Inform : Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi
Vol.7 No.2 July 2022, P-ISSN : 2502-3470, E-ISSN : 2581-0367

Clickbait Detection of Indonesian News Headlines using Fine-Tune Bidirectional Encoder Representations from Transformers (BERT)

Diyah Utami Kusumaning Putri[1], Dinar Nugroho Pratomo[2]

[1]*Department of Computer Science and Electronics, Faculty of Mathematics and Natural Sciences, Universitas Gadjah Mada, Yogyakarta, Indonesia*

[2]*Department of Electrical Engineering and Informatics, Vocational College, Universitas Gadjah Mada, Yogyakarta, Indonesia*

[1]diyah.utami.k@ugm.ac.id (*)

[2]dinar.nugroho.p@ugm.ac.id

Received: 2022-06-25; Accepted: 2022-07-18; Published: 2022-07-30

Abstract— The problem of the existence of news article that does not match with content, called clickbait, has seriously interfered readers from getting the information they expect. The number of clickbait news continues significantly increased in recent years. According to this problem, a clickbait detector is required to automatically identify news article headlines that include clickbait and non-clickbait. Additionally, many currently existing solutions use handcrafted features and traditional machine learning methods, which limit the generalization. Therefore, this study fine-tunes the Bidirectional Encoder Representations from Transformers (BERT) and uses the Indonesian news headlines dataset CLICK-ID to predict clickbait (BERT). In this research, we use IndoBERT as the pre-trained model, a state-of-the-art BERT-based language model for Indonesian. Then, the usefulness of BERT-based classifiers is then assessed by comparing the performance of IndoBERT classifiers with different pre-trained models with that of two word-vectors-based approaches (i.e., bag-of-words and TF-IDF) and five machine learning classifiers (i.e., NB, KNN, SVM, DT, and RF). The evaluation results indicate that all fine-tuned IndoBERT classifiers outperform all word-vectors-based machine learning classifiers in classifying clickbait and non-clickbait Indonesian news headlines. The $\text{IndoBERT}_{\text{BASE}}$ using the two training phases model gets the highest accuracy of 0.8247, which is 0.064 (6%), outperforming the SVM classifier's accuracy with the bag-of-words model 0.7607.

Keywords— Clickbait, Indonesian News Headlines, Word-Vectors, Machine Learning, Fine-Tuned BERT, Indobert Pre-Trained.

I. INTRODUCTION

The development of the internet is currently beneficial for the communication media field. One is online news sites that can provide information quickly and flexibly. Unfortunately, sometimes the information content does not match the news article's title. Misleading news headlines are an online media tactic to attract readers' attention to the news site and increase visitor traffic. This is what is called clickbait [1].

Clickbait is a way to increase online news media revenue by increasing reader and visitor traffic. The increasing competition between online news media to get readers and provide material benefits contributes to the widespread use of clickbait in online media. Despite the fact that the title and content do not explain the same information or the bombastic title is only explained incompletely, this competition is the root cause of the widespread use of clickbait [2].

Clickbait is characterized by using hyperbole sentences with poor content with little value to attract readers to visit the site. It can impact human psychology and provide an experience that frustrates readers because they do not get the information content they expect. News headlines seem promising, but readers get very little valuable content after visiting the site. The amount of clickbait has recently increased as some news publishers use this technique [3].

According to the clickbait misappropriation problem discussed before, a clickbait detector is needed to identify news article titles that include clickbait and non-clickbait. Several studies have been conducted on clickbait detection of Indonesian news articles. In another research using a sentence scoring algorithm based on word frequency to analyze a match between the headline and article content, B.W. Rauf et al. [4] identified clickbait on the Indonesian news website detik.com.

Several studies used handcrafted features and traditional machine learning algorithms to detect clickbait news. A.F. Yavi [1] uses the Naïve Bayes (NB) method to classify clickbait article titles. The research from R. Sagita et al. [5] used the K-Nearest Neighbor (KNN) method to classify clickbait news. In addition, S. Jumun et al. [6] compared Naive Bayes, Decision Tree, and SVM to classify Thai clickbait headline news. P.S. Hadi et al. [7] have proposed machine learning with extra weight to identify clickbait in Indonesian online news headlines.

This research analyzed four machine learning classifiers and used a few characters that typically appear in a certain label to become an extra weight for the TF-IDF weighting term. The findings conclude that using extra weight improves a maximum of 6% performance. Some existing solutions have been done using handcrafted features for feature extraction and traditional machine learning techniques for predicting class, limiting generalization, and not being effective.

