International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. #, No.# (Mon-20..)

Clickbait Detection in Indonesian News Sites Using Deep Learning

Vincent [1], Sharlene Regina [2], and Amalia Zahra [3]

Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta, Indonesia, 11480

E-mail address: [1]vincent053@binus.ac.id, [2]sharlene.regina@binus.ac.id, [3]amalia.zahra@binus.edu

Received ## Mon. 20##, Revised ## Mon. 20##, Accepted ## Mon. 20##, Published ## Mon. 20##

Abstract: Clickbait headlines are disruptive due to their deliberate exploitation of readers' curiosity, creating an information gap between what the reader knows and wants to know. These headlines often focus on sensationalized or misleading information, distorting the reader's perception. Therefore, addressing the issue of clickbait is crucial to maintaining the integrity of news and information dissemination and ensuring that readers are presented with accurate and valuable content that aligns with ethical journalistic standards.In this study, we analyzed clickbait in news headlines using a deep-learning approach. We tested the performance of two deep learning models, FastText+ Bidirectional Long Short-Term Memory (Bi-LSTM) and IndoBERT, to detect whether a news headline is clickbait or not. The results showed that both models can be effectively used as classification methods. Specifically, IndoBERT demonstrated superior accuracy compared to the FastText+Bi-LSTM approach, with an accuracy of 0.79. These findings suggest that IndoBERT is a more accurate and efficient solution for clickbait detection in news headlines. The findings of this study contribute to the ongoing efforts to address the issue of clickbait and its impact on news and information dissemination. By leveraging advanced deep learning techniques, this research provides valuable insights and tools for improving the quality and reliability of online news content, ultimately benefiting both readers and the broader field of journalism.

Keywords: Clickbait, IndoBERT, FastText, Bi-lSTM

1. Introduction

The digital era has changed the way information is presented to the public. The Internet has made information distribution easier, more cost-effective, and faster, with various online information sharing, including social media, blogspot, and news media. According to the Pew Research Center, official news website traffic reached an average of 10 billion unique visitors, with an average view duration of 1.74 minutes between October and December 2022[1]. This shift in information distribution, including journalism, marks a transition towards a more digital and mobile approach.

Clickbait headline news is an intentional strategy that journalists use to increase the number of views and hits on a news article by using sensational headlines. This strategy exploits information gaps to evoke strong emotional responses from readers. Clickbait can identify through several widely used techniques, including ambiguous headlines, exaggeration, inflammatory language, bait and switch, teasing, excessive punctuation, mismatch between headline and news content, and incomplete headlines [2]. Clickbait headlines have raised public concerns, as the credibility of online news sources is often doubted. Readers often need help distinguishing between accurate news articles and clickbait, leading to a decline in public trust in online news. According to the Reuters Institute's 2023 survey, only 40% of respondents trust the news circulating online, marking a 2% decrease from the previous year [3]. Given that digital platforms are the primary source of news dissemination nowadays, clickbait not only stirs journalistic integrity but also poses a significant threat by presenting misleading information in society. It is crucial to develop advanced methods for identifying and filtering out such content. This requires a combination of linguistic analysis and machine learning techniques capable of recognizing the subtle cues that differentiate clickbait from legitimate news headlines. By leveraging natural language processing (NLP) models, researchers can create systems that automatically detect

E-mail: vincent053@binus.ac.id, sharlene.regina@binus.ac.id, amalia.zahra@binus.edu

and flag clickbait, thereby helping readers navigate the vast amount of information available online. This research aims to identify unique characteristics and patterns to construct an accurate clickbait detection model. In addition to mitigating the spread of deceptive information in news delivery, also to enhance public trust and encourage online news producers to uphold journalistic ethics. The study is limited to using Indonesian language datasets, which can restrict the relevance of the discovery. However, this constraint is necessary to ensure the accuracy and precision of the clickbait detection model for the Indonesian language.

Through the analysis of a labeled dataset by employing two distinct models, FastText + Bi-LSTM and IndoBERT, the model discerns distinctive features and patterns inherent in clickbait headlines within the Indonesian language. Bi-LSTM (Bidirectional Long Short-Term Memory) is utilized in both models for its capability to capture sequential dependencies and patterns within textual data. FastText complements the Bi-LSTM by providing robust word embeddings, facilitating the model's understanding of semantic relationships within the text. On the other hand, the IndoBERT uses a pre-trained language representation model tailored for the Indonesian language. This model further enhances the ability to capture intricate linguistic nuances specific to the Indonesian context. By integrating these Pattern Recognition and Representation Learning techniques into two distinct models, the research aims to achieve superior performance in identifying clickbait headlines, thereby advancing the field of computational linguistics.

Ultimately, this study contributes to the growing body of knowledge in computational linguistics and natural language processing. The techniques developed here can be adapted and applied to other languages and contexts, fostering a more reliable and trustworthy digital news ecosystem worldwide. As digital media continues to evolve, such research is essential in ensuring that the integrity of information remains intact.

The research paper will be structured as follows. Section 2 will present related previous work on clickbait detection methods from various sources. Section 3 will discuss the proposed method used in this research. The model development results will be presented in graphs and tables in Section 4. The paper will conclude with a summary in Section 5.

