IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 14 September 2024, accepted 10 October 2024, date of publication 16 October 2024, date of current version 30 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3481499

RESEARCH ARTICLE

Model Semantic Attention (SemAtt) With Hybrid Learning Separable Neural Network and Long Short-Term Memory to Generate Caption

AGUS NURSIKUWAGUS[1,2], (Member, IEEE), RINALDI MUNIR[3], (Member, IEEE), MASAYU L. KHODRA[3], AND DESHINTA ARROVA DEWI[2]

[1]Faculty of Engineering and Computer Science, Universitas Komputer Indonesia, Bandung 40132, Indonesia
[2]Faculty of Data Science and Information Technology, INTI International University, Nilai 71800, Malaysia
[3]School of Electrical Engineering and Informatics, Bandung Institute of Technology, Bandung 40132, Indonesia

Corresponding author: Agus Nursikuwagus (agusnursikuwagus@email.unikom.ac.id)

ABSTRACT Image captioning is a hot topic that combines a multidiscipline task between computer vision and natural language processing. One of the tasks in the geological field is to make descriptions from the images of geological rocks. The task of a geologist is to write a content description of an image and display it as text that can be used in the future. Interpretation of the object is one of the objectives of the research, which is to traverse the image structures in depth. Shapes, colors, and structures are to be focused on to get the image's features. The problem faced is how the separable neural network (SNN) and long short-term memory (LSTM) have an impact on the caption that can meet the geologist's description. SNN is called Visual Attention (VaT), and LSTM is called Semantic Attention (SemAtt) as an architecture of image captioning. The result of the experiment confirms that the accuracy model for captioning gets BLEU-1 = 0.908, BLEU-2 = 0.877, BLEU-3 = 0.750, and BLEU-4 = 0.510. The evaluation score is compared to those of other evaluators, such as Meteor and RougeL, which get 0.670 and 0.623, respectively. The model confirms that it outperforms the baseline model. Referring to the evaluations, we concluded that the model was able to generate captioned geological rock images that met the geologist's description. Precision and recall have supported the models in providing the predicted word that is suitable for the image features.

INDEX TERMS Separable neural network, LSTM, transformers, captioning, semantic, attention, process innovation.

I. INTRODUCTION

An image is often used as a record of an event or as a picture of the surrounding situation. One of the images used to describe information on the surrounding natural conditions is geological rock imagery. Geological imagery provides visualization of the structure and color of rocks used to determine geological information about the area. Geological imagery is also used as follow-up research material for a particular exploration. The use of geological images is not only for geologists but for other purposes as well. The large number of uses of geological images means that, to facilitate the visualization of geological images, geologists provide descriptions for the geological imagery of these rocks. This task must be carried out by geologists so that the interpretation of the geological image of rocks is following the object content of the geological image of the rocks. The work carried out by geologists is routine when conducting research in certain areas. Giving descriptions to certain images can be repeated for certain rock names, so the recording of descriptions will be repeated as well. Moreover, if the number of geologists is limited, the provision of descriptions will be limited as well.

The task of generating text from the image is known as image captioning. Combination methods of deep learning are needed to be able to generate captions from the image. Computer vision and natural language processing are

The associate editor coordinating the review of this manuscript and approving it for publication was Maria Chiara Caschera.

the fundamentals of the construction of image captioning architectures [1].

Most studies of image captioning have covered several concepts and models that focus on encoder and decoder architecture [2]. The encoder architecture is the task performed to perform image extraction and produce the feature maps [3], [4]. The design architecture mostly uses convolutional neural network (CNN) methods and constructs, including many layers and parameter tuning. Most scholars have leveraged CNN as a backbone model for encoder architectures [2], [5], [10]. E.g., Krizhevsky proposed CNN and ImageNet classification, as well as contributing to the image extraction model [11]. YOLO [12], GoogLeNet [13], VGGNet [14], Resnet [15], and InceptionV3 [16] are models for image recognition.