In contrast to the time-consuming and complex feature extraction process, other study uses deep learning to learn

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686

features. A. Agrawal proposed a deep learning model for clickbait detection of headlines using convolutional neural networks (CNN). The developed CNN model performed strongly on the classification and achieved high performance with an accuracy of 0.90 [3]. William and Sari [8] produced an Indonesian news headlines clickbait classification model using Bi-LSTM and CNN models. This research also introduced CLICK-ID, a dataset of 15000 headlines from Indonesian news articles that were collected from 12 Indonesian online news publishers and annotated for clickbait or not. The findings demonstrate that the Bi-LSTM model achieves favorable performance in classifying clickbait. The researchers [9] used a modified backpropagation neural network to classify clickbait using article titles and then compared it to standard algorithms. The results show that the modified algorithm has higher precision, recall, and F1-score than the standard algorithm. M.A. Shaikh and S. Annappanavar [10] conducted research for classifying social media article headlines that are clickbait. This study developed a CNN model, which will then be compared with the RF model. The results indicated that the CNN model outperformed others in identifying content with clickbait headlines, with an accuracy rate of 0.82.

M. Bilal and A.A. Almazroi [11] used a generalized approach to identify helpful and unhelpful reviews from Yelp shopping reviews by fine-tuning the BERT base model. The bag-of-words approaches with machine learning classifiers including KNN, Naïve Bayes, and SVM were compared. The results demonstrated that BERT-based classifiers with fine-tuning outperform bag-of-words techniques. S. Gonzalez-Carvajal and E.C. Garrido-Merchan [12] used several scenarios with varying languages and dataset sources to evaluate BERT, and the conventional TF-IDF vocabulary fed to machine learning algorithms. The experiment results added empirical support for using BERT as the default technique in the NLP problem by demonstrating its superiority and independence from NLP problem aspects.

The BERT method is a state-of-the-art deep learning approach for Natural Language Understanding (NLU) problems based on transformers. This method proposed a faster architecture to train a language model that eliminates recurrences by using a multi-head attention layer. More rapid development, lower data requirements, and improved performance are advantages of employing BERT [13]. B. Willie et al. [14] introduced the first-ever comprehensive resource for training, evaluating, and benchmarking Indonesian NLU (IndoNLU) tasks. This resource includes twelve tasks, ranging from the classification of a single sentence to the labeling sequences of sentences with different complexity levels. This study also provides a collection of pre-trained Indonesian models called IndoBERT. These models were trained using Indo4B, a large, clean Indonesian dataset. This dataset was obtained from publicly accessible websites, social media posts, blogs, and news articles.

The aim of this paper use fine-tuning IndoBERT to identify clickbait and non-clickbait from the Indonesian news headlines dataset CLICK-ID [8]. The BERT tokenizer generated contextualized token embeddings. Thus, no handcrafted features are required. Additionally, the performance result of IndoBERT-based classifiers is compared to different word-vectors-based techniques, i.e., bag-of-words (BoW) and Term Frequency – Inverse Document Frequency (TF-IDF), and numerous classifiers, such as Naïve Bayes, KNN, SVM, Decision Tree, and Random Forest. Therefore, this study also examines how various IndoBERT pre-trained models impact how well a fine-tuned IndoBERT is for the clickbait classification problem.

The remaining of this paper is organized into sections. Section 2 explains the research methodology. The results and discussion are given in Section 3. Finally, the study is concluded in Section 4.

II. Research Methodology

A. Dataset

The dataset used for this study is called CLICK-ID [9], a dataset of news headlines from Indonesian news articles annotated for clickbait or not. There are 15000 annotated articles, with 6290 clickbait and 8710 non-clickbait labels. Figure 1 depicts the ratio of non-clickbait and clickbait classes.

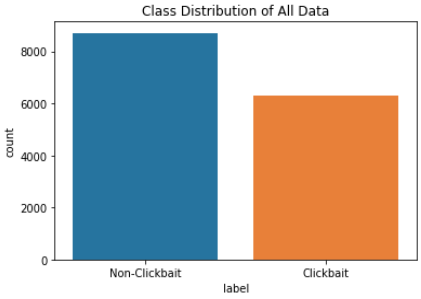

Figure 1. Distribution Of Clickbait and Non-Clickbait Classes for All Data.

A stratified sample strategy is used to split the entire dataset into training and testing datasets, with 12000 (80%) of the news headlines used for training and the remaining 3000 (20%) used for testing. Then, to fine-tune BERT, 2400 (20%) of the training dataset's news headlines are used for validation during every training phase. The dataset used in this study is described in detail in Table I, along with the Maximum (Max), Minimum (Min), and Average (Avg) word counts for news headlines for the train, test, and overall datasets. When sampled using stratified methods, all of the datasets show the same label distributions, with 58 percent of non-clickbait labels and 42 percent of clickbait labels. There is no clear difference in the length of headlines that are not clickbait compared to headlines that are clickbait across all datasets.

