERT in this research

Fig. 5. Training Method

that is originated from data.mendeley.com and dataset of our own creation by leveraging web crawling technology. There are a few experiment that we have run in this research with details summarize as below :

nolistsep

1) Performance experiment based on the text truncating method
2) Performance experiment based on the BERT model that is being used
3) Performance experiment based on the transformer method being used
4) Performance experiment based on the training approach

In each of these experiment, all of the models is ran on Google Collab with hardware specification enlisted in table V

TABLE V
PC SPECIFICATION THAT WE USE

Processor	2 v-core Intel(R) Xeon(R) CPU @ 2.20GHz
RAM	Virtual Memory : 12GB
Storage	SSD : 69GB
GPU	Nvidia Tesla T4 16GB
	Nvidia K80 12GB
Operating System	Ubuntu 18.04.5 LTS (Bionic Beaver) 64-bit

A. Performance experiment based on the text truncating method

Because of BERT at the current state is only able to process up to 512 token at once, and because there are a few different styles in writing a news text, we need to test on which way is best to truncate a long text into a maximum of 512 token.

There are a few alternatives that we can choose on how to truncate the text. We can truncate the first 512 token and delete the rest of the text, we can also get the last 512 token, or we can also combining both text from the first part of the text and from the end part of the text according to some ratio. All of those will be tested with details written below :

1) Truncate the first part of the text
   There are a few distinctive feature that can be easily found in most of Indonesian news content. One of the most prominent however, is writing a summary of the presented news on the first few paragraph. Oftenly, this will help people who want to skim the news rather than read it thoroughly and there are many such styles in Indonesian news site, even more so if said sites is using some form of pages when displaying the content of the news. Because of that, on this type of news, it is easier to determine whether it is a hoax or not by reading only the first paragraph.
2) Truncate the last part of the text
   Another characteristics of Indonesian news writing style is placing the conclusion at the end of the text. This style can often be found when the news is having an in-depth review of a particular problem and the conclusion is placed at the end instead of the front to help readers understand how and what is the relationship of all the previously described information in the news.
3) Combining both parts of the text by taking 129 token from the first part, and 383 token from the last part
   This experiment is based on previous work by Chi Sun et al. which stated that the best truncating strategy on long text for BERT method is by combining both text with the said ratio. This strategy has succesfully attain a higher accuracy compared to other truncating strategy like if we only truncate the first part of the text or doing it only with the last part [13]. The reason as to why this is happening is because when we combine both parts, we should get both the preamble of the news and the conclusion part of the news in which then is being inputted into BERT's training phase. But, this research is done in an english long text so there is still the need to see if the same thing hold true for Indonesian long text.

From a total of 1621 data, we split the 18% of it and we set it as a test dataset resulting in the total of 292 dataset only for test phase. We configured all of the training parameters for this experiment to be the same accross test, which is 7 for the epoch, learning-rate is set at 2e-5, and epsilon at 1e-8. We also using the same model accross all test, an indonesian BERT model that has been created by Indobert. For more information regarding the parameters, kindly look into table VI

TABLE VI
PARAMETER CONFIGURATION FOR TRUNCATION STRATEGY BASED TEST

epoch	3
learning rates	2e-5
epsilon	1e-4
model	indobenchmark/indobert-base-p1

The result of this model will be compared to the label that we got from the dataset in which then will be counted to get its confusion matrix, recall, precision, accuracy and f1-score values according to the appropriate formulas.

TABLE VII
PERFORMANCE FOR TRUNCATION STRATEGY BASED TEST

Truncate Location	**recall**	**precision**	**f1-score**	**accuracy**
first part	**89%**	**90%**	**89%**	**89%**
last part	88%	85%	86%	86%
combine (129 first part + 383 last part)	88%	88%	88%	87%

As we can see from table VII, truncating only the first part of the text has obtained the highest accuracy compared to other truncation strategy. On top of that, it also has a balance recall and precision values, indicating that the model is quite good on detecting both the valid news and the hoax news. Truncating only the last part of the text, however, showing high probability of biasing towards detecting all text into hoax news. In another note, The combination of both strategy has a good balance of its precision and recall values, it just not having high enough values if compared to the first strategy.