The other part is the decoder architecture, which generates words that have a relationship with the image area. Each feature map resulting from the task encoder will be paired with word embedding as input for the decoder [17], [18], [19], [20]. The decoder architecture used for image captioning is a recurrent neural network (RNN) and long short-term memory (LSTM) [5], [8], [21], [22]. In addition to these models, the Transformers language models were also deployed to produce sentences that could represent objects in the image [23], [24].

Karpathy was a pioneer in the implementation of image captioning, using the MS COCO images that had extensive annotations [1]. To help find things in pictures, Karpathy made a good contribution. He also suggested a captioning model using CNN architecture and a bidirectional recurrent neural network (BRNN) for MS COCO and FLICKR [1], [16], [22]. Identification of the main object and model architecture is often used as a contribution in captioning research [23], [25], [26].

Primarily, the study was conducted by Vinyals and Toshev using VGG16-LSTM-OneHotVector model development and only got scores of BLEU-1 = 0.5321, BLEU-1 = 0.4890, BLEU-1 = 0.4898, and BLEU-1 = 0.4381 when we used geological rock imagery datasets [27]. Objects detected in rock images, such as humans, various objects, plants, and objects defined on ImageNet [1], [11]. Object such as rocks, sand, soil, rivers, and trees have not been identified in this model. This result has a gap when we compare it with expert descriptions. Some captions still make mistakes when interpreting the image. Visual Graphic Group (VGG) is an extraction model for images, while bilingual under study (BLEU) is a metric for the precision of text.

Several studies have introduced image captioning models that emphasize foreground objects, as in Fig. 1 [28]. If you pay attention, the rock object in the geological image is an object that sets the background of the main object [14]. The proposed semantic attention is more directed at how the results of image feature extraction can relate to text features so that they can predict captions that are similar to reference captions. A study conducted by Chun on image captioning that describes constructed building objects is one of the studies on these topics [28]. E.g., Fig. 1(a) is the result of image captioning to describe a dirty old room.

Another concept for backbone CNN is a separable CNN [29]. Separable CNN, which was introduced by Chollet to make image classification, is an encoder architecture to produce feature maps. On the other hand, LSTM as a language model is to be a pairwise machine learning-encoder architecture. Separable CNN and LSTM are examples of image captioning architectures that can generate captions that relate to the image area [29]. The construction of the backbone CNN by separable CNN is to be a target of the research to produce feature maps.

FIGURE 1. Illustration (a) "a dirty old room" [28], (b) caption in Bahasa "Batupasir kompak retak-retak kelabu".

Encouraged by the success of image extraction and transfer learning in the research, we proposed the solution for the problem in Fig. 2 [1], [28]. Fig. 2 shows the research problem and output caption in Bahasa. Captions produced by geologists can be in the form of rock names and added with words such as color [30].

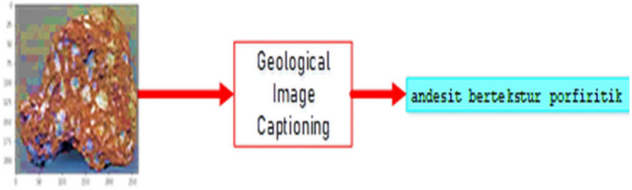


FIGURE 2. Problem research in captioning to geological image in bahasa.

Introducing methods such as separable CNN, regular CNN, LSTM, and transformers can provide valuable insights into proposing image captioning architectures. Model Xception, VGG16, and ResNet50 serve as the image classification methods to achieve the most appropriate feature classification [18], [29], [31]. On the other hand, LSTM, Attention, and Transformers influence providing a good caption that meets human annotation [17], [19], [20]. Based on the rationale and arguments presented in this research, we have proposed a model for captioning geological rock imagery that provides a caption authored by a professional geologist. One of the research's goals is to understand how the alignment process, which generates captions, relates to the object.