TABLE I
THE DETAILED DESCRIPTION OF DATASET

Dataset	Class	Size	Max	Min	Avg
Train	Clickbait	5032	18	2	10.34
	Non-Clickbait	6968	19	2	9.21
	Both	12000	19	2	9.68
	Clickbait	1258	18	2	10.41

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686

Dataset	Class	Size	Max	Min	Avg
Test	Non-Clickbait	1742	18	3	9.19
	Both	3000	18	2	9.70
Overall	Clickbait	6290	18	2	10.35
	Non-Clickbait	8710	19	2	9.20
	Both	15000	19	2	9.69

B. Fine-Tuning BERT

BERT has shown cutting-edge performance on a variety of NLU tasks. Moreover, BERT also has advantages in terms of fewer data needed, quicker development, and better outcomes [13]. A collection of pre-trained models in Indonesian (IndoBERT) were introduced by B. Willie et al. [14] for use in training, evaluating, and benchmarking IndoNLU tasks, including text classification. IndoBERT is a cutting-edge BERT-based language model for Indonesian. The IndoBERT pre-trained model is trained with a masked language modelling objective and a next sentence prediction objective.

In this experiment, we use IndoBERT$_{\text{BASE}}$ and IndoBERT$_{\text{LARGE}}$ models, which are trained in two phases on TPUv3-8 with a maximum sequence length of 128 and 512, respectively. Both IndoBERT models use a 30.552-word vocabulary. In both training phases, IndoBERT$_{\text{BASE}}$ uses a learning rate of 2e-5 and a batch size of 256. However, to maintain the stability of the training, IndoBERTLARGE adjusts the learning rate to 1e-4. Subsequently, the batch size is decreased to 128, and the learning rate is adjusted to 8e-5 during the second training phase. This is done because of the limitations imposed by the memory.. The maximum prediction for each sequence is constrained to 20 tokens for both models during training using the masked language modeling loss.

The following are the steps involved in fine-tuning BERT. Before BERT can be trained on the data, it must first be converted into a particular input format. The news headline's text is changed into lowercase characters and tokenized with the uncased BERT tokenizer. Then, special tokens [CLS] and [SEP] are added at the beginning and end, respectively. Additionally, the classification tasks need to start with the [CLS] token added. The created tokens are then mapped to their respective tokenizer vocabulary indexes. All headline news is either padded or truncated to a single, fixed length, depending on the maximum sequence length. Lastly, attention masks are generated to differentiate between original and padded tokens. These stages are illustrated below with a sequence length of 32 as follows.

- Input Text: *'Pakar Ungkap Analisis Mengejutkan Soal Ekspresi Bebby Fey Saat Ibunda Serangan Jantung'*
- Lowercase: *'pakar ungkap analisis mengejutkan soal ekspresi bebby fey saat ibunda serangan jantung'*
- Tokenized: *['pakar', 'ungkap', 'analisis', 'mengejutkan', 'soal', 'ekspresi', 'bebby', 'fey', 'saat, 'ibunda', 'serangan, 'jantung']*
- Special Tokens: ['[CLS]', *'pakar', 'ungkap', 'analisis', 'mengejutkan', 'soal', 'ekspresi', 'bebby', 'fey', 'saat, 'ibunda', 'serangan, 'jantung',* '[SEP]'*]*
- Truncated and Padded Tokens to a Single Fixed Length: ['[CLS]', *'pakar', 'ungkap', 'analisis', 'mengejutkan', 'soal', 'ekspresi', 'bebby', 'fey', 'saat, 'ibunda', 'serangan,* 'jantung', '[SEP]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]', '[PAD]']
- Input IDs: [2, 6777, 5396, 3169, 10244, 1495, 9502, 334, 5038, 6517, 30371, 305, 23685, 3685, 2937, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
- Attention Masks: [1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

The BERT classifier is trained using the training dataset's input ids, attention masks, and labels concatenated into TensorDataset. Following that, the samples in TensorDataset are randomly divided for train validation in an 80:20 percentage. The train-validation split uses 9600 samples for training and the remaining samples for validation. The IndoBERT$_{\text{BASE}}$ model has 12 transformer layers, 12 attention heads, 768 embedding sizes, and 768 hidden sizes. Furthermore, the IndoBERT$_{\text{LARGE}}$ model has 24 transformer layers, 16 attention heads, 1024 embedding sizes, and 1024 hidden layers.

