Fake News Detection on News Websites Using Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT)

Niar Yuniarti[*,a,1], Siti Yuliyanti[a,2], Irani Hoeronis[a,3]
[a]Siliwangi University, Tasikmalaya, Indonesia

*Corresponding Author: yuniartin85@gmail.com

Abstract

The rapid spread of misinformation in digital media has become a serious challenge, particularly in the context of Indonesian online news consumption. Therefore, this study aims to develop an effective fake news detection system by leveraging a contextual language model tailored to the Indonesian language. This study proposes the use of the Indonesian Bidirectional Encoder Representations from Transformers (IndoBERT) model to detect fake news by optimizing the model architecture and hyperparameter settings. IndoBERT, as a contextual language model, functions to capture the nuances of language in text and can be adjusted at the output layer for classification tasks. In this study, the model architecture is enhanced by adding classification layers, including a linear layer, ReLU activation function, dropout, and a sigmoid output layer, which is expected to improve the accuracy of fake news detection. This study uses Indonesian news data from two sites, Turnbackhoax and Detik.com, covering various topics. The model is trained with different hyperparameter settings, including batch size, dropout, and learning rate, to find the best combination that achieves high accuracy in detecting fake news. The trained model is integrated into a Flask-based web application, allowing users to automatically verify the truth of news through a simple and user-friendly interface. The results show that the best combination, with a learning rate of 2e-5, dropout of 0.1, and a batch size of 32, achieves an accuracy of 96.4%. This study demonstrates that IndoBERT, with the appropriate settings, can be effectively applied to detect fake news with high accuracy, contributing to the development of hoax detection applications.

Keywords: IndoBERT, Fake News Detection, Hyperparameter Optimization, Flask Web Application, Indonesian News Dataset

I. INTRODUCTION

The internet has dramatically enhanced real-time access to global information, yet it has also become a breeding ground for the rapid spread of fake news. According to data from the Ministry of Communication and Information Technology, Indonesia is home to around 800,000 websites disseminating false information. Between 2018 and 2023 alone, 11,357 fake news cases were identified. Fake news, defined as misleading or fabricated content presented as factual information, can distort readers' perceptions [1]. Its impacts are profound, ranging from inducing individual anxiety and fear to eroding societal trust and triggering conflicts among communities [2]. This pervasive influence underscores the urgent need for comprehensive research into this issue, particularly in contexts like Indonesia, where digital literacy varies widely.

Efforts to combat fake news in Indonesia largely rely on manual processes led by organizations such as the Indonesian Anti-Slander Society (MAFINDO). These processes, which involve tracking fake news sources and responding to public complaints, are labor-intensive and limited in scale. This manual approach is insufficient to address the sheer volume and speed at which fake news spreads online. Researchers have thus explored automated solutions, with some relying on conventional machine learning algorithms like Support Vector Machines (SVM). However, these algorithms struggle to capture the nuanced and complex linguistic patterns characteristic of fake news, limiting their effectiveness in tackling the issue comprehensively [3].

With advancements in artificial intelligence, deep learning techniques like Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks have gained traction for fake news detection. These models offer improved capacity to capture complex patterns in textual data [4]. More recently, pre-trained Natural Language Processing (NLP) models, such as the Bidirectional Encoder Representations from Transformers (BERT), have emerged as transformative tools [5]. BERT's ability to understand context and relationships between words in a sentence enables it to outperform both conventional machine learning algorithms and earlier deep learning models [6]. By leveraging its pre-training on large, diverse datasets, BERT can be fine-tuned for specific tasks, including fake news detection, offering a more flexible and robust solution.

Several studies have explored BERT's application in detecting fake news. For instance, Sholikhah et al. (2023) integrated BERT with a Random Forest Classifier, achieving an accuracy of 67% and an F1-score of 0.67 for the positive label on validation data [7]. Similarly, Amiri (2021) implemented tokenized BERT with a Linear Regression model, attaining 89% accuracy [8]. In contrast, Hanum et al. (2024) achieved 76% accuracy using BERT trained with the AdamW optimizer and the NLLLoss function, while a Multilingual BERT model yielded a lower performance at 63%[9]. Despite BERT's overall potential, these studies highlight its inconsistent outcomes and indicate that further optimization and architectural improvements are necessary for maximizing its effectiveness in specific contexts like Indonesia's multilingual and culturally diverse environment.

BERT has frequently been utilized as a word embedding tool, combined with either machine learning or deep learning algorithms for classification tasks [10]. While these methods demonstrate BERT's capabilities, they often fail to fully harness its contextual language understanding potential. This limitation is partly attributed to the suboptimal design of classification architectures built on top of BERT's output. The lack of specialized and optimized classification layers has resulted in models with subpar performance, particularly in tasks like fake news detection, where subtle contextual differences are crucial.