The concept of separable CNN, known as Xception, is used as a model of image feature recognition [29]. CNN's separable model, combined with the Transformers language

model carried by [20], was used as a proposed model for captioning rock images. Identification of rock objects with semantic attention to rocks is the main focus of exploring image captioning models. The proposed Attention is a Transformers model consisting of two parts, namely multi-head Attention as an encoder and a decoder. The accuracy and availability of words are targeted when producing captions that match the reference captions from geologists. Every image captioning model is always measured by its success in producing captions. The metrics used are the Bilingual Evaluation Understudy (BLEU) score, RougeL, and Meteor as a model evaluator of the success of the captions produced [32], [33], [34].

Following the explanation in each paragraph and the arguments from the previous study, there are some issues to propose for the contributions. We propose some contributions that offer accomplishments for the study, as follows:

- The architectural model of extracting image features by utilizing the separable CNN method pays attention to the relationship of features between objects. The goal of this contribution is to make an architecture for extracting geological rock images using separable CNN as an encoder [14], [29], [35].
- A word generator (decoder) architecture model for generating captions that are close to reference This problem is used as a target for solving problems to obtain word predictions and generate rock semantics by relying on the LSTM method, Attention, or the Transformers [17], [18], [19], [20].
- The dataset for captioning the geology rock images is structured by following the MSCOCO format [1], [27]. We designed the dataset on our own to encourage the captioning process. We provided 843 images and 4215 references with captions from geologists [30].

To move forward on the paper, we order the paper into sections like introduction, methods, results, discussion, and conclusion. The method contains supporting formulas and algorithms that encourage architectures such as CNN, LSTM, transformers, and model evaluators. The results section displays the experiment the model proposed and generated by the model. The last section is a conclusion that summarizes what the research has learned and future research.

II. PROPOSED METHODS

The proposed method in Fig. 3 is a model carried out during the research. This operation step needs to be described in a diagram so that the direction and output of the research can be understood. Based on the machine learning model, this model is divided into two parts, namely the image descriptor model and the text extraction.

A. IMAGE DESCRIPTORS

This section describes the parameters used in the CNN method. CNN itself has several operating parameters, such as convolutional operations in which there are filters and strides, ReLU activation functions, pooling, FC, and SoftMax functions [35].

The CNN architecture consists of several unique layers, including convolution, activation, pooling, and Softmax layers with different functions. As a baseline model, this study uses the CNN architecture worked on by [27] and [36]. The convolution operation process itself refers to (1):

$$h_{ij}^{k} = \sum_{i \in M_j} \left(\left(w^{k} \times x \right)_{ij} + b_k \right) \tag{1}$$

where k stands for convolution layer k^{-th} and integer for 1 $\cdots$ n, h_{ij}^{k} as a feature matrix, i and j as a pixel position of the object, w^{k} defined as a convolutional kernel at the k position, b_k stands for bias value, bias (b_k) and kernel convolution (w^{k}) was trained by supervise learning [1]. Operation matrix has followed rules of model Lecun [2], and images will be extract into three RGB matrix. The size of matrix can be formatted by (2) and (3).

$$o = (i - k) + 1, \text{if } (s) = 1, \text{and padding } (p) = 0 \tag{2}$$

$$o = (i - k) + 2p + 1, \text{if } (s) = 1 \tag{3}$$

The common cases if the s and p value is a natural value, then dimension of matrix uses (4).

$$o = \left\lfloor \frac{i + 2p - k}{s} \right\rfloor + 1 \tag{4}$$

where O is a dimension of output matrix, i stands for receptive field, k means a kernel/filter/weighted, p for padding.

Aside from the CNN operation, we use a separable CNN to extract the image. Separable CNN performs matrix operations with two techniques, namely spatial separable convolutions and depth-wise separable convolutions. Spatial separable convolution is primarily concerned with the spatial dimensions of the image and kernel, namely width and height. The operation performed on a separable CNN is to separate the values in the matrix into smaller kernels. For example, in the most common case, a kernel with a size of 3 × 3 will be divided into two parts, namely 3 × 1 and 1 × 3. The second separable convolution technique is depth-wise separable convolution. This technique is unlike spatial separable convolution, where the kernel cannot be engineered into a smaller kernel. It is called depth-wise separable convolution because the operation performed is not only spatial in dimension but also depth-wise in dimension, or the size of the number of channels.