IndoBERT is fine-tuned for the classification task with Tesla P100 GPU using Google Colab Pro in this study. This study uses a single layer of a simple BERT model called "BertForSequenceClassification" for classification. A 32 sequence lengths were used to develop IndoBERT classifiers for training and validation. We fine-tune an IndoBERT pre-trained model with a learning rate of 5e-5 with a linear scheduler type, and the number of steps used for a linear warmup is 100. This study uses a batch size of 32 per device for training and a batch size of 16 per device for validation. We use steps as the evaluation strategy by setting the logging steps and saving steps to 200 by calculating validation loss and accuracy, meaning that we will perform evaluation and save the model weights on each 200 training step. According to 9600 samples used and a batch size of 32 for training, thus the total optimization steps for each epoch is 300. The number of training epochs used is 30. The research use "EarlyStoppingCallback" with early stopping patience 30 to choose the best model. Thus, the training process will be destroyed when the validation accuracy score worsens for 30 evaluation calls. This study fine-tuned four different classifiers based on several IndoBERT models.

C. Word-Vectors Approaches

The performances of fine-tuned BERT models cannot be assessed without comparison to non-BERT classifiers. This study uses five machine learning classifiers, including the NB, KNN, SVM, DT, and RF trained for the classification of non-clickbait and clickbait news headlines. Several preprocessing steps, including removing punctuations and special characters, changing upper case into lower case words, tokenization, and filtering stop words, and stems, are carried out to create unigrams. The bag-of-words (BoW) and term frequency-inverse document frequency (TF-IDF) models are then used to create word vectors, which are further used to train various

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686

machine learning classifiers. In this study, we utilized grid search cross-validation with a k-fold value of 5 to maximize. The best hyperparameters over a parameter grid of several classifiers. As a result, we obtained the best model for each classifier. The optimized hyperparameters using grid search for training classifiers are shown in Table II.

TABLE II
OPTIMIZED HYPERPARAMETERS FOR TRAINING MACHINE LEARNING CLASSIFIERS

Classifier	Hyperparameter	Value
NB (type = multinomialNB)	alpha	[0, 0.1, 0.01, 0.001, 1.0]
	fit_prior	[True, False]
K-Nearest Neighbor (KNN)	n_neighbors	1 – 20
	P	[1, 2]
	weights	['uniform', 'distance']
SVM (type = SVC)	C	[1, 10, 100, 1000]
	gamma	[0.1, 0.01, 0.001, 0.0001]
	kernel	['linear', 'poly', 'rbf', 'sigmoid']
DT	criterion	['gini', 'entropy', 'log_loss']
	max_depth	[5, 10, 15, 20, 25, 30]
	max_features	['auto', 'sqrt', 'log2', None]
	min_samples_leaf	[1, 5, 10, 15, 20]
	min_samples_split	[1, 2, 3, 4, 5]
	splitter	['best', 'random']
RF	n_estimators	[10, 50, 100, 200]
	criterion	['gini', 'entropy', 'log_loss']
	max_depth	[5, 10, 15, 20, 25, 30]
	max_features	['sqrt', 'log2', None]
	min_samples_leaf	[1, 5, 10, 15, 20]
	min_samples_split	[1, 2, 3, 4, 5]

D. Performance Evaluation

Predicting news headlines is a binary classification task with clickbait and non-clickbait classes. The IndoBERT models and word-vectors-based classifiers are evaluated using 3000 news headlines as the testing dataset. Convert the testing dataset into the format required by IndoBERT, and the same input formatting techniques are applied. The sequence lengths for evaluating the IndoBERT model are like those used to fine-tune the corresponding IndoBERT model.

Furthermore, several preprocessing steps are also performed for testing some machine learning classifiers. Then, using the BoW and TF-IDF models, word vectors are produced by mapping unigrams as features used in training and testing classifiers. The grid search cross-validation was used to optimize the best hyperparameters of different classifiers. Then, the best model of each classifier is used to evaluate the testing dataset. The classifier's performance was evaluated in this work using various evaluation metrics. Finally, the overall performance of the fine-tuned IndoBERT model in classifying clickbait and non-clickbait news headlines is evaluated by comparing it to several machine learning classifiers.

III. RESULT AND DISCUSSION

The section provides and discusses the evaluation results of two fine-tuned IndoBERT pre-trained models [14], i.e., IndoBERTBASE and IndoBERTLARGE are trained using one and two training phases. Table III summarizes the results of gathering the best model of fine-tuned IndoBERT classification models in the training and validation process. For each best model, the results include the global steps, epochs, training and validation loss values, validation accuracy, training, and validation durations.