This study seeks to address these limitations by proposing an enhanced BERT architecture specifically tailored for fake news detection. The proposed approach introduces a more complex classification layer designed to improve BERT's contextual understanding and accuracy. Furthermore, the optimized model will be integrated into a Flask-based web application, allowing users to automatically classify news titles as "hoax" or "factual." This system aims to provide a practical and scalable solution to combat fake news, empowering users with a tool to navigate the digital landscape critically and responsibly.

II. METHOD

This study uses the Design Science Research Methodology (DSRM) approach for fake news detection. DSRM is one of the most widely used research methods for problem-solving in information technology and data science [11]. DSRM consists of several main stages: problem identification, solution objective definition, design and development, demonstration, evaluation, and communication. However, in this study, a modified version of the DSRM model is used to adjust it to the research context. The research stages are illustrated in Figure 1.

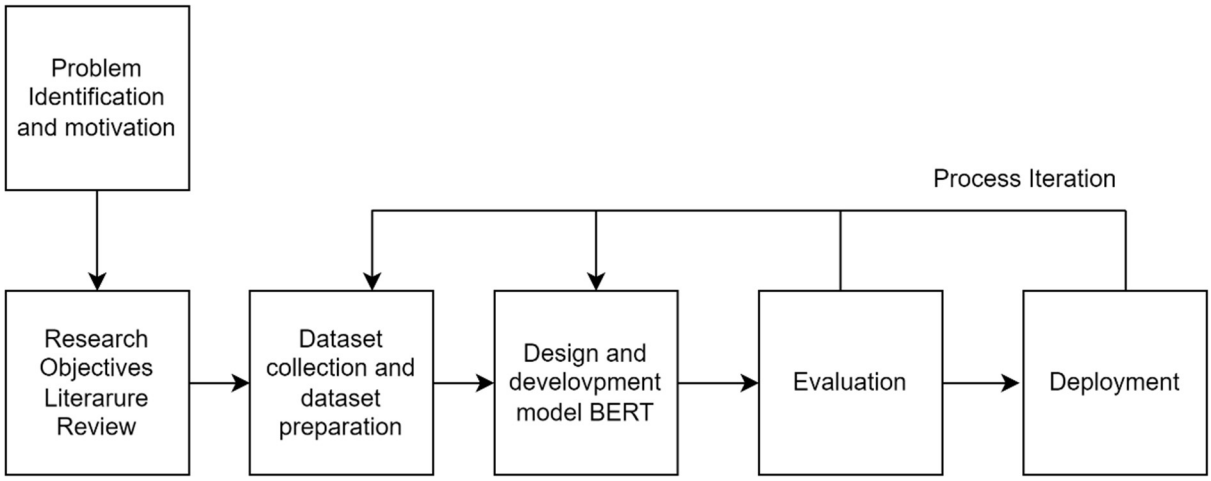

Figure 1. Adapted DSRM for the Proposed Model

A. Problem Identification and Motivation

The first stage involves identifying and defining the research problem clearly. The issue in this study is fake news, which is often crafted to appear indistinguishable from real news, using seemingly reliable sources, convincing language, and even manipulated images or videos [12]. The main motivation of this research is the implementation of the IndoBERT model, which has the ability to understand the context of words bidirectionally, capturing deeper and more complex meanings from text.

B. Research Objectives

This study aims to implement the IndoBERT model for fake news detection, a significant effort in combating the spread of misleading information in the digital age. By leveraging BERT's deep learning power and advanced contextual understanding, this research seeks to develop a more accurate and efficient system for identifying fake news.

C. Dataset Collection and Dataset Preparation

The dataset used in this study was collected through web scraping from detik.com and trunbackhoax.id. The detik.com dataset contains real news articles, while trunbackhoax.id provides fake news articles. The data was collected from January to October 2024, resulting in a total of 10,231 articles from both sources.

Before being used in the model, the input data undergoes preprocessing to improve accuracy and efficiency. The preprocessing steps include data cleaning, labeling, and tokenization [13].

1. Data Cleaning, the data cleaning process was carried out to enhance the quality of the dataset. The cleaning steps included removing duplicate data, feature selection, eliminating irrelevant symbols, and removing ambiguous news articles. As a result of the data cleaning, there are 4,634 article titles from Detik news and 4,429 article titles from Turnbackhoax news.