2. Related Work

Clickbait headlines have become a prevalent problem in the digital era and contribute to the spread of misinformation. Researchers have developed several algorithms to detect clickbait headlines as the demand for reliable information increases. These algorithms rely on linguistic analysis and incorporate and improve deep learning models on multiple datasets. Much previous research has been used as a reference for conducting detecting clickbait. According to paper [4], By using 6,632 training data that has been labeled as much as 3,316 data clickbait and 3,316 data non-clickbait, it is found that using Multilingual Bidirectional Encoder Representation (M-BERT) has a higher accuracy of 91% than using bidirectional long short-term memory (Bi-LSTM), Convolutional Neural Network (CNN) Extreme Gradient Boosting (XGBoost).

According to paper [4], two datasets were utilized for evaluating text similarity measures. The first dataset originates from the SICK dataset, comprises 5,000 English sentence pairs annotated with similarity scores. The second dataset, Lee dataset, consists of 50 short English documents presenting news from the Australian Broadcasting Corporations news mail service. From the study, Word2Vec, NASARI+ Word2Vec, FastText outperform the traditional TF-IDF model. FastText with a score of 0.6, emerged as the top-performing model based on average Spearman correlation values. This indicates that the models are effective in capturing the underlying similarity structure of the text data, as perceived by human annotators.

In the paper [5], more significant data is used from the public dataset from Kaggle, which is 32,000 data consisting of 16,000 data labeled clickbait and 16,000 data labeled non-clickbait. The data division consists of 80% training data and 20% testing data, which have been balanced. Furthermore, the data that has been balanced will be carried out in several hyper-parameter settings, which are also investigated to find the most suitable parameter model for the clickbait classifier. So, through this experiment, a comprehensive evaluation is obtained where the BERT algorithm can be used to detect clickbait titles with the highest accuracy result of 98.86% with a computation time of 0.9 hours compared to machine learning algorithms such as Naive Bayes (NB), Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), Support Vector Machines (SVM). BERT transformer architecture allows for efficient processing of large volumes of text data. Despite its sophisticated architecture and large parameter space, it exhibits impressive computational efficiency. These findings underscore the importance of word embedding, both in terms of improving model understanding of text and enhancing computational efficiency.

Moreover, the paper [6] used 3,465 fake news and 766 real news from turnbackhoax.id that has been resampled with a ratio of 1:1. The results of using the Indonesia Bidirectional Encoder Representations from Transformers (IndoBERT) show unsurprisingly outperforms with a precision of 94.66%, recall 94.5%, F1-score 84.6%, and Accuracy 94.66%. However, this research found problems of data overfitting due to small data trained with huge model parameters. Consequently, the model failed to achieve an optimal balance between complexity and generalization, resulting in suboptimal performance. Therefore, with previous studies, we can make it more efficient by selecting and combining the best ideas from related works.

3. Methodology

In this section, we present the methodology used in this research. Text classification techniques are illustrated in Figure 1. These experiments started with data collection, augmentation, preprocessing, split data, hyperparameter tuning and evaluating each model.

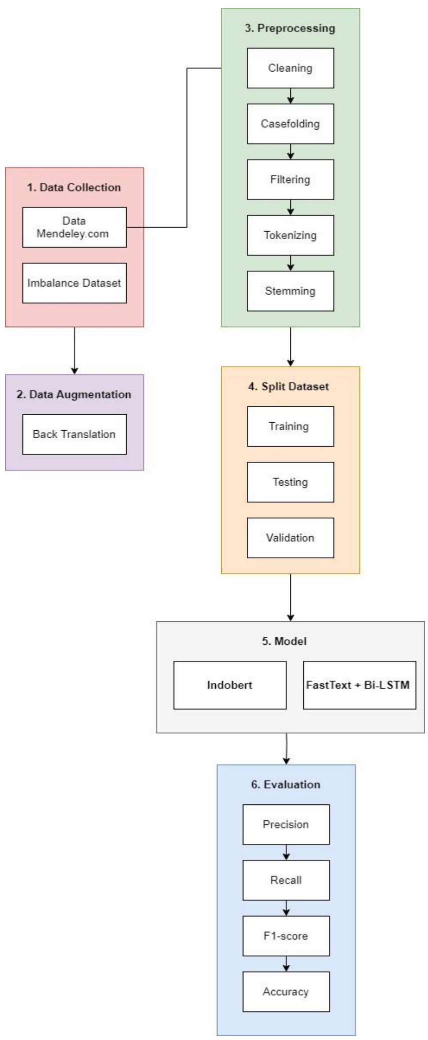

Figure 1. Proposed Method

A. Data Collection

This research uses a public dataset from datamendeley.com, which contains 15,000 samples of news headlines labeled clickbait and not clickbait. The news sources taken are from 12 Indonesian online news sites, namely Tempo, Kompas, Republika, Tribunnews, Wowkeren, Sindonews, Liputan6, Detik News, Okezone, Fimela, Kapanlagi, and Posmetro-Medan. The headlines collected will be labeled score "1" if clickbait, and they will be scored "0" If the headline is non-clickbait.

B. Data Augmentation

The dataset used consists of 8,710 non-clickbait and 6,290 clickbait. Figure 2 depicts the ratio of non-clickbait and clickbait classes before and after data augmentation.