The process of depth-wise separable convolution is divided into two parts: a depth-wise convolution and a point-wise convolution. For example, if the original convolution function is 12 × 12×3 → (5 × 5×3 × 256), then a new convolution operation can be made into 12 × 12×3→(5 × 5×1 × 1)→ (1 × 1×3 × 256) →12 × 12×256 [11].

The ReLU activation function nonlinearly maps a characteristic graph of the convolution layer, activating neurons while avoiding overfitting and improving learning ability.

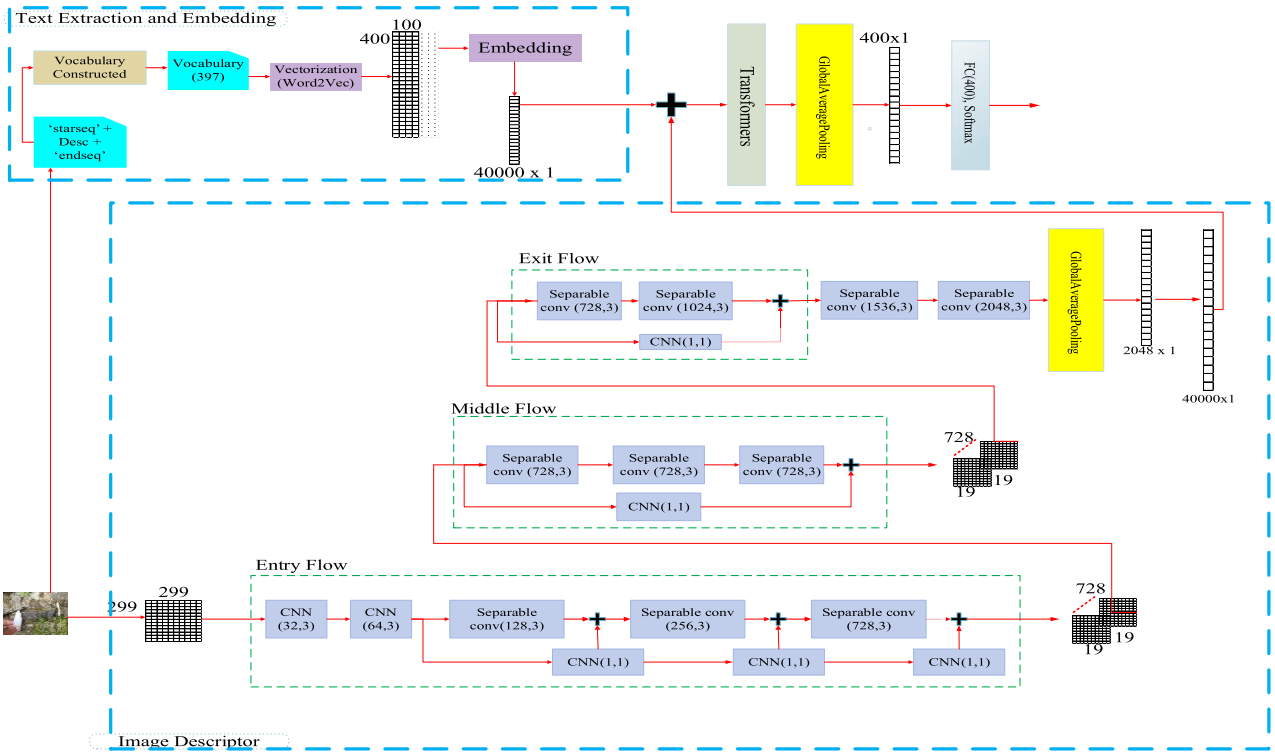


FIGURE 3. Architectures of geological rock image captioning proposed.