TABLE III
THE RESULTS OF TRAINING AND VALIDATION FOR GETTING THE BEST MODEL OF FINE-TUNED INDOBERT CLASSIFIER

Model	Steps	Train Loss	Val Loss	Val Acc	Train Time	Val Time
IndoBERT$_{BASE}$ + phase one	6200	0.079	0.417	0.813	1083.12	2.943
IndoBERT$_{BASE}$ + phase two	6200	0.075	0.404	0.818	1683.77	4.701
IndoBERT$_{LARGE}$ + phase one	6200	0.062	0.431	0.803	3461.66	9.395
IndoBERT$_{LARGE}$ + phase two	6200	0.063	0.392	0.825	3568.51	9.355

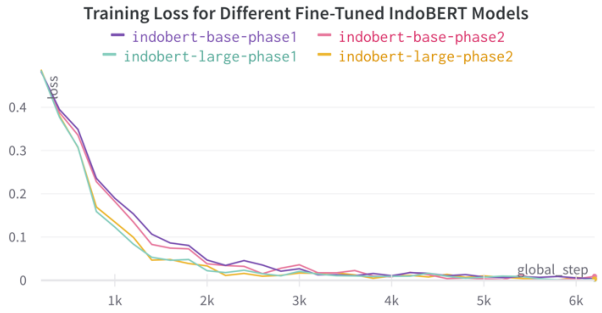

Figure 2. Training Loss for Different Fine-Tuned IndoBERT Models

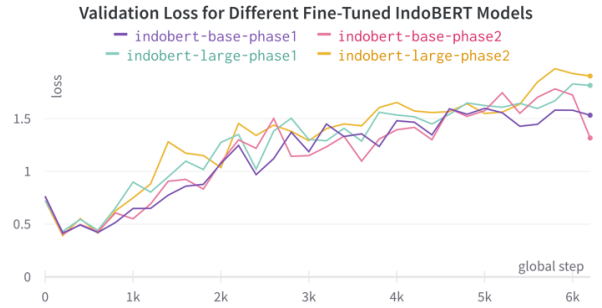

Figure 3. Validation Loss for Different Fine-Tuned IndoBERT Models.

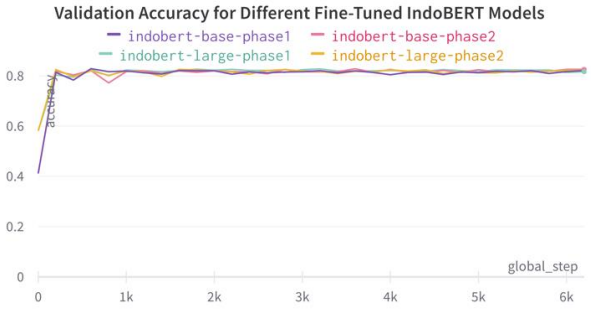

Figure 4. Validation Accuracy for Different Fine-Tuned Indobert Models

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686

The validation accuracy for different fine-tuned IndoBERT models is shown in Figure 4. The results show that the highest validation accuracy for each model is gathered when the global step's value is 200. The best fine-tuned IndoBERT models are used to evaluate the testing dataset. As this study uses "EarlyStoppingCallback" with early stopping patience 30, the training process stops when the validation accuracy worsens for the next 30 evaluation calls.

Additionally, the results for word-vectors-based machine learning classifiers are compared to assess and summarize how well various approaches in this research. The best hyperparameters of classifiers gathered from grid search cross-validation are presented in Table IV. The testing dataset is evaluated using the best model from each classifier.

TABLE IV
BEST HYPERPARAMETERS OF MACHINE LEARNING CLASSIFIERS

Classifier	Hyperparameter and its Value
BoW – NB	{'alpha': 1.0, 'fit_prior': True}
TFIDF – NB	{'alpha': 1.0, 'fit_prior': True}
BoW – KNN	{'n_neighbors': 16, 'p': 1, 'weights': 'uniform'}
TFIDF – KNN	{'n_neighbors': 18, 'p': 2, 'weights': 'distance'}
BoW – SVM	{'C': 1, 'gamma': 0.1, 'kernel': 'rbf'}
TFIDF – SVM	{'C': 1, 'gamma': 0.1, 'kernel': 'linear'}
BoW – DT	{'criterion': 'entropy', 'max_depth': 30, 'max_features': None, 'min_samples_leaf': 15, 'min_samples_split': 2, 'splitter': 'random'}
TFIDF – DT	{'criterion': gini, 'max_depth': 30, 'max_features': None, 'min_samples_leaf': 15, 'min_samples_split': 2, 'splitter': 'random'}
BoW – RF	{'criterion': 'entropy', 'max_depth':30, 'max_features': None, 'min_samples_leaf': 15, 'min_samples_split': 2, 'n_estimators': 10}
TFIDF – RF	{'criterion': 'gini', 'max_depth': 30, 'max_features': None, 'min_samples_leaf': 15, 'min_samples_split': 2, 'n_estimators': 10}