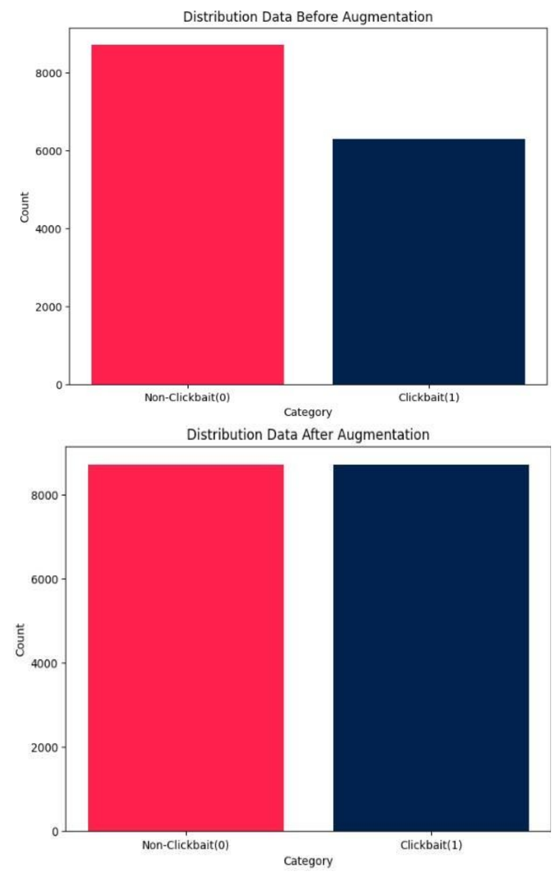

Figure 2. Data Distribution

In handling imbalance, the data augmentation technique uses the back translation method. Data augmentation is commonly used in data processing to generate additional variations in existing datasets [7]. In this context, back translation, as shown in figure 3, is done by translating text from Indonesian to English and back to Indonesian using the Marian MT from the Transformers library. The resulting text might have slight differences from the original but will retain the same overall meaning. This method can help generate synonym-rich text, providing the model with diverse linguistic expressions.

Data augmentation is essential in improving the performance of deep learning classification models, especially in tasks such as text classification [8]. This technique can help the model understand a wider variety of data, thus improving accuracy and reducing the risk of overfitting.

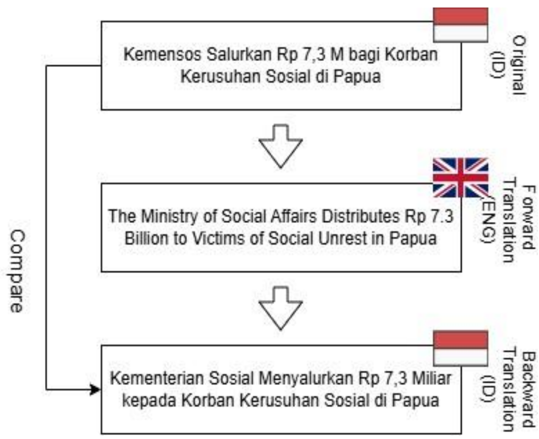

Figure 3. Back Translation

TABLE I. DATA AFTER BACK TRANSLATION

Title	After Back Translation
Kemensos Salurkan Rp 7,3 M bagi Korban Kerusuhan Sosial di Papua	Kementrian Penghormatan Rp 7,3 M bagi Korban Kerusuhan Sosial di Papua
Terkait Mayat Bayi Mengenaskan di Tangerang, Seorang Pria Ditangkap Polisi	Terkait Jenazah Bayi yang Mengerikan di Tangerang, Seorang Pria Ditangkap oleh Polisi
Viral! Driver Ojol di Bekasi Antar Pesanan Makanan Pakai Sepeda	Viral! Pengemudi Ojek Online di Bekasi Mengantarkan Pesanan Makanan Menggunakan Sepeda
Tidur Siang 1-2 Kali Seminggu Kurangi Risiko Serangan Jantung	Tidur Siang 1-2 Kali Seminggu Mengurangi Risiko Serangan Jantung
Pria Misterius Berjubah Putih Viral di Sumut, Ingatkan 'Tuhan Murka'	Pria Misterius Berjubah Putih Viral di Sumatera Utara, Ingatkan 'Tuhan Murka'

Based on Table I, 2,420 clickbait or label score '1' will be added to the dataset. By presenting synthetic variations in the dataset, the model can learn from different contexts of data and make it more adaptable. Therefore, data augmentation becomes an essential strategy in improving dataset quality and directly affects the performance of deep learning models on clickbait and non-clickbait classification tasks.

C. Preprocessing Text

Preprocessing text involves a series of steps to manipulate and clean text for more effective natural language processing[9]. The preprocessing text includes several steps to clean and transform the raw text into a suitable form for analysis or modeling [10]:

- Cleaning: removing special characters, numbers, whitespace, and symbols that are not necessary from the text. The primary goal is to clean the text of irrelevant elements or noise.
- Case Folding: transforms all letters in the text to either lowercase or uppercase. This step helps overcome capitalization differences that may impact model performance.
- Filtering: removing irrelevant words or stop words from the text. Stop words are common words that often do not contribute significantly to analysis
- Stemming: removing word suffixes to return the word to its base form. Specifically for the Indonesian language, the Sastrawi library can be used for stemming to optimize this process.
- Tokenizing: breaks the text into smaller units such as words or phrases.