Table 1. Result Cleaning Dataset

Text Before Cleaning	Text After Cleaning
: "KAPAL FERI BERISI PENGUNGSI ROHINGYA MENUJU INDONESIA"	KAPAL FERI BERISI PENGUNGSI ROHINGYA MENUJU INDONESIA
: "KPU Akui Persiapkan Alat Khusus untuk Gibran saat Debat"	KPU Akui Persiapkan Alat Khusus untuk Gibran saat Debat
PROGRAM PROMO 2024 ***Khusus Seluruh Nasabah BANK JAWA TIMUR***	PROGRAM PROMO 2024 Khusus seluruh Nasabah Bank Jawa Timur

2. Labeling, Fake news was sourced from the Turnbackhoax.com website and assigned the numerical label *1*, while factual news was collected from the Detik.com website and assigned the numerical label *0* . As shown in Figure 2, the dataset used in this study consists of 9,063 news entries. Among these, 4,634 entries (51.1%) are categorized as factual news, while 4,429 entries (48.9%) are categorized as fake news. This distribution demonstrates a relatively balanced proportion between the two categories, enabling effective training of the classification model [14].

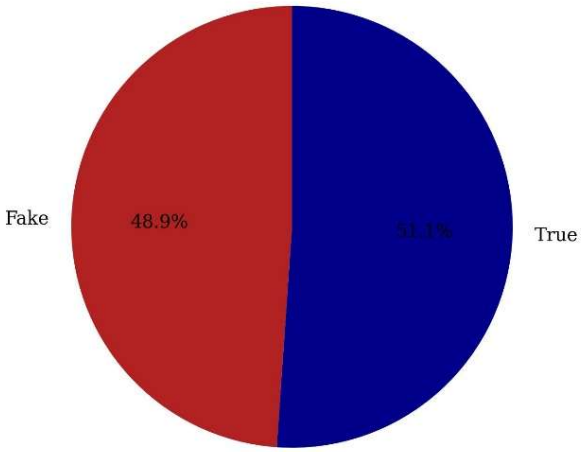

Figure 2. Distribution of Real and Fake News

3. Tokenizer, the tokenization process converts raw text sentences into numerical representations that can be understood by the BERT model. The initial step involves loading the IndoBERT tokenizer using the AutoTokenizer from the Transformers library. This tokenizer is initialized with the pre-trained model 'indobenchmark/indobert-base-p2', a language model specifically trained on a large Indonesian text corpus.
   Before tokenization, a text length analysis is conducted to determine the maximum sentence length to be processed. In this case, the maximum length is set to 20 words, aligning with the average length of news headlines in the dataset. Tokenization and encoding are then performed on all the data, involving several steps: splitting the text into token IDs, adding padding (special tokens) to achieve the maximum length, and truncating text that exceeds the length limit [15]. This process produces outputs such as token IDs, attention masks, and segment IDs, all of which are essential for further processing by the BERT model [16].

4. Data Spliting, the dataset will be split into three subsets: training, test, and validation sets, using a ratio of 80:10:10 [17]. The training set will be used to train the model, the validation set will be used for tuning hyperparameters and preventing overfitting, and the test set will be used to evaluate the model's performance. This data splitting ensures that the model can generalize well and is not biased by the dataset it was trained on.

D. Design and Development IndoBERT Model

The BERT architecture utilized in this study is an enhanced version of the base BERT model, tailored with additional layers specifically for detecting fake news. As the core component, the pre-trained BERT model processes input text to generate contextual representations. These representations, extracted from BERT's pooling layer, are subsequently fed into supplementary layers designed to refine the classification task. This architecture is illustrated in Figure 3.

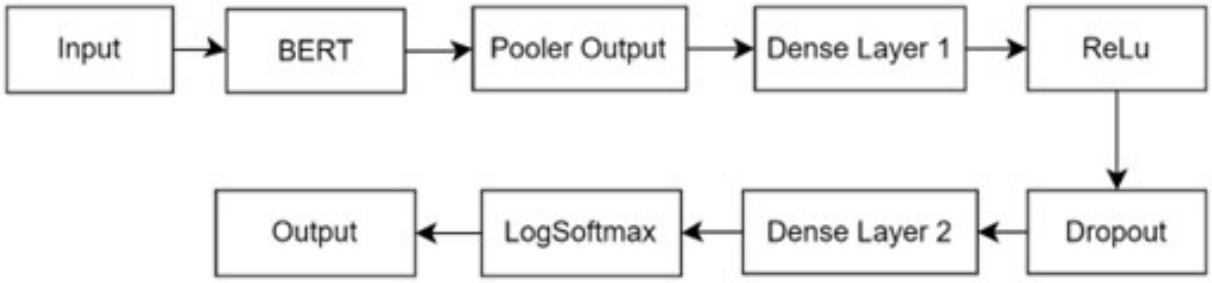

Figure 3. The classification Architecture of IndoBERT

These representations are passed to a dense layer with 768 neurons connected to 512 neurons, using ReLU for non-linearity, followed by a dropout mechanism to prevent overfitting. The output then moves to a second dense layer with 512 neurons connected to 2 output neurons, representing true and fake news classes. Finally, LogSoftmax converts the outputs into log-probabilities, enabling accurate classification by leveraging deep contextual understanding.

