

Sentiment analysis of student evaluation feedback using transformer-based language models

Ibnu Daqiqil ID¹, Hendy Saputra¹, Syamsudhuha², Rahmad Kurniawan¹, Yanti Andriyani¹

¹Department of Computer Science, Faculty of Natural Science, Universitas Riau, Pekanbaru, Indonesia

²Departemen of Math, Universitas Riau, Pekanbaru, Indonesia

Article Info

Article history:

Received Jan 14, 2024

Revised Jun 24, 2024

Accepted Jul 14, 2024

Keywords:

Bert

Deep learning

Language model

Sentiment analysis

Student evaluation analysis

Transformer-based model

ABSTRACT

This paper proposes an approach to sentiment analysis of student evaluation feedback using transformer-based language models. The primary objective of this study is to conduct an in-depth analysis of sentiment expressed in student evaluation feedback, with a focus on introducing contextual understanding into the sentiment classification process. In this research, four different variants of transformer language models were assessed, namely multilingual bidirectional encoder representations from transformers (MBERT), IndoBERT, RoBERTa Indonesia, and generative pre-trained transformer (GPT-2 Indonesia). Additionally, we also compared the performance of transformer models with two traditional models, namely support vector machine (SVM) and Naive Bayes (NB). The evaluation was conducted using feedback data collected from the *Evaluasi Dosen oleh Mahasiswa* (EDOM) system at Riau University, which had been categorized as either positive or negative. The outcomes indicate that IndoBERT base uncased exhibits the highest performance, with precision, accuracy, and recall values of 0.858, 0.929, and 0.911, respectively. This observation highlights the effectiveness of transformer-based language models in sentiment analysis of student evaluation feedback and provides insights for improving educational assessment practices.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Ibnu Daqiqil ID

Department of Computer Science, Faculty of Natural Science, Universitas Riau

Campus Bina Widya KM. 12,5, Simpang Baru, Pekanbaru City, Riau 28293, Indonesia

Email: daqiqil@s.okayama-u.ac.jp

1. INTRODUCTION

In higher education, student feedback emerges as a crucial element in the learning process, albeit frequently posing challenges in both its provision and reception. A substantial body of research highlights that constructive feedback plays a central role in nurturing the growth of students as self-reliant learners and significantly contributes to improving their overall learning achievements [1], [2]. Traditional methods of collecting student feedback, such as surveys or questionnaires, frequently exhibit noteworthy limitations. This approach is often perceived as inefficient, lacking transparency, and tends to yield a restricted quantity of feedback responses. As a solution to these challenges, an “*Evaluasi Dosen oleh Mahasiswa*” (EDOM) system was developed. The EDOM system is an information system specifically designed to collect student feedback efficiently and effectively at Universitas Riau (UNRI). Although the utilization of the EDOM system has improved the process of collecting feedback, it currently lacks the capability to automatically analyze student comments. Within the EDOM system, data is collected through a form featuring a Likert scale, with an additional section for comments located at the bottom of the form. Standard database processing can be used to derive results from the Likert scale values. Nonetheless, the difficulty arises in the

comments section, where each survey produces tens of thousands of comments that require manual examination. This presents a substantial challenge, as analyzing such a substantial volume of comments can be time-consuming and resource-intensive. The resolution to this issue involves the application of a facet within the domain of natural language processing (NLP), specifically sentiment analysis [3]-[7].

The purpose of this paper is to tackle the hurdles associated with employing sentiment analysis for the examination of student feedback by implementing transformer-based language models. By efficiently analyzing the data, the results of this analysis are expected to significantly contribute to institutions in improving service quality for students. While earlier studies have explored the impact of sentiment analysis techniques on evaluating student feedback, they have not explicitly addressed the effectiveness and applicability of transformer-based models in this context. To evaluate the capabilities of these models, we will fine-tune pre-trained transformer-based language models, including auto-regressive models like generative pre-trained transformer (GPT-2 Indonesia), and the autoencoder architecture of multilingual bidirectional encoder representations from transformers (MBERT) [8], IndoBERT, and RoBERTa Indonesia [9]. We will evaluate the performance of each model using a student feedback dataset obtained from the EDOM system. The dataset classifies each feedback into two main sentiment categories: positive and negative. We will gauge the efficiency of these transformer-based language models by employing well-established research metrics, including accuracy, F1-score, precision, and recall. This paper is organized into the following sections: section 2 outlines recent works; section 3 presents our proposed methodology; section 4 deliberates on the experimental outcomes, and lastly, section 5 presents our conclusion.

2. RECENT WORK

Transformer-based models have gained significant prominence in recent years owing to their capacity to capture the contextual and emotional intricacies of text [10]-[12]. Transformer-based language models represent a category of deep learning models that exhibit effectiveness across a range of NLP tasks, including sentiment analysis. These models initially grasp the context of a sentence and subsequently employ this contextual understanding to forecast the sentiment of the sentence. Transformer-based models have surpassed earlier state-of-the-art models in performance, and numerous contemporary pioneering models are built upon their foundation [13]. The primary aim of sentiment analysis is to identify, analyze, and extract emotional states, reactions, or sentiments conveyed in textual data. Over recent years, there has been an increasing interest in employing sentiment analysis to automatically assess student reviews [1], [2], [14]-[17]. This is because sentiment analysis can assist universities in identifying trends and patterns within student feedback, which can subsequently be utilized to enhance teaching and learning. For instance, institutions may uncover a consistent positive sentiment regarding particular teaching methods or a recurring negative sentiment associated with a specific course element. Armed with this information, the university can make data-driven decisions to improve teaching and learning. If positive sentiments are associated with particular teaching approaches, those methods can be strengthened and applied in other courses. Conversely, if negative sentiments highlight recurring challenges, focused enhancements can be introduced to tackle those issues, ultimately enhancing the overall educational experience for students. Nonetheless, implementing sentiment analysis within the framework of analyzing student feedback presents its own array of challenges. Existing sentiment analysis techniques, such as GloVe and Word2Vec embedding models [15]-[20], often overlook the sentimental and contextual nuances of the text. These models necessitate extensive training on vast text corpora to produce precise word vectors and may omit out of vocabulary words (OOV), leading to information loss. Moreover, the limited availability of pre-labeled data and the possibility of inconsistencies between reviews and their assigned labels can lead to misclassification [21].

One of the most widely adopted transformer-based language models for sentiment analysis is BERT [22]. BERT demonstrates remarkable effectiveness in sentiment analysis, even when trained on relatively small datasets [22]-[24]. Another prominent transformer-based language model is GPT, although it is not explicitly tailored for sentiment analysis. However, in this research, we intend to incorporate GPT-2 to assess its efficacy in sentiment analysis. Several other transformer-based language models have been put forth for sentiment analysis, including RoBERTa [25], XLNET [26], ALBERT [27], and ERNIE [28]. Nonetheless, the research centers its attention on models applicable to the Indonesian language.