TABLE II. DATA AFTER PREPROCESSING

Title	After Preprocessing
Kementrian Penghormatan Rp 7,3 M bagi Korban Kerusuhan Sosial di Papua	[[mentri], [hormat], [salur],[rp], [m],[bagi],[korban], [rusuh],[sosial],[papua]]
Terkait Jenazah Bayi yang Mengerikan di Tangerang, Seorang Pria Ditangkap oleh Polisi	[[kait], [jenasah],[bayi],[enas],[tangerang],[orang],[pria],[tangkap],[polisi]]
Viral! Pengemudi Ojek Online di Bekasi Mengantarkan Pesanan Makanan Menggunakan Sepeda	[[viral],[driver],[ojol],[bekas],[antar],[pesan],[makan],[pakai],[sepeda]]
Tidur Siang 1-2 Kali Seminggu Mengurangi Risiko Serangan Jantung	[[tidur],[siang],[kali],[minggu],[kurang],[risiko],[serang],[jantung]]
Pria Misterius Berjubah Putih Viral di Sumatera Utara, Ingatkan 'Tuhan Murka'	[[pria],[misterius],[jubah],[putih],[viral],[sumatera],[utara],[ingat],[tuhan],[murka]]

Based on Table II, preprocessing text has a vital role in improving the quality and consistency of text, especially in text classification tasks. Preprocessing steps help the model to learn more effectively.

D. Data Splitting

A crucial step in developing deep learning models is splitting the dataset into distinct subsets to facilitate practical training, testing, and validation [10].

Specifically, the dataset was partitioned as follows: 80% for training, 13.33% for testing, and 6.67% for validation.

TABLE III. DATA SPLITTING

Class	Splitting By Class			
	Training	*Test*	*Validation*	*Total*
Non-Clickbait (0)	6,968	1,161	581	8,710
Clickbait (1)	6,968	1,161	581	8,710

Based on Table III, by employing stratified sampling, the distribution of clickbait and non-clickbait headlines is balanced across the training, testing, and validation datasets, preserving the representativeness of the data. This balanced distribution is essential for the model to learn and generalize effectively across the different classes.

E. FastText and Bidirectional LSTM

1) FastText.

Word embedding is a technique for creating vector representations of words. Converting text data into numerical vectors makes data processing more manageable and more efficient to analyze [1]. FastText is a commonly used word embedding method in text-based natural language processing. By incorporating subword information, FastText can generate richer word representations, particularly for languages with complex morphology or many rare words [12].

2) Bidirectional LSTM

Word Bidirectional LSTM is an extension of LSTM that processes input data sequences in two directions: forward and backward [13]. This approach allows the model to understand better a word's contextual information based on the words that come before and after it, making it suitable for understanding the meaning of words and sentence paraphrases. Bi-LSTM achieves this by using two separate hidden layers for the forward and backward states and combining the output of these two hidden layers. This combined output can improve the model's ability to make more informed predictions compared to LSTM. As a result, Bi-LSTM is a popular choice for various NLP applications where understanding the sequential nature of the input data is critical for achieving high performance. The Bi-LSTM network consists of several layers. These layers include [13]:

- Embedding Layer: The embedding layer converts the input data into lower-dimensional vectors called embeddings. These embeddings capture the semantic meaning of the input data text. The embedding layer allows the network to process the input data efficiently.
- Dropout Layer: The dropout layer is a regularization technique to prevent network overfitting. It works by randomly setting a fraction of input units to 0 at each update during training, which prevents the network from overreliance on some specific input features.
- Bidirectional Layer: The bidirectional layer captures contextual information from past and future states achieved by processing the input sequence in both the forward and backward directions.
- Global Max Pooling Layer: The global max pooling layer often reduces the output dimensionality by extracting the most critical features from the sequence data. It works by taking the maximum value over the time dimension; regardless of the input sequence length, it will result in fixed-size output.
- Dense Layer: the dense or fully connected layer produces the network's final output. Its processes produce the final output based on the network's learning from input data processing.

For this research, we used hyperparameter tuning using the library from Keras Turner with 30 trials. Hyperparameter tuning is required to determine the ideal collection of hyperparameters for optimal model performance. Here are the 6 best trials in hyperparamter tuning.

TABLE IV. HYPERPARAMETER TUNING TESTING

No	Hyperparameter			Validation Accuracy
	Learning Rate	*Dropout Rate*	*Units*	
1	**0.01**	**0.4**	**16**	**0.788**
2	0.01	0.5	128	0.782
3	0.01	0.3	32	0.781
4	0.01	0.3	16	0.778
5	0.01	0.4	64	0.777
6	0.01	0.5	64	0.777

Based on Table IV, the tuned hyperparameters include the dropout rate, the number of units in the Bidirectional LSTM, and the learning rate for the Adam optimizer. The results of the hyperparameter search are then summarized. Based on the hyperparameter tuning results and the provided resources, the following analysis can made as follow:

- Dropout Rate: The dropout rate is a hyperparameter that randomly controls the rate to set input units to 0. In the context of the hyperparameter tuning results, the dropout rate varied between 0.3, 0.4, and 0.5. The best validation accuracy was achieved with a dropout rate of 0.4.
- Units: Units refer to the number of hidden units in the Bi-LSTM layer. The hyperparameter tuning results varied the number of units between 16, 32, 64, and 128. The best validation accuracy was achieved with 16 units, which suggests that a smaller number of units was more effective in this case.
- Learning Rate: The learning rate is a hyperparameter that defines the step size utilized in optimizing neural network training. In the context of the hyperparameter tuning results, the learning rate varied between 0.001, 0.01, and 0.1. The learning rate of 0.01 is optimal for all trials in the hyperparameter tuning results.

The following results show that the best hyperparameters aligned with the dataset's criteria resulted in the best validation accuracy of 78%. This result was achieved with a learning rate of 0.01, a dropout rate of 0.4, and 16 units. The proposed model architecture is summarized in Table V.