This function was introduced in the CNN model [10], [14], [36], [38]. The equation is written as follows:

$$f(x) = \max(0, x) \tag{5}$$

The ReLU function will conceptually reduce colors with dark intensity. The max function will produce a value greater than 0, so that each value if there is a process result of one negative pixel, the pixel will be set to = 0. The function refers to (5) will give a more dominant or bold color.

The pooling layer performs nonlinear downsampling and reduces the size of feature maps, also accelerating convergence and improving computational performance [39]. Models use max-pooling rather than mean-pooling because the former can acquire more texture features than the latter [40]. The max-pooling operation maximizes feature areas of a certain size and is formulated [35]. Equation (6) to calculate the maximum value for every value in the matrix.

R_j stands for pooling region at j in the feature map of i, an index at the region, and h is an output pooling matrix feature map. We use pooling to extract the dominating pixel from the image. We can divide pooling into three categories: maximum pooling, average pooling, and global pooling. Pooling can contribute to reducing image size by a factor of two. Setting the pooling value to two will divide W and H into two. For example, if the map feature size is 6×6 pixels, then the output matrix size is 3×3 pixels.

$$h_j = \underbrace{\max}_{i \in R_j} \alpha_i \tag{6}$$

Each node of the FC layer is connected to all nodes of the top layer. The FC layer is used to synthesize features extracted from images and to convert two-dimensional feature maps into one-dimensional feature vectors [3]. A fully connected layer maps a distributed representation of features to a sample label space. Fully connected operations are formulated [2] as below:

$$a_i = \sum_{j=0}^{m*n*d-1} w_{ij} * x_i + b_i \tag{7}$$

i stands for indexing fully connected, and m, n, d are width, height, and depth for map feature respectively, w for weighted, and b stands for bias.

The SoftMax layer generates a probability distribution over the class by using the output of the second fully connected layer as its input. The highest value of the SoftMax output vector is considered to be the correct index type for rock images. SoftMax function is formulated (8) as bellows [4]:

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \tag{8}$$

The SoftMax function is often called multi-class logistic regression. It is pinned to the SoftMax function because it is the generalization of regression logistics that can be used for multiclass classification.

Categorical cross entropy is a Loss function used for multi-class classification tasks. The performed operation gets only one value from the possibilities of many categories. Loss

categorical entropy function, this uses entropy processing by measuring the magnitude of entropy using (9) [4].

$$CE = -\sum_{i}^{C} t_i \log\left(f\left(s\right)_i\right) \tag{9}$$

where C.E is an abbreviation for Cross Entropy, C stands for how many classes identified form VGG16 process, t_i is a ground-truth value for (y), s_i is a prediction text from LSTM, $f(s_i)$ is an activate function at (5).

B. TEXT PREPROCESSING

This section provides an explanation of how to use the embedding model, specifically word2vec, developed by Thomas Mikolov. The word2vec model is to obtain vector values from a number of existing words with word dimension modes between 50 and 100 dimensions [41]. The word2vec model seeks vector representations in the hope that words are similar, close, and have many-sized similarities.

The neural network language model (NNLM), which lacks a linear layer of hidden layers, is the basis for the continuous bag of words (CBOW) model, as shown in Fig. 4. NNLM, a neural network-based language model, relies on linear projection layers and non-linear hidden layers for feedforward operations. These layers serve as shared learning tools for word vector representations and statistically based language models. The objective function of the CBOW model is tasked with predicting the position of the middle word from the number of N/2 historical words that exist and the number of N/2 words that may appear. The best result of this process is if the word count is $N = 8$, this process produces the best result. The projection layer allows for the easy averaging of vector words from N word contexts. The position of the word has no bearing on determining the middle word in the ''bag of words'' model. The term continuous refers to a D-dimensional vector space. We will process the average vector through the outer layer using a Softmax hierarchy to obtain a distribution on V, which represents the size of the vocabulary. CBOW itself is a log-linear model, which is the logarithm of the result model represented as a combination of model weights. The total weight involved in the CBOW training model is $N \times D + D \times \log_2 V$.