3. METHOD

The methods chapter delineates the sequential procedures undertaken in this research, commencing with data collection and progressing through data pre-processing, data labeling, data splitting, fine tuning, culminating in model performance evaluation. Each of these stages holds significance in guaranteeing the

efficiency and precision of the constructed model. The comprehensive depiction of this entire process is elucidated in Figure 1, offering a visual representation of the research workflow.

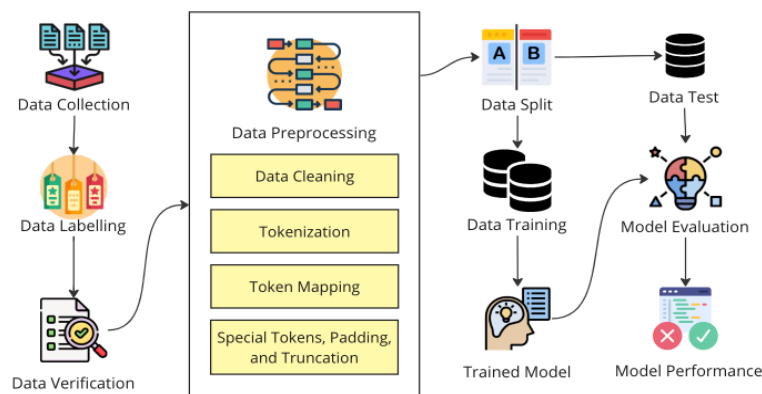


Figure 1. Research stages

3.1. Data collection

In this research, data was gathered from the lecturer evaluation by student’s survey data, which was sourced from the evaluation information system at UNRI. The collected data is in the form of text, encompassing student comments regarding lecturers and lecture activities. Subsequently, this data was transformed into CSV format to facilitate subsequent processing and analysis.

Data collection took place between 2021 and 2023, encompassing inputs from all faculties at UNRI. The total amount of collected data amounted to 11,645 comments. Nevertheless, the data contains a considerable amount of noise, including irrelevant comments, excessively brief remarks, and sensitive or private information such as the lecturer’s name, course, and location. This noise has the potential to impact the clarity and quality of the analyzed data, necessitating data-cleaning procedures to ensure that the analysis centers solely on pertinent and valuable information.

3.2. Data labelling

The data labeling procedure was conducted by a team of five native Indonesian speakers. The CSV data, which had been downloaded, was subsequently partitioned into five separate documents, ensuring that each individual was assigned distinct comments to label. Furthermore, this team had the additional responsibility of pre-filtering the data before the labeling process to guarantee that only pertinent and high-quality data received labels. Prior to labeling, they were tasked with identifying and excluding data that lacked substantial information. This involved filtering out short comments that only consisted of acknowledgments or generic phrases that did not contribute context or additional value for the purpose of data analysis. As a result, this endeavor not only enhanced the quality of the labeled dataset but also improved the efficiency of the subsequent machine learning process, guaranteeing that the model would be trained on genuinely meaningful and representative data.

The total quantity of data successfully labeled amounted to 5,783 comments. Among these, 2,849 were positive reviews, and 2,934 were negative reviews. Figure 2 illustrates that within the EDOM system, positive reviews from students constitute 50.73% of the total, whereas negative reviews make up 49.27%. This indicates a nearly balanced distribution between positive and negative reviews within the cleaned data. Figure 2 shows the proportion of positive and negative data.

3.3. Data verification

After completing the labeling process, the subsequent step involves verifying the labeled data. This stage holds significant importance in guaranteeing the precision and consistency of the data. During the verification process, a distinct team evaluates the labeled data samples. They meticulously assess each assigned label, confirming its appropriateness within the given context. Data that raises doubts or exhibits inconsistencies in labeling undergoes scrutiny and, if deemed necessary, undergoes re-labeling to maintain data integrity. This verification procedure not only enhances the dataset’s quality but also aids in identifying areas that might necessitate improvements in labeling guidelines or additional training for the labelers. The ultimate objective is to generate a dependable dataset, ready for utilization in training machine learning models, with a high level of confidence.

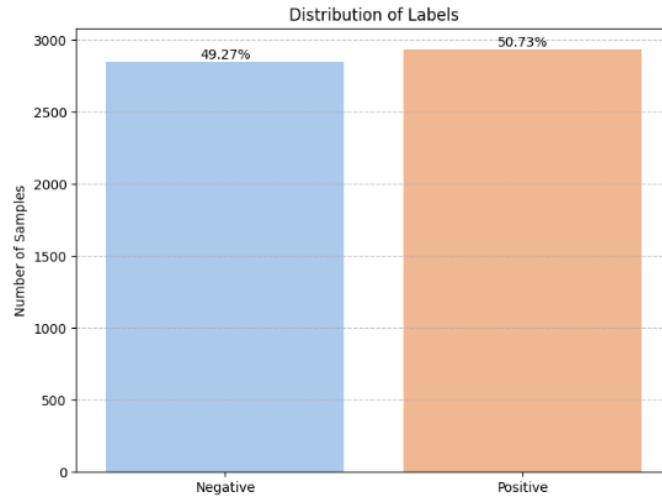


Figure 2. Proportion of positive and negative data

3.4. Data pre-processing

Data pre-processing encompasses several essential tasks, including data cleaning, tokenization, creating token-to-ID mappings, introducing special tokens, and performing padding and truncation, all of which are carried out before fine-tuning and training the model. Figure 3 provides a visual representation of the pre-processing steps.

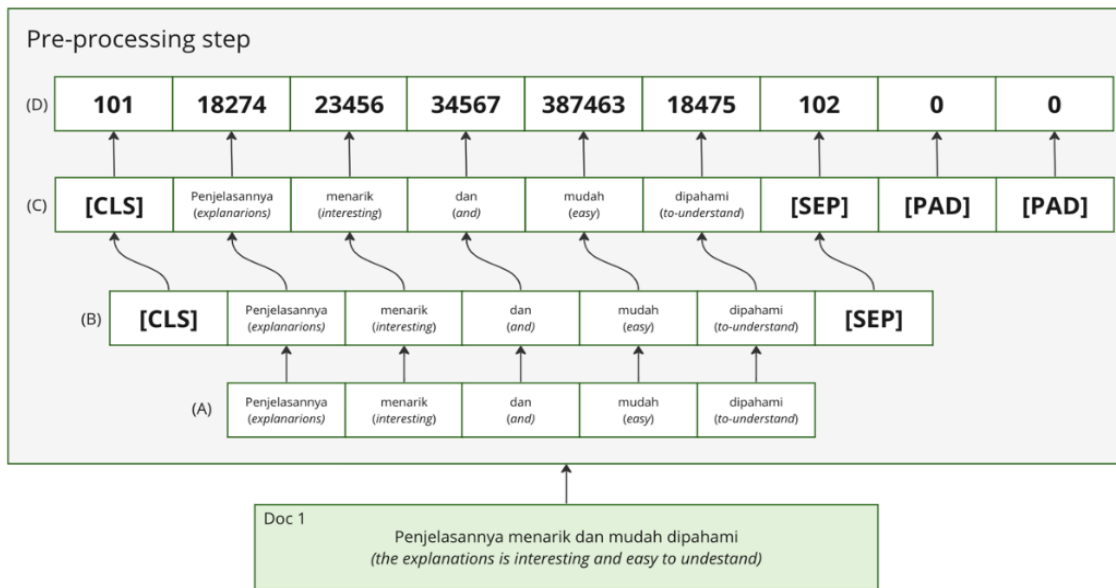


Figure 3. Pre-processing illustration

In this research, the pre-processing stages are delineated as follows:

- a) Data cleaning and tokenisation. The initial step in the pre-processing phase encompasses data cleaning tasks, such as the removal of extraneous symbols, rectification of spelling mistakes, conversion of abbreviations to their expanded forms, and anonymization of names of individuals, locations, or entities through the utilization of substitutes like ‘XYZ’. Subsequent to the cleaning process, tokenization of the document is executed. During tokenization, the input text is dissected into tokens, which can encompass complete words, sub-words, or individual characters. For the purposes of this research, the transformer-based model adopts the WordPiece tokenization approach to segment the text. Table 1 shows an example of data cleansing results. In the example, we censor the name of the class and lecturer and fix some typos

- and grammar errors in the sentence. The resultant cleaned and annotated dataset is accessible in our repository [25].
- b) Add special tokens. Within the BERT architecture, the incorporation of special tokens holds significant importance in conveying structural information to the model. Notably, two such tokens are “CLS” and “SEP.” The “[CLS]” token, also known as the classification token, is positioned at the inception of every input sequence. It serves as a focal point for consolidating the comprehensive representation of the sequence, particularly in tasks related to classification. The “[SEP]” token, referred to as the separator token, assumes the role of enabling the model to differentiate between segments within the sequence. This differentiation proves crucial in scenarios involving distinct text fragments or signifying the conclusion of a sequence.
 - c) Padding, and truncation padding and truncation represent indispensable procedures in the pre-processing of text data for the BERT model, involving the incorporation of [PAD] tokens. Padding involves the addition of [PAD] tokens to shorter sequences, thereby equalizing their length with the longest sequence in the batch. This ensures the model’s ability to efficiently process the data within a single batch. Truncation serves the purpose of reducing the length of sequences that surpass the model’s maximum allowable limit, effectively trimming the text while preserving the fixed dimensionality. Notably, the [PAD] tokens are intentionally engineered so as not to impact the model’s analytical outcomes. They enable the model to disregard the padded sections and concentrate solely on the substantive content contained within the sequences. This procedural approach strikes a balance between preserving crucial contextual information and adhering to the model’s technical constraints, ensuring that only pertinent information is incorporated and processed by BERT.
 - d) Token to ID mapping. The token to ID mapping process within the pre-processing workflow for the BERT model stands as a pivotal stage. It involves the conversion of tokens derived from the original text into distinct identification numbers (IDs). Following the segmentation of the text into tokens based on predefined tokenization rules (such as WordPiece in the case of BERT), each individual token undergoes transformation into an ID based on the model’s vocabulary. To illustrate, the phrase “Hello World” might be represented by the ID 7592 within the BERT lexicon. This procedure transforms the textual structure into an array of numerical values, enabling the model to handle textual data as numeric input. Consequently, a sentence like “Hello World” would undergo conversion into a sequence of numeric IDs (e.g., [7592, 9999]), thereby facilitating the model in its analysis and learning from the data. This mapping process guarantees that the text presented to the model adheres to a uniform format, ultimately enhancing the efficiency and precision of processing.

Table 1. Example of data cleansing record

[1] Before cleansing	[2] After cleansing
[3] <i>Untuk kelas {name of class} dengan bapak {name of lecturer} saya rasa kurang efektif dikarenakan dosen yang bersangkutan jarang masuk kelas dan untuk pengerjaan tugas krg jelas diberikan sehingga mahasiswa bingung tugas nya seperti apa yang harus dikerjakan</i>	[4] <i>Untuk kelas XYZ dengan bapak YXZ saya rasa kurang efektif dikarenakan dosen yang bersangkutan jarang masuk kelas dan untuk pengerjaan tugas kurang jelas diberikan sehingga mahasiswa bingung tugasnya seperti apa yang harus dikerjakan</i>

3.5. Data split

The processed data undergoes division into distinct segments. In the context of this research, the dataset is partitioned into three primary components: training data, validation data, and testing data. The training data, comprising 80% of the entire dataset, is allocated for the purpose of model training. This allocation enables the model to acquire knowledge and adapt to the underlying patterns within the data. The validation dataset, comprising 10% of the entire dataset, serves the purpose of assessing the model’s performance during the training phase. It furnishes feedback on the model’s efficacy and assists in fine-tuning model parameters to mitigate overfitting. On the other hand, the test dataset, also representing 10% of the total data, serves as the ultimate metric for evaluating the model’s performance on previously unseen data. This dataset comprises new data that the model has not encountered before. This division of the dataset adheres to an 80:10:10 ratio, allocating percentages for training, validation, and testing data, respectively.

3.6. Fine tune model

In this research, the pre-trained model architecture is tailored for the purpose of conducting sentiment analysis. These adaptations may encompass alterations to the model’s layers, including the addition or removal of layers, with the objective of enhancing the model’s capacity to effectively learn from and process the distinct characteristics of the curated dataset. Specifically, this research will employ four pre-

trained models, namely MBERT, IndoBERT, RoBERTa Indonesia, and GPT-2 Indonesia. Each of these models belongs to the category of transformer-based models, sharing identical attributes in terms of the number of transformer layers, attention mechanisms, and hidden embedding size. A detailed comparison among the pre-trained models is available in Table 2.

The fine-tuning process included the calibration of hyperparameters for the transformer-based language models, aiming to optimize their performance in the task of sentiment analysis. The adjustment of hyperparameters was guided by the specific attributes of the student feedback data, taking into account factors like the feedback's length and the distribution of positive and negative reviews. The hyperparameters taken into consideration for the proposed model are as follows:

- The batch size was set to 32. The batch size denotes the quantity of training examples employed in a single iteration. A smaller batch size was selected to promote the model's ability to generalize effectively to new, unseen data.
- The AdamW optimizer was used, which is a variant of the Adam optimizer that provides weight decay by decoupling the gradient of the L2 regularization from the update of Adam [29]. The learning rate (lr) was configured to be $2e-5$, and epsilon (eps) was established at $1e-8$. The learning rate governs the magnitude of adjustments made to the model in response to the estimated error during each update of the model's weights. Epsilon serves as an exceedingly small value, implemented to safeguard against division by zero.
- The number of epochs was set to 4. An epoch signifies a single iteration through the complete training dataset. The number of epochs serves as a hyperparameter, determining how many times the learning algorithm will iterate over the entire training dataset.