IndoBERT adopts the same base architecture as BERT-base, requiring fine-tuning and hyperparameter optimization for optimal classification performance. This study uses proven ranges for various tasks, including batch sizes of 16 and 32 and learning rates of 5e-5, 3e-5, and 2e-5 [18]. The model is trained for three epochs to minimize overfitting due to the limited dataset size, while AdamW optimizer is employed for its ability to adaptively adjust learning rates, ensuring faster convergence and stable training [19], as summarized in Table 2.

Table 2. Hyperparameter Configurations

Hyperparameter Configuration	value
Learning rate	5e-5, 3e-5, 2e-5
Batchsize	16, 32
Dropout	0.1, 0.2, 0.5
Optimizer	AdamW
Epoch	3

E. Deployment

Deployment of the model involves loading and running the trained BERT model within an application accessible to users. In this thesis, BERT is used for fake news detection, and deployment refers to integrating the model into a web application powered by Flask. Flask, a Python microframework, serves as the backend for the application, enabling user interaction with the deep learning model [20]. The BERT_Arch model converts input text into vector representations, which are then processed through dropout, linear, and softmax layers for classification into two categories: hoax or fact. Users can input news headlines through a form on the main page, and the application predicts whether the news is a hoax or fact using the model loaded from a .pth file. The prediction results are displayed directly on the page, with the application running on a Flask server in debug mode for development ease.

III. RESULTS AND DISCUSSION

A. Model Evaluation

Table 3 presents a comprehensive overview of the model's performance across various combinations of learning rate, dropout rate, and batch size during the training process. The primary metrics evaluated include Training Loss, Validation Loss, and Accuracy, as these indicators collectively reflect the model's learning behavior and its ability to generalize to unseen data. Training Loss illustrates how effectively the model fits the training dataset, while Validation Loss serves as a critical measure for detecting overfitting or under fitting. A lower and more stable Validation Loss indicates

that the model maintains good generalization performance. Accuracy is used to quantify the overall classification effectiveness of each configuration. By comparing these metrics across different hyperparameter settings, it becomes possible to identify configurations that achieve an optimal balance between learning stability and predictive performance. The results in Table 3 demonstrate that certain combinations lead to more consistent loss convergence and higher accuracy, highlighting the significant impact of hyperparameter selection on model performance.

Table 3. Result Training Model

No	Learning Rate	Dropout	Batch size	Training Loss	Validation Loss	Accuracy
1	2e-5	0.1	16	0.045	0.236	0.931
2	2e-5	0.1	32	0.041	0.234	0.964
3	2e-5	0.2	16	0.049	0.286	0.942
4	2e-5	0.2	32	0.057	0.200	0.947
5	2e-5	0.5	16	0.056	0.226	0.960
6	2e-5	0.5	32	0.051	0.339	0.924
7	3e-5	0.1	16	0.040	0.213	0.951
8	3e-5	0.1	32	0.054	0.205	0.944
9	3e-5	0.2	16	0.046	0.245	0.955
10	3e-5	0.2	32	0.052	0.192	0.958
11	3e-5	0.5	16	0.063	0.374	0.935
12	3e-5	0.5	32	0.046	0.229	0.959
13	5e-5	0.1	16	0.071	0.475	0.928
14	5e-5	0.1	32	0.056	0.255	0.934
15	5e-5	0.2	16	0.067	0.145	0.949
16	5e-5	0.2	32	0.060	0.282	0.948
17	5e-5	0.5	16	0.080	0.460	0.926
18	5e-5	0.5	32	0.053	0.211	0.948

Starting with the learning rate, the results show that a lower learning rate (2e-5) generally leads to the best performance. For instance, with a dropout of 0.1 and batch size of 32, the model achieved the highest accuracy of 0.964 with a relatively low training loss (0.041) and validation loss (0.234). This indicates that a learning rate of 2e-5 strikes a good balance between the speed of learning and model stability, resulting in optimal generalization. On the other hand, using a higher learning rate (5e-5) tends to reduce performance, with a notable drop in accuracy (0.928 at 5e-5 and batch size 16) and a higher validation loss (0.475), suggesting that larger learning rates can hinder the model's ability to generalize effectively.

Regarding dropout, the smallest value (0.1) consistently produced the best results. When dropout = 0.1 and learning rate = 2e-5, the model achieved an accuracy of 0.964, demonstrating the effectiveness of mild regularization. Conversely, higher dropout values, particularly dropout = 0.5, generally led to increased validation loss and decreased performance, indicating that too much regularization could restrict the model's ability to learn effectively.