TABLE V. MODEL ARCHITECTURE SUMMARY

Layer (type)	***Output Shape***	***Parameters***
FastText (Embedding)	(None, 17, 300)	4,500,000
Dropout (Dropout)	(None, 17, 300)	0
Bidirectional (Bidirectional)	(None, 17, 32)	40,576
Dropout_1 (Dropout)	(None, 17, 32)	0
GlobalMaxPooling1D (GlobalMaxPooling1D)	(None, 32)	0
Dense (Dense)	(None, 1)	33
Total Parameters: 4,540,609 (17.32 MB)		
Trainable Parameters: 40,609 (158.63 KB)		
Non-trainable Parameters: 4,500,000 (17.17 MB)		

Based on Table V, the following provides description of each layer within the model architecture:

- Embedding Layer: Each word in the input text is mapped to a 300-dimensional vector. The embedding matrix has a size of 4,500,000 parameters, reflecting the vocabulary size and the embedding dimension.
- Dropout Layer: This layer does not add any parameters to the model, also it does not introduce additional weights or biases that need to be trained.
- Bidirectional LSTM Laye: The LSTM has 16 units in each direction, leading to an output dimension of 32 (17, 32). The total number of parameters for this layer is 40,576, which includes the weights and biases for both the forward and backward LSTMs.
- GlobalMaxPooling1D Layer: This layer reduces the output dimension to (None, 32), summarizing the most important features from the sequence.
- Dense Layer: The dense layer adds 33 parameters to the model, consisting of weights and a bias term.

In this research, we use sigmoid activation and binary cross entropy is an appropriate choice because the data consists of two classes, namely clickbait and non-clickbait. The sigmoid activation function was chosen because it can convert inputs into outputs in the range of 0 to 1, which is very suitable for binary classification. With sigmoid, the model can generate the probability of whether a headline belongs to the clickbait or non-clickbait class.

Meanwhile, binary cross entropy is used as a loss function because it effectively measures the error between the probability predicted by the model and the actual binary class label. The use of binary cross entropy allows the model to learn faster and more accurately in distinguishing between clickbait and non-clickbait, as it provides a larger penalty for significant prediction errors. This combination ensures that the model can capture the complexity of the data well and provide reliable classification results.

As for, we also use the Adam optimizer for the model training process. The use of Adam optimizer was chosen due to its proven ability to adaptively adjust the learning rate for each model parameter. it is important to ensure that the learning process runs efficiently, and the model can overcome the challenge of learning a good representation of both classes.

F. IndoBERT

Bidirectional Encoder Representation from Transformers is a pre-trained language model that provides the INDOLEM dataset to address the shared challenges faced in Indonesian natural language processing research [14]. Although Indonesian is the world's tenth-largest spoken language, researchers working with the language often need more annotated datasets, less sparse language resources, and more standardized resources. IndoBERT is a pre-trained model based on the BERT structure and trained using two learning approaches: Masked Language Modelling (MLM) and Next Sentence Prediction (NSP). MLM involves filling in the blanks by using the context of the words around the masked token to predict the word, enabling the model to figure out the meaning of phrases in an instance. On the other hand, NSP predicts the following sentence given two provided models, enabling the model to fully understand the connection between the two phrases. By leveraging both approaches, the IndoBERT model can understand the meaning and relationships among sub-words, enabling it to recognize languages with high morphological complexity [15]. The model architecture is shown in figure 4.

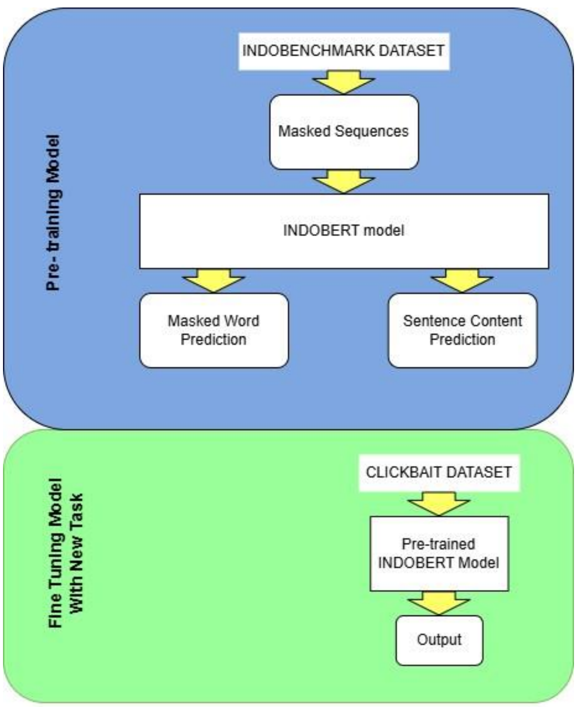

Figure 4. IndoBERT

G. *Evaluation Metrics*

For evaluating text classification models, such as those for detecting clickbait headlines, several key metrics are commonly used. These metrics help in understanding the performance and effectiveness of the model in various aspects:

- Accuracy

  Accuracy represents the precise proportion of correct predictions compared to the entire dataset. Equation (1) shown below illustrates the calculation for accuracy of the model.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

- Precision

  Precision measures the proportion of actual positive labels that correctly identified among all the labels that predicted as positive. Equation (2) shown below illustrates the calculation for precision of the model.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

- Recall

  Recall measures the proportion of actual positive labels that correctly identified among all the true positive labels in the dataset. Equation (3) shown below illustrates the calculation for recall of the model.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

- F1 Score

  The F1 Score calculating the weighted mean of precision and recall. Equation (4) shown below illustrates the calculation for F1 Score of the model.

$$\text{F1-score} = 2 * \frac{(Recall*Precision)}{(Recall+Precision)} \quad (4)$$

Where TP is True Positives, TN is True Negatives FP is False Positives and FN is False Negatives.