The customised model architecture is as follows:

Table 2. Parameter's comparison of pre-trained models. L=numbers of transformer layers, H=numbers of hidden embedding size, and A=numbers of attention heads

[5]	Pre-trained model	[6]	L	[7]	H	[8]	A	[9]	Total parameter	[10]	Language type
[11]	BERT-base multilingual	[12]	12	[13]	768	[14]	12	[15]	110 million	[16]	Multilanguage
[17]	IndoBERT-base	[18]	12	[19]	768	[20]	12	[21]	124.5 million	[22]	Mono language
[23]	RoBERTa	[24]	12	[25]	768	[26]	12	[27]	125 million	[28]	Multilanguage
[29]	GPT-2 Indonesia	[30]	12	[31]	768	[32]	12	[33]	117 million	[34]	English

3.6.1. MBERT architecture

MBERT, short for multilingual BERT, represents a variant of BERT that undergoes training not solely on text from a single language but on an extensive collection of text spanning multiple languages. This unique training approach equips MBERT with the capability to comprehend and handle information across a multitude of languages, as opposed to being limited to a single language. MBERT employs an identical architecture and self-attention mechanism as the original BERT. However, due to its training with multiple languages, MBERT acquires the capacity for transfer learning across diverse languages.

BERT, an acronym for BERTs, constitutes a groundbreaking advancement in the field of NLP. The BERT architecture leverages transformers to acquire language representations through the examination of bidirectional context. Transformers represent a category of neural network architecture characterized by a self-attention mechanism. This mechanism enables the model to concurrently process words within a sentence and comprehend the contextual relationships among them by considering the entire sentence. Consequently, this approach facilitates more rapid and in-depth comprehension when compared to sequential architectures that process words one after the other.

Before embarking on the fine-tuning process, it is imperative to prepare and tokenize the input data. Each textual segment is disassembled into discrete tokens, which BERT subsequently maps to corresponding token IDs according to its predefined vocabulary. As an illustrative example in Figure 4, consider the sentence “*Penjelasan menarik dan mudah dipahami,*” which would be segmented into a token list as follows: [“[CLS]”, “*Penjelasan*”, “*menarik*”, “*dan*”, “*mudah*”, “*dipahami*”, “[SEP]”]. These tokens are then translated into a series of numerical IDs. Within BERT's framework, the embedding matrix E is used to transform each token t_i into $E(t_i)$, a vector of numeric IDs. This vector encapsulates both the semantic and syntactic information of the tokens and serves as the initial input for the BERT network. Subsequently, the network utilizes this embedding vector to acquire an understanding of the context and interrelationships among tokens within a sentence throughout both the pre-training and fine-tuning phases.

The embedding of the [CLS] token, $h_{[CLS]}$, from the final layer of BERT is used as a summarized representation of the entire input sequence. Next, this vector undergoes processing through a linear classification layer, which in turn converts it into a two-dimensional vector that aligns with the number of

classes in a binary classification task. The linear transformation is given by the vector h is used as input of the final fully-connected classification layer. Given the parameter $W \in \mathbb{R}^{K \times 768}$ of the classification layer, where K is the number of categories. The output of linear or classification layer is calculated as logit Z (1).

$$Z = hW^T \tag{1}$$

Moreover, the value of the logit Z is subjected to a transformation using the SoftMax function. In this particular scenario, there exist two classes: namely, ‘positive,’ and ‘negative.’ The probability value can be observed in (2).

$$P(\text{class}_i) = \frac{e^{Z_i}}{\sum_{j=1}^2 e^{Z_j}} \tag{2}$$

In the fine-tuning process, the model’s parameters are adjusted iteratively to minimize a loss function, typically the cross-entropy loss, which is commonly used for classification tasks. The loss for a single instance is computed (3) where y is the true label of the instance, $P(\text{class}_1)$ is the model’s estimated probability for the positive class, and $P(\text{class}_2)$ is the probability for the negative class.

$$L = -[y \log(P(\text{class}_1)) + (1 - y)\log(P(\text{class}_2))] \tag{3}$$

An overview of fine-tuning a BERT-based model is presented in Figure 4. The fine-tuning process entails sequential iterations over the complete dataset, encompassing prediction generation, loss calculation, and the subsequent adjustment of the model’s parameters. This adjustment encompasses both the transformer layers and the classification head, and it is facilitated through backpropagation alongside optimization algorithms such as Adam. The iterative process continues for several epochs until the loss either reaches convergence or starts to rise on a validation set that is set aside, signaling the conclusion of effective training.

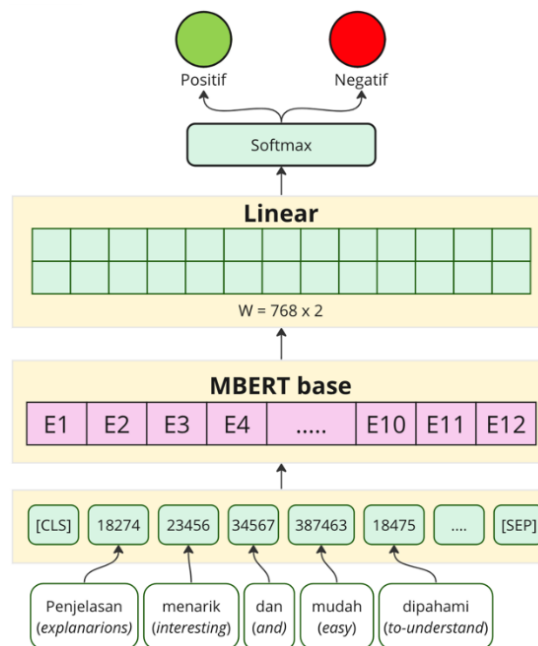


Figure 4. MBERT architecture

3.6.2. IndoBERT system architecture

IndoBERT and MBERT share a common foundation as they are both rooted in the BERT architecture. Each of them employs the transformer model, which encompasses a multi-headed self-attention mechanism and a feed-forward neural network for the processing of input tokens. Both models are engineered with the primary objective of comprehending the context of individual words through the examination of the neighboring words, leveraging the bidirectional training capabilities of the transformer. Additionally, both models employ a consistent tokenization approach, such as the WordPiece algorithm,

which dissects words into subword units. This methodology enhances the model's ability to effectively handle infrequent words and facilitates more robust generalization.