The performance of five machine learning classifiers with different word-vectors-based techniques and four IndoBERT model classifiers is assessed using a testing dataset consisting of 3000 news headlines. Table V summarizes the prediction results for each classifier employed in this experiment, with the abbreviations TP for true positive, TN for true negative, FP for false positive, and FN for false negative. A true positive (TP) results in the model accurately predicting the clickbait class. Similar to a true positive, a true negative (TN) results in the model accurately predicting the non-clickbait class. A false positive (FP) results in the model inaccurately predicting the clickbait class. A false negative (FN) results when the model inaccurately predicts the non-clickbait class.

TABLE V
SUMMARY OF PREDICTION RESULTS USING THE TESTING DATASET

Classifier	TP	FP	TN	FN
BoW – NB	815	316	1426	443
TFIDF – NB	714	217	1525	544
BoW – KNN	763	580	1162	495
TFIDF – KNN	694	256	1486	564
BoW – SVM	726	169	1573	532
TFIDF – SVM	788	248	1494	470
BoW – DT	462	103	1639	796
TFIDF – DT	485	123	1619	773
BoW – RF	469	109	1633	789
TFIDF – RF	460	101	1641	798
$IndoBERT_{BASE}$ + phase one	802	100	1642	456
$IndoBERT_{BASE}$ + phase two	925	193	1549	333
$IndoBERT_{LARGE}$ + phase one	754	66	1676	504
$IndoBERT_{LARGE}$ + phase two	907	180	1562	351

Table VI displays the evaluation performances for all classifiers, including accuracy, precision, recall, and F1-score derived from the Table V results. The performance results for the word-vector-based classifiers show that KNN with bag-of-words has the smallest accuracy (0.6417) and F1-score (0.6352). Compared to KNN, DT and RF classifiers with both BoW and TF-IDF perform somewhat better, with an accuracy score of around 0.70 and an F1-score of around 0.65. Interestingly, KNN with TF-IDF outperformed DT and RF classifiers in terms of accuracy (0.7267) and F1-score (0.7062). Moreover, the NB classifier with BoW and TF-IDF models produces slightly better results than KNN with TF-IDF, with an accuracy of around 0.75 and an F1-score of around 0.73. The highest performance is obtained using the SVM, the highest accuracy (0.7607) is gathered with the BoW model, and the highest F1-score (0.7466) is gathered with the TF-IDF model. According to the word-vector-based results, we can conclude that there are slightly different results when using bag-of-words and TF-IDF, except for the KKN. For the KNN and DT classifiers, the use of the TF-IDF model is slightly better in both accuracy and F1-score than the BoW model. In contrast, for the NB and RF classifiers, the use of the BoW model is slightly better than TF-IDF. Furthermore, the SVM classifier obtained better accuracy with the BoW model, but a better F1-score was obtained with TF-IDF.

According to the IndoBERT classifier results, the $IndoBERT_{LARGE}$ using one training phase model has the smallest accuracy (0.8100) and F1-score (0.7902). In contrast, the $IndoBERT_{BASE}$ using the two training phases model reaches the greatest accuracy (0.8247) and the greatest F1-score (0.8167). Moreover, it is evident that the $IndoBERT_{BASE}$ produces a slightly better result than the $IndoBERT_{LARGE}$ in both accuracy and F1-score. The pre-trained model, which uses two training phases, also produces competitively better results than the pre-trained model, which uses one phase.

Figure 5. Performance Comparison of Word-Vectors-Based And Fine-Tuned BERT Classifiers.

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686

Figure 5 compares the accuracy scores of word-vectors-based machine learning classifiers and fine-tuned IndoBERT classifiers. According to performance comparison results, fine-tuned IndoBERT classifiers outperform word-vectors-based classifiers.