Finally, batch size also plays a role in model performance. A larger batch size of 32 generally resulted in higher accuracy, particularly when combined with learning rate = 2e-5 and dropout = 0.1. For example, with batch size = 32, the model achieved an accuracy of 0.964, while with batch size = 16 and the same learning rate and dropout, the accuracy was only 0.931. Larger batch sizes appear to improve stability and speed up convergence, leading to better generalization.

The configuration that achieved the highest performance for the proposed model consists of a learning rate of 2e-5, a dropout rate of 0.1, and a batch size of 32. This combination not only resulted in the highest accuracy but also demonstrated a stable and balanced trend between training and validation losses, indicating good generalization. Experimental results further show that increasing the learning rate or dropout value leads to a noticeable decline in model performance, particularly reflected in higher validation loss and reduced accuracy. These findings suggest that excessively large learning rates or

dropout rates may hinder the model's ability to converge effectively and capture meaningful linguistic representations.

B. Deployment

After identifying the model with the highest accuracy, the model will be deployed using a Flask API. The deployment process integrates the trained model into a web application, allowing users to interact with it. The web interface enables users to input news headlines through a form, and the model will predict whether the news is a hoax or fact. The prediction results are displayed directly on the page, providing immediate feedback. The Flask API serves as the backend, handling requests and responses between the model and the user interface. The application is hosted on a server running Flask in debug mode for ease of development and testing. The user-friendly design ensures accessibility, and the application operates seamlessly for both technical and non-technical users. The interface of the web application is shown in Figure 4.

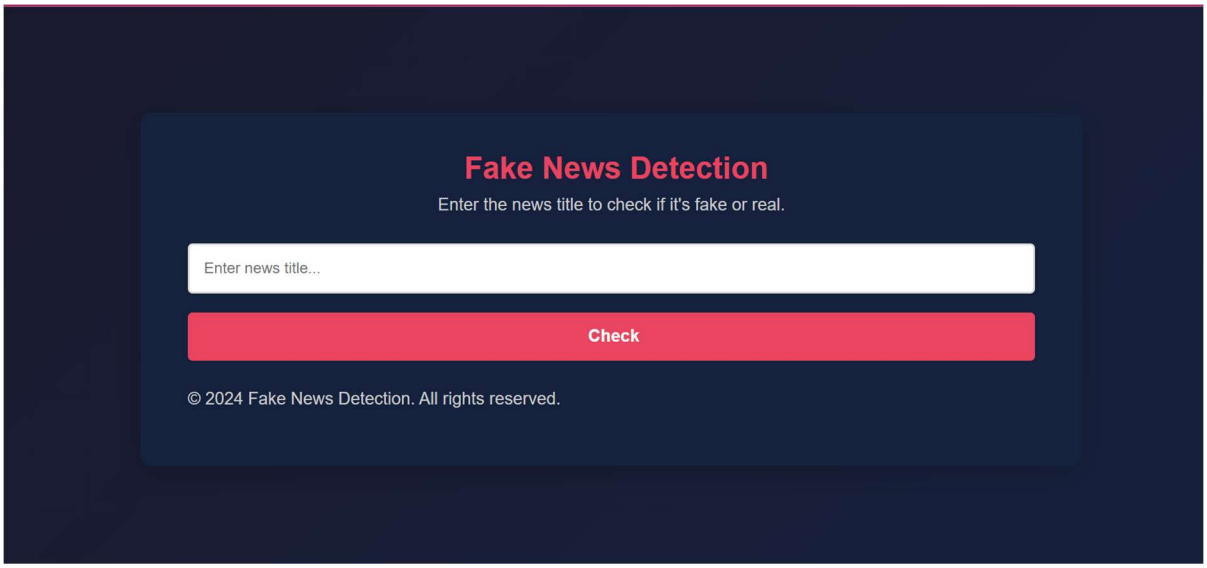

Figure 4. Web Application Page

When a user enters the news title for verification, such as the example in Figure 4 where the user inputs the title "*Perusahaan Asuransi* 'Prudential' *Bangkrut*," the application predicts that the news is a hoax. The web application features include an input field, an action button, and an output area displaying the prediction result. The text box, located at the top or center of the screen, allows users to enter the news title they wish to verify. Below the input field is a "Check" button, which triggers the backend to process the input, send it to the fake news detection model, and await the prediction result. After the verification process is complete, the predicted result is displayed at the bottom of the application in clear text, such as "Hoax" or "Fact."