While MBERT undergoes training on a corpus encompassing text from 104 languages, including Indonesian, IndoBERT, in contrast, is trained exclusively on an extensive Indonesian corpus. This specialized training equips IndoBERT with a heightened sensitivity to the intricacies of the Indonesian language, encompassing idiomatic expressions, slang, and syntactical structures. In terms of mathematical representation, the difference lies in the embedding layer, where IndoBERT may have a vocabulary matrix V_{Indo} optimized for Indonesian tokens, as opposed to the more general matrix V_{multi} of MBERT. In (4) and (5) show the different of these methods.

$$E_{IndoBERT}(t_i) = V_{Indo}(t_i) + P(position_i) + S(segment_i) \quad (4)$$

$$E_{MBERT}(t_i) = V_{multi}(t_i) + P(position_i) + S(segment_i) \quad (5)$$

3.6.3. RoBERTa system architecture

RoBERTa enhances the performance of the BERT model through several significant modifications. Two noteworthy distinctions involve alterations in the loss function and the removal of segment embeddings. RoBERTa eliminates the segment embeddings and the next sentence prediction (NSP) pretraining task, thereby simplifying the input representation. By concentrating exclusively on an extended form of masked language modeling, RoBERTa eliminates the necessity to differentiate between multiple segments within its input. This design decision simplifies the model's architecture and is rooted in the recognition that segment embeddings and next sentence prediction have limited impact on the model's performance in downstream tasks. In the absence of segment embeddings, RoBERTa relies exclusively on the positional information and the inherent content of the tokens themselves. In (6) presents the embedding vector for RoBERTa.

$$E_{RoBERTa}(t_i) = V_{single}(t_i) + P(position_i) \quad (6)$$

The consequence of this distinction lies in the fact that while BERT may inherently perform better in tasks necessitating a profound comprehension of segment relationships, RoBERTa's enhanced training efficiency and streamlined model design position it as a superior choice for tasks where such differentiation holds lesser significance. With regard to the loss function, RoBERTa employs an adapted version of the masked language model loss, featuring dynamic masking. The loss function for RoBERTa, which centers solely on the masked tokens rather than on NSP, is computed according to (7).

$$L_{RoBERTa} = -\frac{1}{N} \sum_{i=1}^N \log P(t_i | t_{masked}; \theta) \quad (7)$$

N is the number of masked tokens, t_{masked} is the masked token, and θ represents the parameters of the model. This contrast in loss functions aligns with RoBERTa's training objective that concentrates exclusively on the prediction of masked tokens, enhancing its language modelling capabilities without the NSP constraint.

3.6.4. GPT-2 system architecture

GPT-2, which stands for GPT-2, is a model created by OpenAI, advancing upon the initial GPT architecture. Fundamentally, GPT-2 is structured as a generative model with the ability to grasp context from a provided text and generate coherent and contextually appropriate text in response. The model adheres to the principles of the transformer architecture, emphasizing self-attention mechanisms to assess the significance of each word within the context of all the words in a given sequence. Having been pre-trained on an expansive and diverse dataset, GPT-2 acquires a comprehensive understanding of the language's general structure and subtleties. This proficiency enables it to excel in a broad spectrum of tasks when subsequently fine-tuned on data that is more specific to those tasks.

In contrast, BERT also relies on the transformer architecture, but its primary emphasis lies in generating extensive bidirectional representations by concurrently considering both left and right context throughout all layers of the model. Consequently, BERT demonstrates exceptional proficiency in grasping the subtleties and semantics of words within their contextual framework. This characteristic renders it particularly well-suited for tasks demanding an in-depth comprehension of language, including sentiment analysis, named entity recognition, and question-answering. The primary distinction between GPT-2 and BERT resides in their respective approaches to context and the specific tasks they are optimized for. GPT-2's unidirectional approach empowers it to forecast the subsequent section of text solely dependent on the

antecedent context, a feature inherently advantageous for tasks like text completion and text generation. Conversely, BERT, by examining context from both directions, furnishes a comprehensive comprehension that proves valuable for tasks demanding precise contextual portrayals. Fundamentally, while GPT-2 focuses on text generation, BERT adopts a more discriminative role, structured to fine-tune toward a specific output according to the input, rather than generating novel content. This differentiation delineates their distinct applications in the field of NLP.

In technical terms, GPT-2 does not employ specific tokens like [CLS] and [SEP], which are commonly used in the BERT model. Instead, GPT-2 processes each input token independently. Once tokenized, each token ID x_i is mapped to a dense vector space yielding the token embedding $E(x_i)$. GPT-2 enhances these embeddings by incorporating positional information to preserve the sequential order of words, a critical aspect for comprehending the structure of language. Thus, the embedding for a token at position i is the sum of its token embedding and its position embedding, represented by $E(x_i) + P(x_i)$.

Following that, the embeddings are passed through the numerous layers within the GPT-2 model. Within each layer of the model, self-attention and neural network operations are applied to iteratively enhance and refine the representation of each token. The result obtained from the ultimate layer of GPT-2 for the given sequence consists of a collection of vectors, and typically, the vector associated with the initial token (or an alternative strategy based on the specific task) is utilized for subsequent classification tasks.

The chosen output vector from GPT-2, frequently the output corresponding to the first token, is subsequently forwarded through a linear layer that functions as the classification head. This layer performs a dimensionality reduction of the high-dimensional vector into the designated label space. The logit z , which signifies the unprocessed classification score, is computed through this linear transformation. In the case of binary classification tasks, the sigmoid activation function is employed on the logit to generate a probability value ranging from 0 to 1, indicating the likelihood of class membership.

3.7. Model performance evaluation

A model assessment was carried out to assess the efficiency of the transformer-based language model in analyzing the sentiment expressed in student reviews. This evaluation encompassed the utilization of various metrics to offer a comprehensive assessment of the model's performance. The Matthews correlation coefficient (MCC) was employed, considering the imbalanced distribution of data labels. The confusion matrix plot served as a visualization tool to depict the classification model's performance. Furthermore, additional evaluation metrics, including accuracy, precision, recall, and the F1-score, were employed. These evaluation outcomes were utilized to ascertain the model's capability to accurately classify reviews as either positive or negative to a satisfactory degree.

4. RESULTS AND DISCUSSION

In the examination of sentiment analysis within student evaluation feedback, four transformer-based language models have been utilized: MBERT base, IndoBERT base, RoBERTa base Indonesia, and GPT-2 Small Indonesia. The effectiveness of these models was thoroughly assessed using a variety of metrics, as outlined in Table 3. From these data, it can be observed that all transformer-based model outperforms traditional model (SVM and Naive Bayes (NB)) and the IndoBERT base uncased model achieved the highest MCC of 0.858, an accuracy of 0.929, and a recall of 0.911. Additionally, the MBERT base uncased model exhibited strong performance, recording the highest F1-score of 0.936. Furthermore, the GPT-2 small Indonesia uncased model demonstrated the highest precision at 0.950, while the RoBERTa base Indonesia uncased model exhibited the least favorable performance metrics among the four models.