4. Result and Discussion

In this section, we compared the performance of models using Curve and Metric Evaluation.

A. *Training and Validation Curve*

Training and validation curve graphs are visual tools used in deep learning to monitor model performance during the training process. It is commonly used to assess how effectively a model learns from training data and how effectively it can generalize patterns discovered in training data to previously unidentified data (validation or test data).

One common issue in deep learning modeling is overfitting, which occurs when the model performs exceptionally well on the training data but poorly on the validation data. This happens when the model learns not only the underlying patterns but also the noise and details specific to the training set. In training and validation curves, overfitting is typically indicated when the training accuracy continues to increase while the validation accuracy plateaus or decreases. Similarly, the training loss keeps decreasing while the validation loss starts to increase after a certain number of epochs to address overfitting.

Several techniques in reducing overfitting have been carried out in this study, namely using l2 regularization, dropout, early stopping and data augmentation. From the use of these techniques, based on Figures 5 and 6 we can see that the IndoBERT model has a larger overfitting distance than FastText+Bi-LSTM. However, the IndoBERT model has the advantage of a faster computational process in learning data compared to FastText+Bi-LSTM. This suggests that while IndoBERT may require more careful regularization to prevent overfitting, its efficiency in processing and learning from data could make it more suitable for applications where computational speed is critical. Meanwhile, FastText+Bi-LSTM, with its lower overfitting distance, may be preferable where model generalization is more crucial

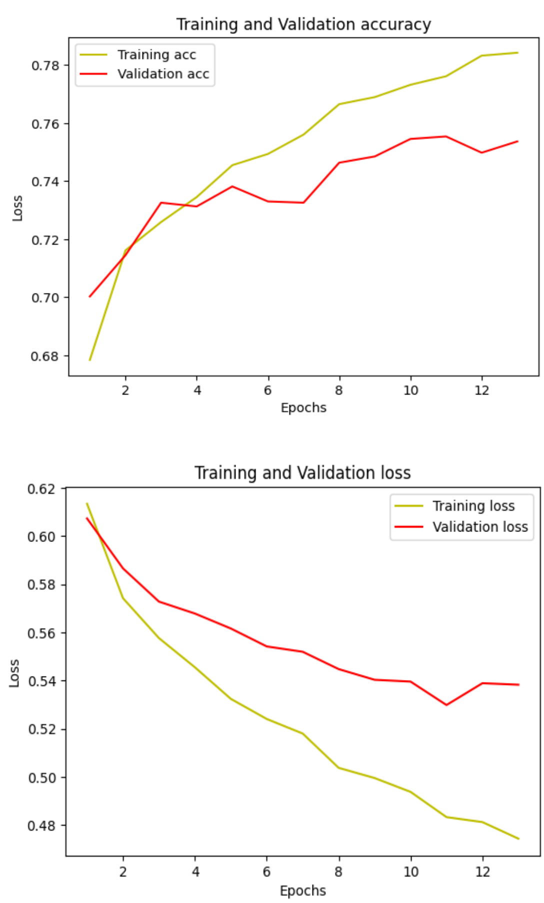

Figure 5. Training and validation Curve Using FastText +Bi-LSTM

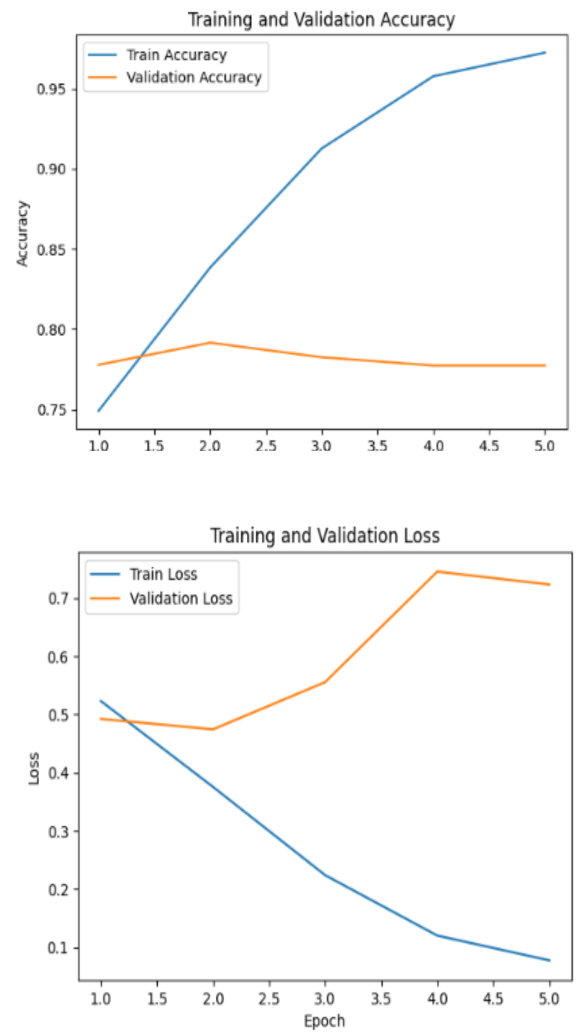

Figure 6. Training and validation Curve Using Indobenchmark/IndoBERT-base-p1.