B. Performance experiment based on the BERT model that is being used

There are lots of BERT models that have been created by many people on the internet. Unfortunately, most of those models is only supporting a specific language. Of course, While there are some that is able to do multilingual tasks, the number is not that great and only a few in between. More often than not, this is because creating a multilingual model require not only massive amount of pre-training time and data, but also the resources it will take even after the model has already finished pre-training and is deployed. Not to mention that the benefit of having a multilingual model is not that great because having only support a specific language will result in a model with a higher accuracy in that language compared to the one with multilingual support. That is why the aim of this experiment is to see which BERT models with different pre-trained data sizes and sources is best for our specific tasks. Below are the details of the models that we used in this subsection.

1) bert-base-bahasa-standard-case *(bert-bahasa)*
   It is a BERT model created by huzeinzol05. By design, this model supposedly only support Malay language, but the creator claimed that this model should be able to do just fine on Indonesian language tasks, thanks to the closeness of both the Malay language and the Indonesian language in which sometime have the same meaning on a few words and structures. This model is trained on quite a lots of data originating from the Malay version of Wikipedia, Wattpad, and also social media [14].
2) bert-base-multilingual-uncaseda *(bert-base)*
   This is a base BERT model that is also being used in the BERT's original paper created by Devlin et al. in which it is first introduced into the world. Created by the team at Google, this model is pre-trained by using all of the languages that Wikipedia have. Resulting in a model that is able to do tasks from all 104 languages at once [15].
3) indobert-base-p1 *(indobert)*
   This BERT model is one of the BERT model that is created specifically for the Indonesian language. This model is the product of Indobenchmark team as a part of benchmarking test for Indonesian language Natural Language Understanding (NLU). Compared to other BERT models, this model has the largest pretained dataset. It is originated from many sources such as Indonesian version of Wikipedia, Twitter, OpenSubtitle. All of those combined, resultinig in 23 GB worth of dataset used only for its pre-training phase.
4) bert-base-indonesian-522M *(cahya-522M)*
   An Indonesian-only BERT model that is the creation of Cahya Wirawan. Pretrained on the lowest dataset size if compared to other models used in this experiment that is of only 522M data. All of which is from the Indonesian version of Wikipedia.
5) bert-base-indonesian-1.5G *(cahya-1.5G)*
   This has model has the same creature as the previous model. The only difference there, is that this model has an additional 1GB of data taken from many Indonesian news sites. The resulting size of the dataset used for pretraining is 1.5G of data.

TABLE VIII
THE CONFIGURATION OF THE BERT MODELS

Model	epoch	dropout	learning rates
bert-base-bahasa-standard-case	4	0.2	2e-5
bert-base-multilingual-uncased	4	0.2	2e-5
indobert-base-p1	3	0.1	2e-5
bert-base-indonesian-522M	3	0.1	2e-5
bert-base-indonesian-1.5G	3	0.2	2e-5

Before we start the training process, we need to configure the parameters of the BERT models. Table VIII is the details of the configuration that is being used in this experiment. There are a couple differences in the configuration like for example the epoch and the dropout values. This is mainly because using the same parameter for all models is considered to be not feasible as there are cases of overfitting or underfitting in some models.

TABLE IX
THE RESULTING PERFORMANCE OF ALL THE BERT MODELS

model	**recall**	**precision**	**f1-score**	**accuracy**	**avg. training time**
bert-bahasa	89%	82%	85%	85%	03:43
bert-base	**97%**	75%	85%	86%	02:07
indobert	89%	**90%**	**89%**	**89%**	02:05
cahya-522M	88%	80%	84%	84%	**02:03**
cahya-1.5G	93%	80%	86%	87%	02:08

Table IX can be summarized with if there is a model that utilized small dataset in its pre-training phase, it will take smaller time at the fine-tuning process, but, this is also sacrificing on the accuracy as it is has lower accuracy compared to other models. Another thing is that the Malay version of the BERT model is not a good match for Indonesian hoax news detection. BERT model created from Indobenchmark has the best accuracy coupled with balanced precision and recall values so it is safe to say that the Indobert model is more reliable when used as a hoax news detection model.