Table 3. Performance comparison of fine-tuned language models

Model name	Letter case	Evaluation parameters				
		MCC	Accuracy	Precision	Recall	F1-score
MBERT base	Uncased	0.850	0.927	0.945	0.927	0.936
IndoBERT base	Uncased	0.858	0.929	0.951	0.911	0.931
RoBERTA base Indonesia	Uncased	0.782	0.891	0.887	0.891	0.889
GPT-2 small Indonesia	Uncased	0.839	0.920	0.950	0.905	0.927
SVM	-	0.732	0.743	0.736	0.745	0.744
NB	-	0.634	0.645	0.665	0.642	0.647

An in-depth comparison of performance metrics for sentiment analysis reveals the models' performance characteristics. Commencing with the MCC, the IndoBERT base model stands out with an MCC of 0.858, slightly surpassing MBERT base's 0.850. This suggests a slightly higher overall quality in

binary classification for IndoBERT base. Although the advantage is modest, it can hold significance, especially when assessing subtle sentiments conveyed in student feedback. Conversely, GPT-2 small Indonesia, boasting an MCC of 0.839, and RoBERTa base Indonesia, with a score of 0.782, indicate that while they exhibit reasonable performance, they might not capture sentiment as effectively as the BERT-based models do.

Accuracy measurements further emphasize the tight competition between IndoBERT and MBERT, attaining scores of 0.929 and 0.927, respectively, highlighting their dependability in accurately classifying sentiments. GPT-2 closely follows at 0.920, establishing itself as a robust contender, whereas RoBERTa lags behind at 0.891, which, though lower, still signifies a commendable level of accuracy.

Precision, a crucial metric in sentiment analysis to minimize false positive errors, demonstrated IndoBERT base as the leader with a score of 0.951. This was closely followed by GPT-2 small Indonesia, which achieved a precision of 0.950, indicating its proficiency in confidently identifying positive sentiment. MBERT base was not far behind with a precision score of 0.945, while RoBERTa base Indonesia had the lowest result at 0.887.

Considering recall, MBERT base takes the lead with a score of 0.927, indicating its capacity to encompass a substantial portion of positive sentiments an essential characteristic for comprehensive sentiment analysis. IndoBERT base and RoBERTa base Indonesia exhibit lower recall scores at 0.911 and 0.891, respectively, which could potentially lead to the omission of certain true positives. GPT-2 achieves a recall of 0.905, placing it in a comparable position to IndoBERT by striking a balance between precision and the capability to identify true positives.

The F1-score, which harmonizes precision and recall, highlights MBERT base's proficiency, boasting the highest score of 0.936. This suggests that it achieves a superior balance between these two metrics. IndoBERT base, with an F1-score of 0.931, indicates a slight preference for precision over recall. GPT-2 small Indonesia's F1-score of 0.927 positions it as a well-balanced model, while RoBERTa base Indonesia's score of 0.889 suggests room for potential enhancements in achieving a better balance between precision and recall.

Essentially, IndoBERT base exhibits outstanding precision, while MBERT base showcases an impressive equilibrium across all metrics. GPT-2 small Indonesia consistently upholds its reputation as a precision-oriented model, and even though RoBERTa base Indonesia performs less satisfactorily, it still holds potential for improvement through optimization. Hence, the particular model selected for sentiment analysis of student evaluations would depend on whether the task prioritizes minimizing false positives or guaranteeing the detection of all instances of positive sentiment. To facilitate a more intuitive comprehension, we present this data graphically. Figure 5 through 5 offer visual depictions of the MCC, accuracy, precision, recall, and F1-score for each model, respectively.

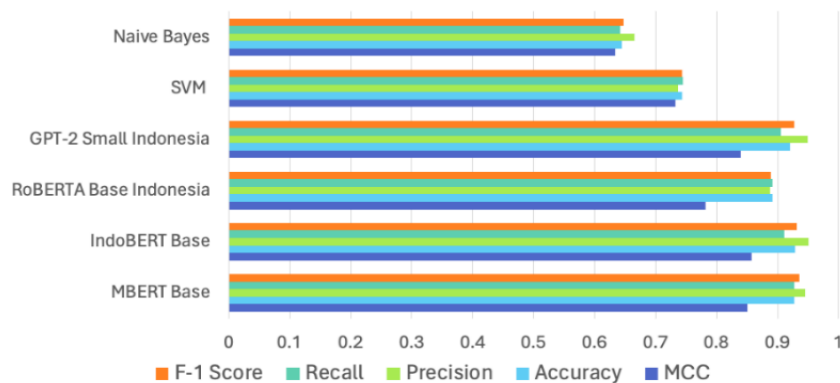


Figure 5. Comparison of performance result

During the fine-tuning process, all models displayed a reduction in training loss as the epochs advanced, signifying successful learning and enhanced performance on the training data. Nevertheless, the validation losses for all models started to increase after the third or fourth epoch, even as the training losses continued to decrease. The discrepancy observed between training and validation loss implies that, although the models are effectively learning from the training data, their ability to generalize to new, unseen data may be limited. Figure 6 illustrates the training and validation loss for these four models. In contrast, the MBERT base uncased model (Figure 6(a)) maintained relatively low validation loss from the second epoch

onwards. In particular, the IndoBERT base uncased model (Figure 6(b)) exhibited a consistent decline in both training and validation loss during the initial three epochs. Nonetheless, by the fourth epoch, there was a slight upturn in the validation loss, suggesting a potential issue with overfitting. However, the RoBERTa Indonesia (Figure 6(c)), and GPT-2 Indonesia models (Figure 6(d)) exhibited more noticeable increases in validation loss at the fourth epoch.

Subsequently, the one-way ANOVA test was employed on this dataset of evaluation metrics. The objective of this test is to ascertain whether there exists a statistically significant difference among the average evaluation metrics of three or more models. The outcomes of the one-way ANOVA test reveal a significant difference between at least two groups, as evidenced by the F-statistic value of 8.842 and the p-value of 0.00071, which is lower than the threshold of 0.05.