TABLE VI
SUMMARY OF EVALUATION PERFORMANCE OF CLASSIFIERS ON THE TESTING DATASET

Classifier	Accuracy	Precision	Recall	F1-Score
BoW – NB	0.7470	0.7418	0.7332	0.7361
TFIDF – NB	0.7463	0.7520	0.7215	0.7263
BoW – KNN	0.6417	0.6347	0.6368	0.6352
TFIDF – KNN	0.7267	0.7277	0.7024	0.7062
BoW – SVM	0.7663	0.7792	0.7400	0.7461
TFIDF – SVM	0.7607	0.7607	0.7420	0.7466
BoW – DT	0.7003	0.7454	0.6541	0.6458
TFIDF – DT	0.7013	0.7373	0.6575	0.6515
BoW – RF	0.7007	0.7428	0.6551	0.6476
TFIDF – RF	0.7003	0.7464	0.6538	0.6454
$IndoBERT_{BASE}$ + phase one	0.8147	0.8359	0.7901	0.7989
$IndoBERT_{BASE}$ + phase two	0.8247	0.8252	0.8123	0.8167
$IndoBERT_{LARGE}$ + phase one	0.8100	0.8442	0.7807	0.7902
$IndoBERT_{LARGE}$ + phase two	0.8230	0.8255	0.8088	0.8141

Compared to other word-vector-based classifiers, the SVM classifier with the BoW model has the best accuracy (0.7607). The $IndoBERT_{BASE}$ using the two training phases model shows the greatest accuracy of 0.8247, which is 0.064 (6%) greater than the SVM classifier with a BoW model accuracy of 0.7607. In addition, all IndoBERT models outperform all word-vectors-based machine learning classifiers. The better performance of IndoBERT models can be related to using BERT features to fine-tune and assess the classifiers. As contrasted to word-vectors-based features, which perform preprocessing steps by removing most of the words and do not consider the location of words, BERT features retain the context of the words in bidirectional, and no stop words should be removed. Before generating features with a BoW or TF-IDF, word-vector-based classifiers require mandatory textual data preparation to produce better results.

Another issue with word-vectors-based methods is that it generates a large list of features which may result in longer training times, or we need some feature selection techniques to filter the important features. On the other hand, BERT did not require any textual data preprocessing. We simply used an uncased BERT tokenizer to convert contextual data into a particular input format. In contrast to word-vector-based approaches, BERT employs bidirectional models that conjointly condition both word's left and right contexts.

IV. CONCLUSION

The number of clickbait news continues to significantly increase in recent years due to some news publishers using this tactic to engage readers' attention and increase visitor traffic to the news site. According to this problem, a clickbait detector is required to automatically identify news article headlines that include clickbait and non-clickbait. This work aims to address the shortcomings of earlier research on handcrafted features, which limit the solution's generalizability. As a result, BERT has proven to be cutting-edge for a variety of NLU tasks.

In this work, IndoBERT is used as the pre-trained model. We aim to fine-tune the IndoBERT model to predict clickbait of Indonesian news headlines using a dataset of CLICK-ID [9]. The performance of different fine-tuned IndoBERT classifiers is evaluated and compared to different word-vector-based (BoW and TF-IDF) and different machine learning NB, KNN, SVM, DT, and RF classifiers. The evaluation results indicate that all fine-tuned IndoBERT models outperform all word-vectors-based machine learning classifiers in classifying clickbait and non-clickbait Indonesian news headlines.

Furthermore, the highest accuracy of 0.8247 achieves with $IndoBERT_{BASE}$ using the two training, which is 0.064 (6%) greater than the accuracy of the SVM classifier with the BoW model 0.7607. The different approach of BERT causes this compared to word-vector-based. The BERT tokenizer generates contextualized token embeddings. Thus, no handcrafted features are needed. BERT also employs bidirectional representations that conjointly condition both words' left and right contexts to represent the word features.

This study comes with a few limitations. First, this study only used the annotated CLICK-ID dataset. Second, this work just fine-tunes the IndoBERT pre-trained model for identifying clickbait. Future research will fine-tuned and analyze other BERT model variants, including the Lite BERT (ALBERT) [15], the RoBERTa [16], the ELECTRA [17], the ConvBERT [18], the DistilBERT [19], and the AMBERT [20] for the classification of clickbait and non-clickbait Indonesian news headlines. This work advances knowledge by analyzing the effects of fine-tuning the BERT model with various pre-trained model approaches to predicting the clickbait of Indonesian news headlines. Additionally, we can use this IndoBERT-based model to build a clickbait detector to help readers to avoid clickbait news by automatically classifying clickbait and non-clickbait news with sufficient accuracy without depending on any handcrafted features.

ACKNOWLEDGMENT

This research was partially funded by a grant from the Faculty of Mathematics and Natural Sciences at the Universitas Gadjah Mada [grants number: 12/J01.1.28/PL.06.02/2020].