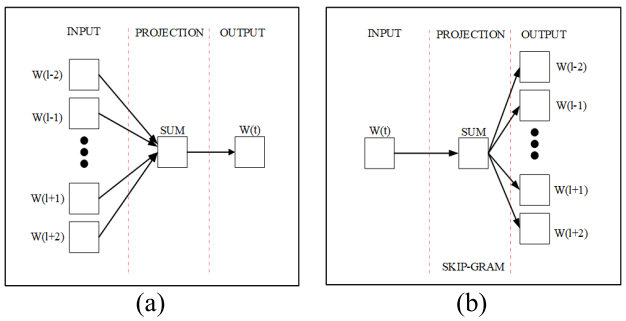


FIGURE 4. (a) CBOW model, (b) skip-gram [5].

This next model is the opposite of the CBOW model, the Skip-gram model in Figure 4(b). The method is to give a current word, the next is to predict the adjacent word context between the word in history and the word future. This model is known as the skip-gram. The value of N is given $N = 10$, from the 10 selected words will be calculated the shortest distance with the word entered, from this result will be labeled as adjacent words. If $N = 10$ and random R are given values between 1 and 10 with the sampling strategy described above, then the historical values of R and the next word will be used as the correct labels of the skip-gram model. The total complexity of this model is $N \times D + N \times D \times \log_2(V)$, provided that N will also be multiplied by $D \times \log_2(V)$ as a problem not in a single classification of CBOW, but a problem of N class. So that overall, the gram skip value of the model will be greater than the CBOW model.

Here is a more detailed pseudocode of the Word2Vec model [5]:

- Corpus preparation: given a corpus of text, vocabulary V is defined as the set of unique words in the corpus, and each word is given an index i in the range [1, |V|].
- Word encoding: every word at the corpus will be represented as OneHotVector |V|. Given word w with index at i, OneHotVector 'x' defined as: $x = [0, \ldots, 0, 1, 0, \ldots, 0]$, and 1 placed on i-th position.
- Training: Word2Vec model is two layers neural network. Input layer is a OneHotVector that supporting value on targeting word at w_t, and output layer is a Softmax function layer that predict a probability every word w_c to be a context word at w_t. Input layer and output layer are connected by weighted matrix W with $|V| * d$ dimension, and d is a dimension from word embeddings.
- The target word that inject by w_t, if $h_t = Wx_t$ is to be a representation layer from w_t, and x_t stands for OneHotVector from w_t. Output layer will calculate a probability Softmax as: $P(w_c \mid w_t) = softmax\left(h_t * W'_{e_c}\right)$, W' is a transpose matrix fromW, e_c stand for OneHotVector encoding from the context word from w_c, and $*$ is a symbolize for matrix products.
- Due the training, model parameter (weighted matrix W) learned by reduce of negative log probability from the observing context target words. This method aligns with maximum average of observing likelihood log probability of context words: $L = \left(\frac{1}{N}\right) * \sum_{i=1^N} \sum_{j \in C_i} logP\left(w_j|w_i\right)$, N is a number of training sample, C_i is a set of target context word from w_i, and $P\left(w_j|w_i\right)$ SoftMax probability from context word target w_j from certainty of w_i.
- Word Embedding: after model learning completely, every matrix rows with weighted w uses as a word embedding.