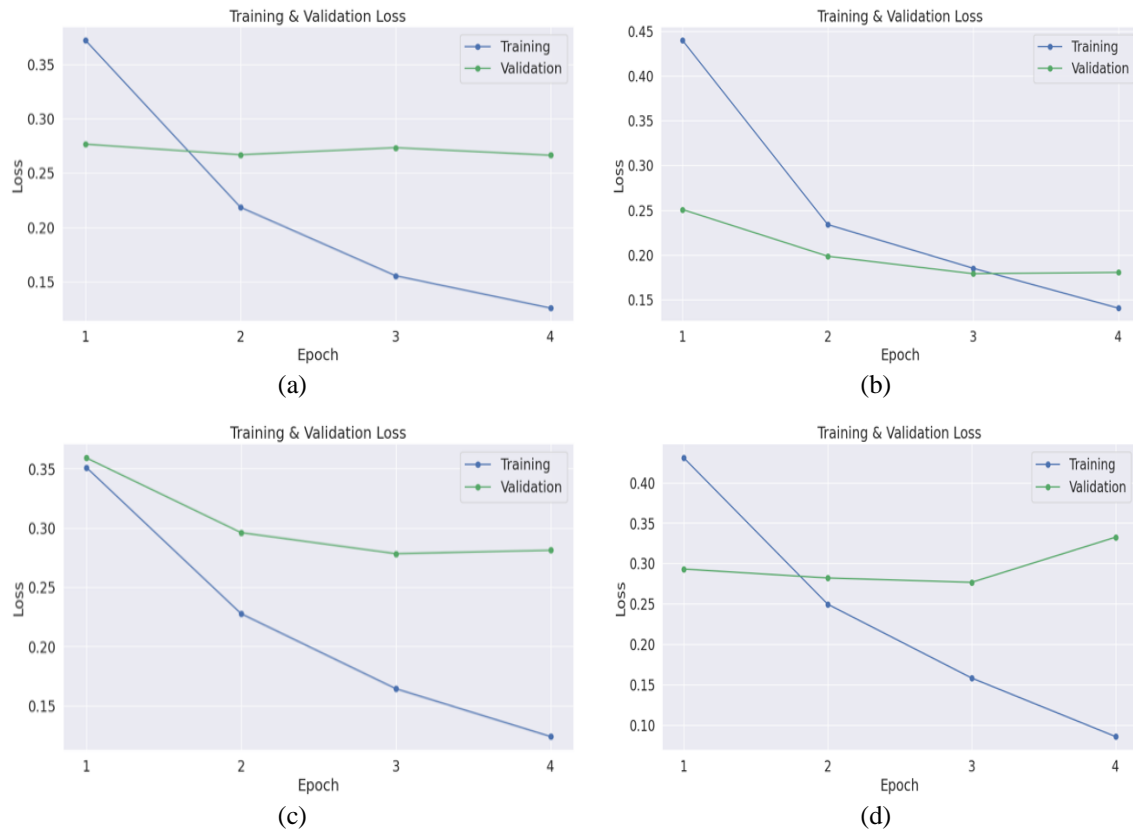


Figure 6. Model training loss: (a) loss metrics of MBERT models, (b) loss metrics of IndoBERT models, (c) loss metrics of RoBERTa Indonesia models, and (d) loss metrics of GPT-2 Indonesia model

5. CONCLUSION

In this research paper, the effectiveness of four fine-tuned pre-trained language models (PLMs) was assessed in the context of a student feedback sentiment analysis classification task. These four PLMs encompassed MBERT, IndoBERT, RoBERTa Indonesia, and GPT-2 Indonesia. The outcomes revealed that the IndoBERT base uncased model yielded the most favorable results, attaining the highest MCC of 0.858, accuracy of 0.929, and recall of 0.911. The MBERT base uncased model exhibited commendable performance as well, recording the highest F1-score (0.936). The GPT-2 small Indonesia uncased model achieved the highest precision (0.950), whereas the RoBERTa base Indonesia uncased model exhibited the least favorable performance metrics among the four models.

ACKNOWLEDGEMENTS

We would like to extend our sincere gratitude for the funding and generous support received from the DIPA of Universitas Riau, under Grant No. 8320/UN19.5.1.3/AL.04/2023, in the year 2023. This support played a pivotal role in the successful completion of our research project.