B. *Confusion Matrix*

The confusion matrix presents an in-depth evaluation of true positives, true negatives, false positives, and false negatives. The elements of the matrix are the counts of how many samples are classified into each class. The four different quadrants represent True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). This matrix is extremely useful for understanding the performance of a classification model beyond simple accuracy. It allows for the calculation of important metrics such as precision, recall, F1-score, and accuracy, which provide a more detailed assessment of a model's effectiveness.

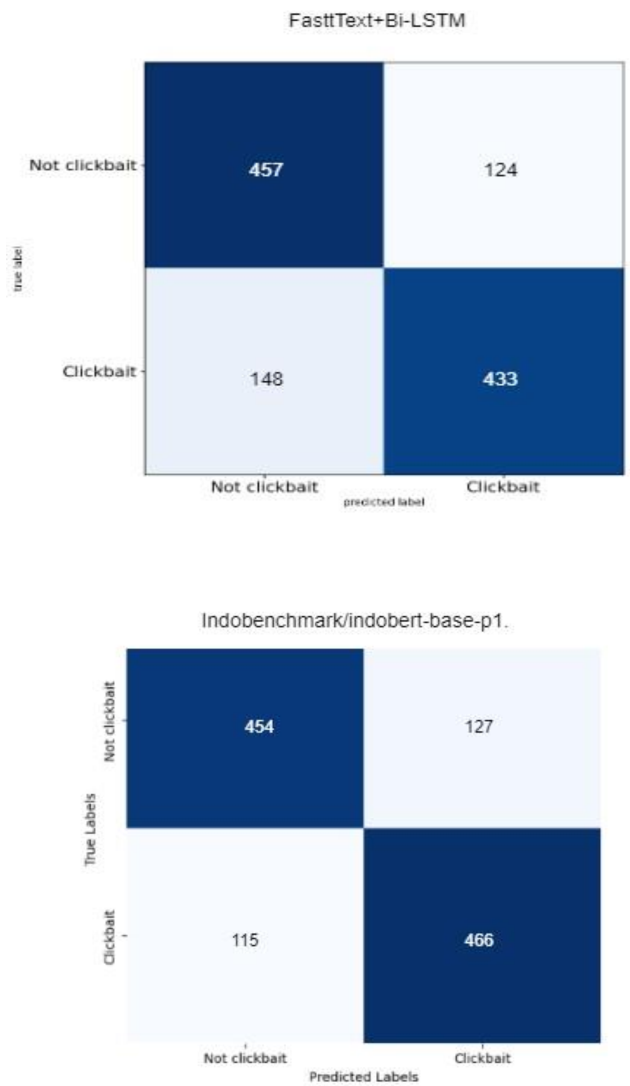

Figure 7. Confusion Matrix

Based on the confusion matrices provided in Figure 7, the matrix for FastText + Bi-LSTM indicates 457 instances classified as True Negatives (TN), 124 as False Positives (FP), 148 as False Negatives (FN), and 433True Positives (TP). Conversely, the matrix for Indobenchmark/IndoBERT-base-p1 denotes 454 instances as True Negatives (TN), 127 as False Positives (FP), 115 as False Negatives (FN), and 466 as True Positives (TP). The superiority of FastText + Bi-LSTM in accurately classifying negative samples is evident from the lower count of False Positives. At the same time, Indobenchmark/IndoBERT-base-p1 performs better in correctly classifying positive samples, as indicated by the lower count of False Negatives.

TABLE VI. COMPARISON BETWEEN FASTTEXT AND INDOBERT

Model	**Model Evaluation**			
	Accuracy	***Precision***	***Recall***	***F1-Score***
FastText +Bi-LSTM	0.76	0.77	0.74	0.75
Indobenchmark/IndoBERT-base-p1	0.79	0.78	0.80	0.79

Based on table VI, here are four things that are needed to consider:

- Accuracy: FastText and Bi-LSTM score is 0.76, while IndoBERT score is 0.79. This indicated that IndoBERT has a higher accuracy.
- Precision: FastText and Bi-LSTM score is 0.77, while IndoBERT score is 0.78. This indicated that IndoBERT has a higher precision score.
- Recall: FastText and Bi-LSTM score is 0.74, while IndoBERT score is 0.80. This indicated that IndoBERT has a higher recall score.
- F1 Score: FastText and Bi-LSTM score is 0.75, while IndoBERT score is 0.79. This indicated that IndoBERT has a higher F1 score.

Based on the evaluation results, although both models demonstrate good performance, IndoBERT-base-p1 slightly outperforms FastText + Bi-LSTM regarding accuracy, recall, precision and F1-Score. Therefore, both models can be considered effective choices for the given classification task, with IndoBERT-base-p1 having a slight advantage in several evaluation metrics. To elaborate further, IndoBERT-base-p1 exhibits a slightly higher overall accuracy (0.79), precision (0.78), recall score (0.80), and F1 Score (0.79) compared to FastText + Bi-LSTM.