REFERENCE

[1] A.F.Yavi, “Klasifikasi Artikel Berbahasa Indonesia untuk Mendeteksi Clickbait menggunakan Metode Naïve Bayes,” *Journal of Information and Technology (J-INTECH)*, vol. 06, no. 01 pp. 141–147, 2018.

[2] M.Rizky and M.R. Kertanegara, “Penggunaan Clickbait Headline pada Situs Berita dan Gaya Hidup Muslim dream.co.id,” *Mediator Jurnal Komunikasi*, vol. 11, no. 1 pp. 31–43, 2018.

[3] A. Agrawal, "Clickbait Detection using Deep Learning," in *International Conference on Next Generation Computing Technologies*, 2016, pp. 268–272.

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686

[4] B.W. Rauf, S. Raharjo, H. Sismoro, "Deteksi Clickbait dengan Sentence Scoring Based On Frequency di Detik.Com," *Jurnal Teknologi Informasi (JurTI)*, 2020, vol. 4, no.2, pp. 247–252.

[5] R. Sagita, U. Enril, A. Primajaya, "Klasifikasi Berita Clickbait menggunakan K-Nearest Neighbor (KNN)," *Journal of Information System*, 2020, vol. 5, no.2, pp. 230–239.

[6] S. Jumun, L. Lou, N. Wongsap, "Thai Clickbait Headline News Classification and its Characteristics," in *International Conference on Embedded Systems and Intelligent Technology & International Conference on Information and Communication Technology for Embedded Systems (ICESIT-ICICTES)*, 2018.

[7] P.S. Hadi, Muljono, A.Z. Fanani, G.F. Shidik, Purwanto and F. Alzami, "Using Extra Weight in Machine Learning Algorithms for Clickbait Detection of Indonesia Online News Headlines," *2021 International Seminar on Application for Technology of Information and Communication (iSemantic)*, 2021, pp.37-41, doi: 10.1109/iSemantic 52711.2021.9573213.

[8] William and Y. Sari, "CLICK-ID: A Novel Dataset for Indonesian Clickbait Headlines," *Data in Brief*, vol. 32, 2020, doi: 10.106/j.dib.2020.106231.

[9] B.U. Nadia and I.A. Iswanto, "Indonesian Clickbait Detection Using Improved Backpropagation Neural Network," *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 2020, pp.252-256, doi: 10.1109/ISRITI54043.2021. 9702872.

[10] M.A. Shaikh and S. Annappanavar, "A comparative approach for clickbait detection using deep learning," in *2020 IEEE Bombay Section Signature Conference (IBSSC)*, pp. 21-24, 2020, doi:10.1109/IBSSC51096.2020.9332172.

[11] M. Bilal, A.A. Almazroi, "Effectiveness of Fine-tuned BERT Model in Classification of Helpful and Unhelpful Online Customer Reviews", *Electronic Commerce Reseach*, 2022, doi: 10.1007/s10660-022-09560-w.

[12] Gonzalez-Carvajal and E.C. Garrido-Merchan, "Comparing BERT against traditional machine learning, *Preprint arXiv: 2005.13012,* 2020, http://arxiv.org/ abs/2005.13012.

[13] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171-4186, 2019.

[14] B. Wilie, K. Vincentio, G.I. Winata, S. Cahyawijaya, X. Li, Z.Y. Lim, S. Soleman, R. Mahendra. P. Fung, S. Bahar, 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, doi: 10.4850/arXiv.2009.05387.

[15] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," in *International Conference on Learning Representations*, 2020.

[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", *Preprint arXiv:19071.1692*, 2019, http://arxiv.org/abs/19071.1692.

[17] K. Clark, M.T. Luong, Q.V. Le, and C.D. Manning, "ELECTRA: Pre-Training Text Encoders as Discriminators Rather Than Generators," in *The International Conference on Learning Representations (ICLR) 2020*, 2020

[18] Z. Jiang, W. Yu, D. Zhou, Y. Chen, J. Feng, S. Yan, "ConvBERT: Improving BERT with Span-based Dynamic Concolution," *Preprint arXiv: 2008.02496*, 2020, http://arxiv.org/ abs/2008.02496.

[19] V. Sanh, L. Debut, J. Chaumond, T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *Preprint arXiv:1910.01108*, 2020, https://arxiv.org/abs/1910.01108

[20] X. Zhang, P. Li, and H. Li, "AMBERT: A pre-trained language model with multi-grained tokenization. *Preprint arXiv:2008.11869*, 2020, https://arxiv.org/abs/2008.11869

DOI : http://dx.doi.org/10.25139/inform.v7i2.4686