REFERENCES

- [1] S. Yu and C. Liu, "Improving student feedback literacy in academic writing: an evidence-based framework," *Assessing Writing*, vol. 48, p. 100525, Apr. 2021, doi: 10.1016/j.asw.2021.100525.
- [2] S. Gentrup, G. Lorenz, C. Kristen, and I. Kogan, "Self-fulfilling prophecies in the classroom: teacher expectations, teacher feedback and student achievement," *Learning and Instruction*, vol. 66, p. 101296, Apr. 2020, doi: 10.1016/j.learninstruc.2019.101296.
- [3] A. Shirahatti, V. Rajpurohit, and S. Sannakki, "Transformer based multi-head attention network for aspect-based sentiment classification," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 1, pp. 472–481, Apr. 2022, doi: 10.11591/ijeecs.v26.i1.pp472-481.
- [4] N. Raychawdhary, N. Hughes, S. Bhattacharya, G. Dozier, and C. D. Seals, "A transformer-based language model for sentiment classification and cross-linguistic generalization: empowering low-resource African languages," in *2023 IEEE International Conference on Artificial Intelligence, Blockchain, and Internet of Things, AIBThings 2023 - Proceedings*, Sep. 2023, pp. 1–5, doi: 10.1109/AIBThings58340.2023.10292494.
- [5] A. R. Lubis, Y. Fatmi, and D. Witasryah, "Comparison of transformer based and traditional models on sentiment analysis on social media datasets," in *Proceedings - 2023 6th International Conference on Computer and Informatics Engineering: AI Trust, Risk and Security Management (AI Trism), IC2IE 2023*, Sep. 2023, pp. 163–168, doi: 10.1109/IC2IE60547.2023.10331232.
- [6] Z. Kastrati, F. Dalipi, A. S. Imran, K. P. Nuci, and M. A. Wani, "Sentiment analysis of students' feedback with NLP and deep learning: A systematic mapping study," *Applied Sciences (Switzerland)*, vol. 11, no. 9, p. 3986, Apr. 2021, doi: 10.3390/app11093986.
- [7] K. F. Hew, X. Hu, C. Qiao, and Y. Tang, "What predicts student satisfaction with MOOCs: a gradient boosting trees supervised machine learning and sentiment analysis approach," *Computers and Education*, vol. 145, p. 103724, Feb. 2020, doi: 10.1016/j.compedu.2019.103724.
- [8] H. Xu, B. V. Durme, and K. Murray, "BERT, MBERT, or BIBERT? A study on contextualized embeddings for neural machine translation," in *EMNLP 2021 - 2021 Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2021, pp. 6663–6675, doi: 10.18653/v1/2021.emnlp-main.534.
- [9] 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 *COLING 2020 - 28th International Conference on Computational Linguistics, Proceedings of the Conference*, 2020, pp. 757–770, doi: 10.18653/v1/2020.coling-main.66.
- [10] Z. Wu, S. Huang, R. Zhang, and L. Li, "Video review analysis via transformer-based sentiment change detection," in *Proceedings - 3rd International Conference on Multimedia Information Processing and Retrieval, MIPR 2020*, Aug. 2020, pp. 330–335, doi: 10.1109/MIPR49039.2020.00074.
- [11] A. K. Durairaj and A. Chinnalagu, "Transformer based contextual model for sentiment analysis of customer reviews: a fine-tuned BERT a sequence learning BERT model for sentiment analysis," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 11, pp. 474–480, 2021, doi: 10.14569/IJACSA.2021.0121153.
- [12] S. Kanmani and S. Balasubramanian, "Leveraging readability and sentiment in spam review filtering using transformer models," *Computer Systems Science and Engineering*, vol. 45, no. 2, pp. 1439–1454, 2023, doi: 10.32604/csse.2023.029953.
- [13] A. Gillioz, J. Casas, E. Mugellini, and O. A. Khaled, "Overview of the transformer-based models for NLP tasks," in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, FedCSIS 2020*, Sep. 2020, pp. 179–183, doi: 10.15439/2020F20.
- [14] I. Solaiman *et al.*, "Release strategies and the social impacts of language models," *arXiv preprint*, 2019, [Online]. Available: <http://arxiv.org/abs/1908.09203>.
- [15] R. Adipradana, B. P. Nayoga, R. Suryadi, and D. Suhartono, "Hoax analyzer for Indonesian news using mns with fasttext and glove embeddings," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 4, pp. 2130–2136, Aug. 2021, doi: 10.11591/eei.v10i4.2956.
- [16] I. M. Fadhil and Y. Sibaroni, "Topic classification in Indonesian-language tweets using fast-text feature expansion with support vector machine (SVM)," in *2022 International Conference on Data Science and Its Applications, ICoDSA 2022*, Jul. 2022, pp. 214–219, doi: 10.1109/ICoDSA55874.2022.9862899.
- [17] I. N. Khasanah, "Sentiment classification using fasttext embedding and deep learning model," *Procedia CIRP*, vol. 189, pp. 343–350, 2021, doi: 10.1016/j.procs.2021.05.103.
- [18] K. Boonchuay, "Sentiment classification using text embedding for Thai teaching evaluation," *Applied Mechanics and Materials*, vol. 886, pp. 221–226, Jan. 2019, doi: 10.4028/www.scientific.net/amm.886.221.
- [19] M. I. Alfarizi, L. Syafaah, and M. Lestandy, "Emotional text classification using TF-IDF (term frequency-inverse document frequency) and LSTM (long short-term memory)," *JUITA: Jurnal Informatika*, vol. 10, no. 2, p. 225, Nov. 2022, doi: 10.30595/juita.v10i2.13262.
- [20] R. Kusumaningrum, I. Z. Nisa, R. P. Nawangsari, and A. Wibowo, "Sentiment analysis of Indonesian hotel reviews: from classical machine learning to deep learning," *International Journal of Advances in Intelligent Informatics*, vol. 7, no. 3, pp. 292–303, Nov. 2021, doi: 10.26555/ijain.v7i3.737.
- [21] S. Tabinda Kokab, S. Asghar, and S. Naz, "Transformer-based deep learning models for the sentiment analysis of social media data," *Array*, vol. 14, p. 100157, Jul. 2022, doi: 10.1016/j.array.2022.100157.
- [22] N. Raghunathan and K. Saravanakumar, "Challenges and issues in sentiment analysis: a comprehensive survey," *IEEE Access*, vol. 11, pp. 69626–69642, 2023, doi: 10.1109/ACCESS.2023.3293041.
- [23] M. Pota, M. Ventura, R. Catelli, and M. Esposito, "An effective bert-based pipeline for twitter sentiment analysis: a case study in Italian," *Sensors (Switzerland)*, vol. 21, no. 1, pp. 1–21, Dec. 2021, doi: 10.3390/s21010133.
- [24] B. G. Bokolo and Q. Liu, "Deep learning-based depression detection from social media: comparative evaluation of ML and transformer techniques," *Electronics (Switzerland)*, vol. 12, no. 21, p. 4396, Oct. 2023, doi: 10.3390/electronics12214396.
- [25] Y. Liu *et al.*, "RoBERTa: a robustly optimized BERT pretraining approach," *ArVix*, 2019, [Online]. Available: <http://arxiv.org/abs/1907.11692>.
- [26] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, "XLNet: generalized autoregressive pretraining for language understanding," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- [27] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: a lite bert for self-supervised learning of language representations," *8th International Conference on Learning Representations, ICLR 2020*, 2020.
- [28] Y. Sun *et al.*, "ERNIE 2.0: a continual pre-training framework for language understanding," *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, pp. 8968–8975, Apr. 2020, doi: 10.1609/aaai.v34i05.6428.




- [29] Z. Zhuang, M. Liu, A. Cutkosky, and F. Orabona, "Understanding AdamW through proximal methods and scale-freeness," *arXiv preprint*, 2022, [Online]. Available: <http://arxiv.org/abs/2202.00089>.

BIOGRAPHIES OF AUTHORS






Ibnu Daqiqil    is a faculty member in the Department of Computer Science at Universitas Riau. He holds a Ph.D. in Interdisciplinary Science and Engineering in Health System from Okayama University, a master's degree in Information Technology from the University of Indonesia, and a bachelor's degree in Computer Science from Brawijaya University. His academic interests lie in machine learning and signal processing, particularly in the areas of semi-supervised learning and nonstationary learning. He can be contacted at email: ibnu.daqiqil@lecturer.unri.ac.id.






Hendy Saputra    is currently pursuing his studies in the Information Systems program at Universitas Riau. His academic interests are deeply rooted in the realms of Machine Learning and Deep Learning, as well as the development of Intelligent Systems. As a student, he is likely engaged in projects and research that contribute to advancements in these cutting-edge fields, preparing him to make significant contributions to the technology sector. He can be contacted at email: hendy.saputra@student.unri.ac.id.






Syamsudhuha    serves as a faculty member in the Department of Mathematics at Universitas Riau. He has an impressive academic background, having earned his Ph.D. in Mathematics from the University of Manchester Institute of Science and Technology (UMIST) in the United Kingdom. He also obtained his Master's degree in Mathematics from Pittsburg State University in the United States, and his undergraduate degree from Universitas Riau in Indonesia. His scholarly pursuits are focused on numerical mathematics and computation. He can be contacted at email: syamsudhuha@lecturer.unri.ac.id.



Rahmad Kurniawan    received the BE(IT) degree in Informatics Engineering from the State Islamic University of Sultan Syarif Kasim Ria, Indonesia, in 2011, and the Master of Information Technology (MIT) in computer science from The National University of Malaysia, in 2014. From October 2011–June 2022, he has been a Lecturer with the Department of Informatics Engineering, State Islamic University of Sultan Syarif Kasim Riau. In 2019 he received a Ph.D. from The National University of Malaysia. In 2022 as a Senior Lecturer in the Department of Computer Science, University of Riau. His current research interests include machine learning, expert system, data mining and optimization, big data, and intelligent system. He can be contacted at email: rahmadkurniawan@lecturer.unri.ac.id.



Yanti Andriyani    is a dedicated Information Systems lecturer with a strong academic background. She holds a Bachelor's degree in Computer Engineering from STT-PLN, a Master's degree in Information Technology from the University of Indonesia, and a Doctorate in System Engineering from The University of Auckland. Beyond her role as an educator, Mrs. Yanti actively contributes to research in information systems, focusing on practical applications of contemporary issues. She can be contacted at email: yanti.andriyani@lecturer.unri.ac.id.