C. Performance experiment based on the transformer method being used

BERT is a state-of-the-art method in Natural Language Processing (NLP) tasks and is a further development of the Transformer method. Meanwhile, aside from BERT there are other models that is based on the same Transformer method with their own advantages and disadvantages. That is why in this experiment, we try to see whether BERT is the best method when is being used as an Indonesian hoax news detection. As a control, we use models that has the same creator - that is Cahya Wirawan - and all of those models is using the same dataset size for its pretraining phase that is 522M. Below are the details of the transformer models that we use in this experiment :

1) ROBERTA
   This method is the better-and-newer version of BERT. According to its inital journal, ROBERTA is trained using dynamic masking method, compared to the static masking method found in BERT. On top of that, the original ROBERTA is trained on a larger dataset compared to BERT and while this add the time required to do the pre-training phase, but also resulting in a more robust model in general [16].
2) GPT-2
   One of the most famous model thanks to its ability to do automatic text continuation or generation while still retaining the context accross sentences and is understandable by humans. The main difference between BERT and GPT-2 is the configuration of its attention head. If we are looking BERT, its attention head is designed to looks both backward and forward to calculate the word bias. Meanwhile, GPT-2 only see the subsequent words.
3) BERT
   The BERT model that we used in this experiment is the same model that we used in the previous experiment which in this case, we choose to use *bert-base-indonesian-522M* by Cahya Wirawan. Eventhough if we see at the previous experiment it just has the littlest accuracy, but this model also has the shortest training time.

TABLE X
THE CONFIGURATION OF THE TRANSFORMER MODELS

Model	epoch	dropout	learning rates
Roberta	3	0.2	2e-6
GPT-2	2	-	2e-5
BERT	3	0.1	2e-5

The details of this experiment configuration for each of the transformer models can be seen at table X. There are a few difference between the configuration of the models so it can reach its optimal state. GPT-2 is using the lowest epoch values as it don't have any dropout layer on its output.

Table XI is the performance result of all the tested models. As we can see based on that table, BERT's f1-score and accuracy value has obtained the highest value compared to the rest, especially on a condition that it is pre-trained on a similar dataset with the same size.

D. Performance experiment based on the training approach

Aside from comparing some models that have been created previously, we also try to improve our models so it can be more optimized for our specific task compared the stock version of said model. We choose to use *indobert-base-p1* model as our base model as it has attained the highest accuracy on the previous experiments if compared to other BERT models. Our approaches is described in detail below :

1) parameter freeze
   Parameter freeze is a technique in machine learning study when we don't want to change any of the pre-set weight that has already attained at pre-training phase. In this experiment, we tried to freeze all of the BERT parameters while still allowing the classifier weight to change accordingly.
2) parameter dropout
   Dropout is a parameter that is being used by the optimizer algorithm as a way to determine how much weight values that is already calculated during the training process to be deleted per epoch. While this practice seems counter-intuitive at first, but shedding some weight values during the training phase will help the model to be more general and not being affected by the change in input. This dropout parameter will also prevent the model from overfitting.

TABLE XI
THE RESULTING PERFORMANCE OF ALL THE TRANSFORMER MODELS

model	recall	precision	f1-score	accuracy	avg. training time
ROBERTA	**90%**	73%	80%	82%	02:12
GPT-2	86%	**81%**	83%	83%	02:15
BERT	88%	80%	**84%**	**84%**	**02:03**

TABLE XII
THE CONFIGURATION OF THE MODEL ON THE TRAINING APPROACH EXPERIMENT

Model	epoch	dropout	learning rates
baseline	3	0.1	2e-5
parameter freeze	50	0.1	2e-5
dropout	3	0.2	2e-5

Table XII contains the parameters that is being used in this experiment. For the parameter freeze experiment, we configured the epoch value to be as high as 50 because even then, the loss value has not reached its overfitting state and is not yet to be in its optimal condition. But, at that point, the loss value is already reaching the state of diminishing of return, henceforth, we decided to stop the experiment at the epoch value of 50.