C. LANGUAGE GENERATORS

The LSTM model used at this baseline is a language model for text generation [6]. In LSTM, all gates will be updated

using the following equation:

$$i_t = \sigma\left(W_{ix}x_t + W_{im}m_{t-1}\right) \quad (10)$$
$$f_t = \sigma\left(W_{fx}x_t + W_{fm}m_{t-1}\right) \quad (11)$$
$$o_t = \sigma\left(W_{ox}x_t + W_{om}m_{t-1}\right) \quad (12)$$
$$c_t = f_t \odot c_{t-1} + i_t \odot h\left(W_{cx}x_t + W_{cm}m_{t-1}\right) \quad (13)$$
$$m_t = o_t \odot c_t \quad (14)$$
$$p_{t+1} = \text{Softmax}(m_t) \quad (15)$$

where i_t, f_t, o_t stand for input, forget, and output gata respectively, $\odot$ is a symbolize for product operator with input value from the gate, W stands for train parameter matrix, σ (.) is a sigmoid function, h (.) is a hyperbolic tangent function, m_t is a repository from product operation between output gate o_t and cell gate c_t, and then p_t is distribution probability for whole words by using Softmax function.

Instead of LSTM, we leveraged the Transformers language model. Vaswani developed the new Transformers model. Most sequence transduction models have an encoder-decoder structure. Here, the encoder maps the input sequence of the symbol representation $(x_1,\ldots, x_n)$ to the continuous representation sequence $z = (z_1,\ldots, z_n)$. Given z, the decoder then generates sequence results $(y_1,\ldots, y_m)$ from symbols one element at a time. The model follows an auto-progressive step, utilizing previously generated symbols as additional input for subsequent predictions. Transformers follow this entire architecture, which uses stacked self-attention and point-wise, fully connected layers for the encoder and decoder [20].

Fig. 5 displays a generator caption model. The language generator is powered by transformers with 22 time steps. The value 22 refers to the length of a word. The output model is always calculating the probability value for each predicted word. The displayed word's position determines the probability value.

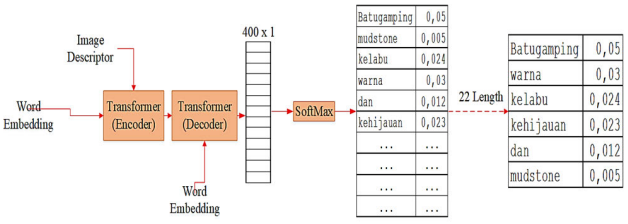


FIGURE 5. Language generator transformers model for 22 length words in Bahasa [7].

D. METRICS

This section evaluates the caption the model produces. Algorithm 1 [32] outlines the process of calculating the BLEU score.

Instead of the BLEU score, we used RougeL and Meteor to support the experiment's explanation. We confirm that the entire score method is viewed from a different perspective. Nevertheless, we identified evaluators who use a similarity process to calculate the distance between the prediction and the references. These evaluators share a common task of measuring precision and recall. Providing and acquiring words from a dictionary is crucial for accurate word prediction. Precision and recall have to be important metrics as a foundation formula in BLEU, RougeL, and Meteor [32], [33], [34].

Algorithm 1 pipeline Process BLEU Score

1. Input token (w_1, w_2, …, w_n).
2. For each caption makes into tokenize (w_1, w_2, …, w_n).
   a. Calculate the variable "count" and "clip-count" from reference token and candidate token,
   b. Compute precision modification
   c. If length of candidate <= reference, calculate brevity penalty (BP)
3. Calculate BLEU

III. RESULTS

We conducted the experiment on the computer system in accordance with the standard process requirements. This experiment used twelve-generation I7 computers, as shown in Table 1. We used a library similar to Python version 3.8 and TensorFlow version 2.x, along with a GPU and 32GB of RAM, which we can expand based on the application needs. The GPU in use is the GeForce RTX 3050. The libraries used to perform this experiment are NumPy, Pandas, strings, Pickle, and OS. Keras 2.3.0 and TensorFlow 2.x are the libraries utilized for model generation.

We deployed the model by reengineering the separable CNN and LSTM components. The model is a novel proposal for a caption model that differs from the baseline. As mentioned in the first chapter, the research aims to explore new approaches to captioning using separable neural networks [29] and Transformer [20] The proposed models are referred to as VaT Visual Attention (VaT) for Xception and Semantic Transformers (SeTrans) for transformers.

TABLE 1. Computer configuration.