Figure 5. Result of the News Verification Process Displayed

C. Testing

System testing for the fake news detection model is a testing process aimed at evaluating the performance and effectiveness of the model in identifying fake news while ensuring that the model functions as expected under various conditions and scenarios. This testing involves using new test data consisting of recent news articles collected from various news sources. The goal is to assess how well the model handles news that was not included in the training data. The testing results are presented in Table 4.

Table 4. Testing

No	News Title	News Source	Model Test Result	Manual Test Result
1.	*Gencatan senjata Dilanggar Saat Israel Lancarkan Serangan di Lebanon*	Detik.com	Fact	Fact
2.	*Kerja Sama Bareng China, Menlu Tegaskan Kedaulatan di Natuna Tak Berubah*	Detik.com	Fact	Fact
3.	*Terdakwa Pungli Rutan KPK Rela Pacar Cari Pria Lain: Bahagiamu, Bahagiaku.*	Detik.com	Fact	Fact
4.	*Prabowo Tekankan Indonesia Tak Gabung Blok Pertahanan Mana Pun*	Liputan6.com	Fact	Fact
5.	*Gaji Guru Naik, Puan Ingatka Pemerintah Tak Lupakan Nasib Honorer*	Liputan6.com	Fact	Fact
6.	*Ternyata, Pelaku Pembunuhan Ibu Kandung di Cileungsi Bogor Anggota Porli*	Liputan6.com	Fact	Fact
7.	*Anak Tusuk Ayah dan Nenek hingga Tewas di Cilandak Jadi Tersangka*	Cnnindonesia.com	Fact	Fact
8.	*Mentri Ekstremis Israel Larang Masjid Kumandangkan Azan*	Cnnindonesia.com	Fact	Fact
9.	*Tim Ridwan Kamil Perintahkan Sejumlah Saksi Tolak BAP Rekap Kecamatan*	Cnnindonesia.com	Fact	Fact

No	News Title	News Source	Model Test Result	Manual Test Result
10.	*Gaji UMR Dipotong PPN 12%*	Turnbackhoax.com	Hoax News	Hoax News
11.	*Perusahaan Asuransi "Prudential" Bangkrut*	Turnbackhoax.com	Hoax News	Hoax News
12.	*Presiden Prabowo Mau Liburkan Sekolah Satu Bulan saat Puasa*	Turnbackhoax.com	Fact	Hoax News
13.	*Istri Bunuh Suami yang Mabuk di Belitung karena kesal paksa hubungan badan saat menstruasi*	Twitter	Hoax News	Unverified
14.	*Viral ijazah Gibran Rakabuming Raka dengan predikat Second Class Honours - Second Division disebut setara IPK 2,3.*	Twitter	Hoax News	Hoax News

Data in table 4 show that the testing conducted on 14 news titles showed that the BERT-based model accurately classified 13 of them, with only one misclassification where a fake news item was mistaken as factual. This strong performance aligns with findings in recent studies that demonstrate high accuracy for transformer-based models in fake news detection tasks, with several works reporting classification accuracy well above traditional machine learning approaches when fine-tuned appropriately. For example, comparative research indicates that fine-tuned BERT models can achieve high classification performance across multiple datasets, often outperforming conventional text classifiers and demonstrating the power of contextualized language representations for recognizing subtle differences in text semantics [21]. Nonetheless, even high-performing models may produce occasional incorrect classifications, especially in ambiguous cases where linguistic patterns or stylistic features of fake news closely resemble those of true news. Such misclassification behavior has been acknowledged in the academic literature as a known challenge, emphasizing that even state-of-the-art models are not immune to errors in edge cases or when confronted with subtle rhetorical or contextual nuances that blur the distinctions between classes [22-23].

The presence of a single misclassification suggests that while the model exhibits promising accuracy, there is still room for improvement, particularly in handling ambiguous or linguistically complex cases. Research evaluating explainability and error analysis methods alongside BERT models has underscored the importance of interpreting why certain predictions fail, as this can reveal weaknesses in a model's internal representation or sensitivity to specific features of language [24-25]. Recent advances in multimodal and hybrid classification techniques further support the notion that integrating additional contextual or structural information, such as image content or meta-features, can enhance model robustness and reduce misclassification rates in challenging scenarios [26-30]. Furthermore, comparative evaluations of transformer-based methods consistently highlight that expanding training datasets, applying cross-domain evaluation techniques, and employing ensemble strategies can improve a classifier's ability to generalize beyond the most straightforward examples, thereby addressing limitations observed in isolated misclassification events.