TABLE XIII
THE PERFORMANCE OF THE MODEL ON THE TRAINING APPROACH EXPERIMENT

model	recall	precision	f1-score	accuracy	avg. training time
baseline	89%	**90%**	**89%**	**89%**	02:05
parameter freeze	**90%**	73%	81%	82%	**00:44**
dropout	83%	88%	85%	84%	03:44

According to the table XIII, we can safely assume that when we experiment with the parameter freeze approach, the average training time per epoch is way shorter compared to the baseline. This is most likely because the optimizer did

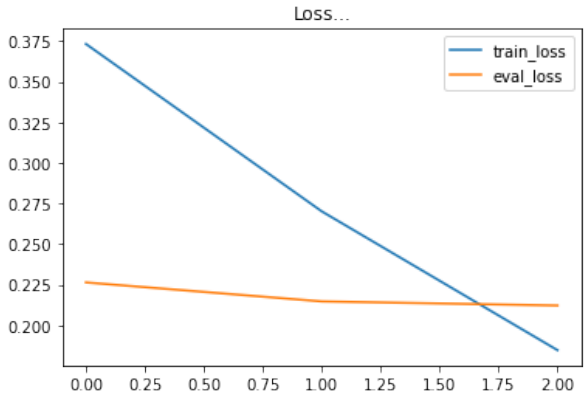

Fig. 6. Baseline loss value

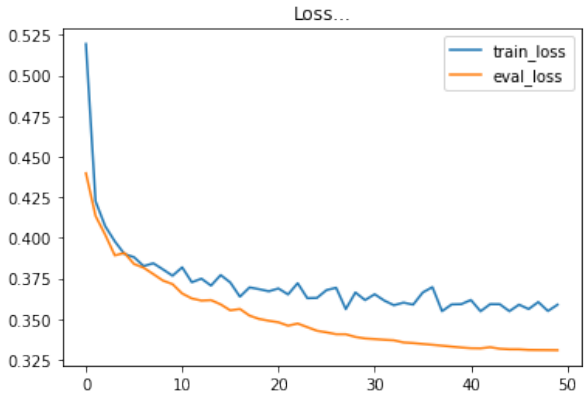

Fig. 7. Loss value with parameter freeze

not need to re-count all of the BERT layer's weight every epoch hence reducing the computational needs and in turn, make the training phase faster. Next, if we are comparing the loss graphic of the baseline model at figure 6, with the loss graphic of the parameter dropout approach experiment at figure 8, it can be said that the baseline graphic and by extension the model, has showing signs of overfitting, while on another side, the loss graphic looks nice with no signs of overfitting at all and all of these is achievable with a little drop in the model accuracy.

IV. Conclusions and Improvements

From the entire experiment, there are a few things in which it can be concluded with has been listed below :

1) The more dataset is being used on its pre-training phase of BERT, the more accurate a model is. This has been proved by model *indobert-base-p1* in which the model has been pre-trained with more than 23GB worth of data.
2) Best truncating method in our approach is by truncating only the first few sentences of an entire news. This approach has successfully obtained the highest accuracy with the the gap of 3% on nearly all metrics if compared to other truncation approach. This is most likely because more often than not, the Indonesian news site is started with a lead or a shorter and denser news content written in a single paragraph, and by taking the starting part of the news text, the lead has been included into the processed text as well.

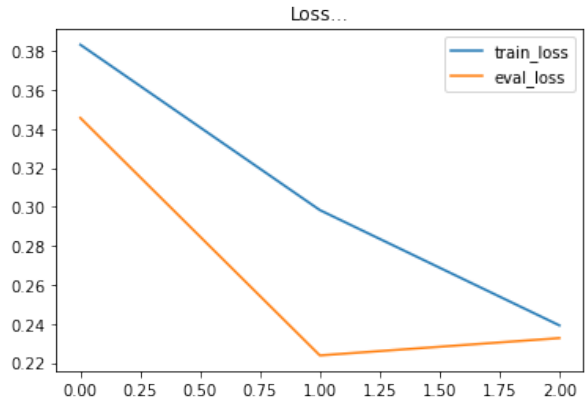

Fig. 8. Loss value with parameter dropout

3) The usage of the BERT model that specifically support Indonesian langauge generally has better accuracy with the difference around 10% on precision metric if compared to the multilingual version of the BERT model and the Malay version of the BERT model especially for Indonesian hoax detection.
4) Eventhough there are lots of other models that is based on the original BERT model itself, but the original BERT model is still good enough for task that is classifying text. This can be seen on our experiment where BERT has a better accuracy value with around 1% gap between BERT and other transformer model. In addition, BERT has the shortest average training time per epoch with around 2 minute, 3 seconds.
5) Because of the complexity, BERT is susceptible to overfititng state, thus, configuring parameter with method such as dropout and parameter freeze can be an immense help to create more robust model while sacrificing very little amount of accuracy of only 5 - 6%.