5. CONCLUSIONS

After analyzing the two deep learning models, Bi-LSTM and IndoBERT, it can be concluded that both can be effectively utilized as classification methods to determine whether a news title is clickbait or not clickbait. IndoBERT demonstrates superior accuracy compared to the FastText+Bi-LSTM approach. Additionally, employing IndoBERT also leads to faster computational times during training, thereby allowing the model to learn better in a shorter period. These findings suggest that IndoBERT presents a more accurate and efficient solution for clickbait detection in news headlines.

REFERENCES

[1] Pew Research Center, "Pew Research Center's Journalism Project," Accessed: Feb. 06, 2024. [Online]. Available: https://www.pewresearch.org/journalism/fact-sheet/digital-news/?tabId=tab-c4c19f15-4b7c-4229-864a-3da864622ddc

[2] A. Pujahari and D. S. Sisodia, "Clickbait detection using multiple categorisation techniques," J. Inf. Sci., vol. 47, no. 1, pp. 118–128, Feb. 2021, doi: 10.1177/0165551519871822.

[3] N. Newman, "Overview and key findings of the 2023 Digital News Report. Reuters Institute for the Study of Journalism," Reuters Institute for the Study of Journalism.

[4] M. N. Fakhruzzaman, S. Z. Jannah, R. A. Ningrum, and I. Fahmiyah, "Clickbait Headline Detection in Indonesian News Sites using Multilingual Bidirectional Encoder Representations from Transformers (M-BERT)," Feb. 2021.

[5] A. Chowanda, N. Nadia, and L. M. M. Kolbe, "Identifying clickbait in online news using deep learning," Bull. Electr. Eng. Inform., vol. 12, no. 3, pp. 1755–1761, Jun. 2023, doi: 10.11591/eei.v12i3.4444.

[6] S. Isasani, G. Nico, and M. Permana, "INDOBERT FOR INDONESIAN FAKE NEWS DETECTION," ICIC Express Lett., vol. 16, no. 3, pp. 289–297, Mar. 2022.

[7] D. R. Beddiar, M. S. Jahan, and M. Oussalah, "Data expansion using back translation and paraphrasing for hate speech detection," Online Soc. Netw. Media, vol. 24, p. 100153, Jul. 2021, doi: 10.1016/j.osnem.2021.100153.

[8] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text Data Augmentation for Deep Learning," J. Big Data, vol. 8, no. 1, 2021, doi: 10.1186/s40537-021-00492-0.

[9] N. Garg and K. Sharma, "Text pre-processing of multilingual for sentiment analysis based on social network data," Int. J. Electr. Comput. Eng., vol. 12, no. 1, p. 776, Feb. 2022, doi: 10.11591/ijece.v12i1.pp776-784.

[10] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text preprocessing for text mining in organizational research: Review and recommendations," Organ. Res. Methods, vol. 25, no. 1, pp. 114–146, 2022, doi: 10.1177/1094428120971683.

[11] S. Selva Birunda and R. Devi, "A review on word embedding techniques for text classification," in Innovative Data Communication Technologies and Application. Proc. ICIDCA 2020, pp. 267–281.

[12] M. Umer et al., "Impact of convolutional neural network and FastText embedding on text classification," Multimed. Tools Appl., vol. 82, no. 4, pp. 5569–5585, Feb. 2023, doi: 10.1007/s11042-022-13459-x.

[13] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "The Performance of LSTM and BiLSTM in Forecasting Time Series," in 2019 IEEE Int. Conf. Big Data (Big Data), IEEE, Dec. 2019, pp. 3285–3292, doi: 10.1109/BigData47090.2019.9005997.

[14] F. Koto, A. Rahimi, J. H. Lau, and T. Baldwin, "IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained Language Model for Indonesian NLP," in Proc. 28th Int. Conf. Comput. Linguistics, Stroudsburg, PA, USA: Int. Comm. Comput. Linguistics, 2020, pp. 757–770, doi: 10.18653/v1/2020.coling-main.66.

[15] D. U. K. Putri and D. N. Pratomo, "Clickbait Detection of Indonesian News Headlines using Fine-Tune Bidirectional Encoder Representations from Transformers (BERT)," Inform: J. Ilm. Bidang Teknol. Inf. Komun., vol. 7, no. 2, pp. 162–168, Jul. 2022, doi: 10.25139/inform.v7i2.4686.

Vincent is a student pursuing a Bachelor's degree in Computer Science at Bina Nusantara University. They have demonstrated passion for Natural Language Processing, which presented in this paper published in International Journal of Computing and Digital Systems. This publication marks their inaugural contribution to the academic community and reflects eagerness to engage with scholarly discourse.

Sharlene Regina is a student pursuing a Bachelor's degree in Computer Science at Bina Nusantara University. They have demonstrated passion for Natural Language Processing, which presented in this paper published in International Journal of Computing and Digital Systems. This publication marks their inaugural contribution to the academic community and reflects eagerness to engage with scholarly discourse.

Amalia Zahra has been a lecturer at the Master of Computer Science department, Bina Nusantara University since 2017. She earned her Ph.D. from the School of Computer Science and Informatics at University College Dublin (UCD), Ireland, in 2014. She completed her Bachelor degree at the Faculty of Computer Science, University of Indonesia (UI), in 2008. She gained significant research experience as a research assistant working on the development of an Indonesian Speech Recognition System after graduating from UI. Before joining Bina Nusantara University, she was a lecturer and researcher at the Faculty of Computer Science, University of Indonesia. Her research interests encompass Speech Processing, Speech Recognition, Speaker Recognition, Spoken Language Identification, and other areas related to Speech Technology.