IV. CONCLUSION

Based on the experiment results, this study identifies the optimal hyperparameter combination for fake news detection using IndoBERT. A learning rate of 2e-5 and 3e-5 yielded the highest accuracy, while lower dropout rates (0.1 and 0.2) improved performance, and a batch size of 32 ensured training stability. The best combination (learning rate 2e-5, dropout 0.1, batch size 32) achieved a high accuracy of 96.4%, showcasing the model's potential in classifying fake news effectively. For class 0 (non-fake news), precision was 95.2%, recall was 97.8%, and the F1-score was 96.5%. For class 1 (fake news), precision was 97.7%, recall was 94.8%, with an F1-score of 96.2%. Furthermore, the study successfully developed a fake news detection web application that integrates the BERT model using Flask API. This web application enables users to verify news accuracy easily by entering news titles through a simple interface, providing quick predictions of whether the news is fake or real. To improve the model's

generalization ability, expanding the dataset with diverse sources of news is recommended. Additionally, enriching the dataset with full narratives and visual elements, such as images, could provide more context and help the model detect fake news across different formats and contexts.

REFERENCES

[1] J. P. Baptista and A. Gradim, "Understanding Fake News Consumption: A Review," *Soc Sci*, vol. 9, no. 10, p. 185, Oct. 2020, https://doi.org/10.3390/socsci9100185.

[2] Yopita Desriana Butar, "Analisis Penyebaran Hoax Di Media Sosial Dan Dampaknya Terhadap Masyarakat," *Jurnal Pendidikan, Bahasa dan Budaya*, vol. 3, no. 2, pp. 252–258, May 2024, https://doi.org/10.55606/jpbb.v3i2.3201.

[3] N. F. Baarir and A. Djeffal, "Fake News Detection Using Machine Learning," in *2020 2nd International Workshop on Human-Centric Smart Environments for Health and Well-being (IHSH)*, 2021, pp. 125–130. https://doi.org/10.1109/IHSH51661.2021.9378748.

[4] R. Yunanto, A. P. Purfini, and A. Prabuwisesa, "Survei Literatur: Deteksi Berita Palsu Menggunakan Pendekatan Deep Learning," *Jurnal Manajemen Informatika (JAMIKA)*, vol. 11, no. 2, pp. 118–130, 2021, https://doi.org/10.34010/jamika.v11i2.5362

[5] Rajarshi SinhaRoy, "A Study on the journey of Natural Language Processing models: from Symbolic Natural Language Processing to Bidirectional Encoder Representations from Transformers," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, pp. 331–345, Dec. 2021, https://doi.org/10.32628/CSEIT217688.

[6] M. V Koroteev, "BERT: a review of applications in natural language processing and understanding," *arXiv preprint arXiv:2103.11943*, 2021, https://doi.org/10.48550/arXiv.2103.11943

[7] I. I. Sholikhah, A. T. J. Harjanta, and K. Latifah, "Machine Learning Untuk Deteksi Berita Hoax Menggunakan BERT," in *Prosiding Seminar Nasional Informatika*, 2023, pp. 524–531. https://conference.upgris.ac.id/index.php/infest/article/view/3818

[8] Aufa Nabil Amiri, "Deteksi Berita Palsu Otomatis Berbahasa Indonesia Menggunakan BERT," Institut Teknologi Sepuluh Nopember, 2021. https://repository.its.ac.id/91367/

[9] A. R. Hanum *et al.*, "Analisis Kinerja Algoritma Klasifikasi Teks Bert dalam Mendeteksi Berita Hoaks," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 3, pp. 537–546, 2024. https://doi.org/10.25126/jtiik.938093

[10] R. K. Kaliyar, A. Goswami, and P. Narang, "FakeBERT: Fake news detection in social media with a BERT-based deep learning approach," *Multimed Tools Appl*, vol. 80, no. 8, pp. 11765–11788, Mar. 2021, https://doi.org/10.1007/s11042-020-10183-2.

[11] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, Dec. 2007, https://doi.org/10.2753/MIS0742-1222240302.

[12] S. M. Isa, G. Nico, and M. Permana, "Indobert for indonesian fake news detection," *ICIC Express Letters*, vol. 16, no. 3, pp. 289–297, 2022. https://doi.org/10.24507/icicel.16.03.289

[13] H. Murfi, T. Gowandi, G. Ardaneswari, and S. Nurrohmah, "BERT-based combination of convolutional and recurrent neural network for indonesian sentiment analysis," *Appl Soft Comput*, vol. 151, p. 111112, 2024. https://doi.org/10.1016/j.asoc.2023.111112

[14] R. Peranginangin, E. J. G. Harianja, I. K. Jaya, and B. Rumahorbo, "Penerapan Algoritma Safe-Level-Smote Untuk Peningkatan Nilai G-Mean Dalam Klasifikasi Data Tidak Seimbang," *METHOMIKA Jurnal Manajemen Informatika dan Komputerisasi Akuntansi*, vol. 4, no. 1, pp. 67–72, Apr. 2020, https://doi.org/10.46880/jmika.Vol4No1.pp67-72.