As no creation of human being is perfect in which this experiment can not escape from, there are a few things that can be improved in this experiment that can be used as a basis for subsequent experiment which should help on improving said experiment results. These improvement is listed in detail as below :

1) Eventhough we have increased the amount of dataset in this experiment, the size is still pale in comparison with its English language counterpart. The larger the dataset pool is, the higher the accuracy should be.
2) One of the major drawbacks in BERT method is the limitation of the token that it can be procesed at once which is only 512 token. There is already another advancement in the NLP field which based on the BERT architecture that remove this limitation completely.
3) What this experiment created is ultimately, only a model that can't be used yet by the people. It would be the best if there is a system that leveraging this experiment result so people can easily use it.

References

[1] Rani and N. L. R. Maha, "Persepsi jurnalis dan praktisi humas terhadap nilai berita," 2013.
[2] Wikipedia, "Berita bohong," diakses 27 November 2020. [Online]. Available: https://id.wikipedia.org/wiki/Berita_bohong
[3] K. Kominfo, "Temuan isu hoaks," 03 2020. [Online]. Available: https://eppid.kominfo.go.id/storage/uploads/2_12_Data_Statistik_Hoax__Agustus_2018_-_31_Maret_2020.pdf

[4] ——, "Laporan isu hoax juni 2020," 07 2020. [Online]. Available: https://eppid.kominfo.go.id/storage/uploads/2_31_Laporan_Isu_Hoaks_Bulan_Juni_2020.pdf

[5] M. L. Khan and I. Idris, "Recognize misinformation and verify before sharing: A reasoned action and information literacy perspective," *Behaviour and Information Technology*, 01 2019.

[6] K. Wibowo, D. Rahmawan, and E. Maryani, "Penelitian di indonesia: umur tidak mempengaruhi kecenderungan orang menyebarkan hoaks," 2019, diakses 27 November 2020. [Online]. Available: https://theconversation.com/penelitian-di-indonesia-umur-tidak-mempengaruhi-kecenderungan-orang-menyebarkan-hoaks-114631

[7] A. Aggarwal, A. Chauhan, D. Kumar, M. Mittal, and S. Verma, "Classification of fake news by fine-tuning deep bidirectional transformers based language model," p. 163973, 04 2020.

[8] P. Bahad, P. Saxena, and R. Kamal, "Fake news detection using bi-directional lstm-recurrent neural network," *Procedia Computer Science*, vol. 165, pp. 74–82, 02 2020.

[9] A. B. Prasetijo, R. R. Isnanto, D. Eridani, Y. A. A. Soetrisno, M. Arfan, and A. Sofwan, "Hoax detection system on indonesian news sites based on text classification using svm and sgd," in *2017 4th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 10 2017.

[10] F. Rahutomo, I. Pratiwi, and D. Ramadhani, "Eksperimen naïve bayes pada deteksi berita hoax berbahasa indonesia," *JURNAL PENELITIAN KOMUNIKASI DAN OPINI PUBLIK*, vol. 23, 07 2019.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[12] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "Indolem and indobert: A benchmark dataset and pre-trained language model for indonesian nlp," *arXiv preprint arXiv:2011.00677*, 2020.

[13] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune bert for text classification?" in *China National Conference on Chinese Computational Linguistics*. Springer, 2019, pp. 194–206.

[14] Z. Husein, "Natural-language-toolkit library for bahasa malaysia, powered by deep learning tensorflow," https://github.com/huseinzol05/malaya, 2018.

[15] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2019.

[16] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019. [Online]. Available: http://arxiv.org/abs/1907.11692