[15] T. Wolf, "Huggingface's transformers: State-of-the-art natural language processing," *arXiv preprint arXiv:1910.03771*, 2019. https://doi.org/10.48550/arXiv.1910.03771

[16] Q. Chen, Z. Zhuo, and W. Wang, "Bert for joint intent classification and slot filling," *arXiv preprint arXiv:1902.10909*, 2019. https://doi.org/10.48550/arXiv.1902.10909

[17] K. Devi and A. K. Sharma, "Detection of Brain Tumor Using Novel Convolutional Neural Network with Magnetic Resonance Imaging," in *2023 Seventh International Conference on Image Information Processing (ICIIP)*, IEEE, Nov. 2023, pp. 57–62. https://doi.org/10.1109/ICIIP61524.2023.10537668.

[18] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423.

[19] I. Loshchilov, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017. https://doi.org/10.48550/arXiv.1711.05101

[20] D. J. Evan and P. O. N. Saian, "Implementasi Python Framework Flask Pada Modul Transfer Out Toko Di Pt Xyz," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 8, no. 4, pp. 1121–1131, Nov. 2023, https://doi.org/10.29100/jipi.v8i4.4020.

[21] A. Nayak, "Fake News Detection Using Machine Learning and BERT: A Comparative Study," *Int J Res Appl Sci Eng Technol*, vol. 13, no. 5, pp. 1369–1374, May 2025, https://doi.org/10.22214/ijraset.2025.70428.

[22] K. I. Roumeliotis, N. D. Tselikas, and D. K. Nasiopoulos, "Fake News Detection and Classification: A Comparative Study of Convolutional Neural Networks, Large Language Models, and Natural Language Processing Models," *Future Internet*, vol. 17, no. 1, p. 28, Jan. 2025, https://doi.org/10.3390/fi17010028.

[23] M. Szczepański, M. Pawlicki, R. Kozik, and M. Choraś, "New explainability method for BERT-based model in fake news detection," *Sci Rep*, vol. 11, no. 1, p. 23705, Dec. 2021, https://doi.org/10.1038/s41598-021-03100-6.

[24] M. Al-alshaqi, D. B. Rawat, and C. Liu, "A BERT-Based Multimodal Framework for Enhanced Fake News Detection Using Text and Image Data Fusion," *Computers*, vol. 14, no. 6, p. 237, Jun. 2025, https://doi.org/10.3390/computers14060237.

[25] S. F. N. Azizah, H. D. Cahyono, S. W. Sihwi, and W. Widiarto, "Performance Analysis of Transformer Based Models (BERT, ALBERT, and RoBERTa) in Fake News Detection," in *2023 6th International Conference on Information and Communications Technology (ICOIACT)*, IEEE, Nov. 2023, pp. 425–430. https://doi.org/10.1109/ICOIACT59844.2023.10455849.

[26] R. Kumalasanti and N. M. Dina Aprilianti, "Sentiment Analysis of Bali Calendar Application Reviews using K-Nearest Neighbour," International Journal of Engineering, Technology and Natural Sciences, vol. 6, no. 1, pp. 339, 2024, doi: https://doi.org/10.46923/ijets.v6i1.339

[27] Y. Arta, S. M. Samuri, N. Syafitri, A. Hanafiah, W. Oktaria, and Maripati, "Application of Machine Learning for Classifying and Identifying Security Threats Using a Supervised Learning Algorithm Approach," International Journal of Engineering, Technology and Natural Sciences, vol. 7, no. 2, pp. 182–192, 2025, doi: https://doi.org/10.46923/ijets.v7i2.548

[28] F. C. Utami and F. P. Putra, "Developing an Optical Character Recognition Application for Mobile-Based Recognition and Translation of Arabic-Malay Characters," International Journal of Engineering, Technology and Natural Sciences, vol. 7, no. 1, 2025, doi: https://doi.org/10.46923/ijets.v7i1.437

[29] Sutriawan, A. Z. Fanani, F. Alzami, and R. S. Basuki, "Deep Convolution Neural Network to Solve Problems for Apple Leaf Disease Detection," International Journal of Engineering, Technology and Natural Sciences, vol. 5, no. 2, pp. 232, 2023, doi: https://doi.org/10.46923/ijets.v5i2.232

[30] H. Mahmood, Y. I. Ibrahim, and N. T. Saeed, "Personal Identification Using Palm Features Recognition," International Journal of Engineering, Technology and Natural Sciences, vol. 4, no. 2, pp. 170, 2022, doi: https://doi.org/10.46923/ijets.v4i2.170