Items	Configurations
CPU	Processor Intel(R) Core(TM) i7 12700H CPU @ 2.3 GHz, 20 Core(s), 16 Logical Processor(s)
Graphic Processor Unit	NVDIA GeForce RTX3050, 2304 CUDA cores
Memory	DDR6 4GB
Solid Stated Disk	512 GB
Python	3.8.5
Tensorflow	2.5.0

The separable CNN technique is used to extract identifying edge features by providing faster calculation parameters than regular CNN operations. The expected results of using separable CNN, such as time efficiency and number of train

parameters, result from the separable convolution process. By carefully arranging the Transformers mechanism, you can create a captioning model.

The research aims to prove the hypothesis by observing the significance of the model in producing a high BLEU score. The model engineering carried out is based on the extraction of image features with CNN separable structures, transformers as a word generator, and images using 299 × 299 pixels. Transformers serve as a language model in caption generation, with the input being a vector value from the word2vec word embedding results [41]. The word2vec word embedding was selected based on the success of increasing the BLEU score value in visual attention (VaT) experiments [29] and semantic transformer (SeTrans) [23], [24], [43].

The experiment confirms that there is an effect of lever-454 aging the number of layers and their parameters on captions. 455 The CNN model configuration for image captioning involves 456 parameter tuning on multiple CNN layers (1), filters, strides 457 (2), padding methods (3), (4), (6), and pooling layers (7). The 458 goal of this engineering process is to create a detailed feature 459 map. Some of the parameters that are tuned for CNN are:

- The CNN layer on VaT is followed by Chollet's Xception [24].
- Three parts make up the architectural approach: the entry flow, middle flow, and exit flow.
- The initial layer, the entry flow, receives a 299 × 299 vector input before convolution processes it to produce a 19 ×19 × 728 feature map output.
- Middle flow is an advanced layer that accepts input from an entry flow with a vector size of 19 ×19 × 728 and gives an output shape of 19 ×19 × 728 by repeating eight times.
- The exit flow, the final step in Xception, involves inputting the feature map of the middle flow, which has a shape size of 19 ×19 × 728. In the final stage, pooling is carried out using the Global Average Pooling function to produce units of 2048. We will use flattened units totaling 2048 as input for LSTM and transformers as word generators.
- The ReLU activation function refers to the (5) approach, which allows the value of each matrix to remain on the dominant information.
- Use the ADAM Optimizer as an SGD function to obtain the minimum local and global values of a gradient.

Each experiment yields a unique number of training parameters for each machine, as Table 2 illustrates. The results shown in Table 2 indicate that every CNN deployment will have a trainable parameter. This depends on defining each filter and channel set. The model also produces different BLEU scores [32]. However, the results provided may exceed the baseline.

You can conduct experiments with transformers as part of the decoder using the BLEU results listed in Table 3. Fig. 6 reveals that the generated BLEU surpasses the baseline

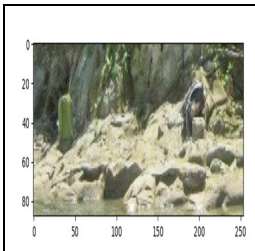
TABLE 2. Number of train parameter for image extraction CNN model.

Model	Xception	VGG16
Trainable parameters	23,626,728	138,357,544

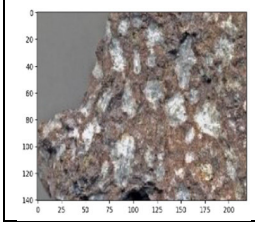
model by more than 40%. SeTrans, as a decoder that produces words, provides less word length than reference captions. Efficiency in producing sentences yields results that go to the object.

Fig. 7 shows the movement of each calculation of the loss and accuracy function with the cross-categorical entropy function referred to in (9). The figure shows a rise in the accuracy value around epoch 25 and above.

Fig. 8 illustrates that the accuracy value increases at epoch 80 and beyond, as indicated in (9). Transformers have two tasks, namely multi-head attention and dot product. This task causes the calculated gradient descent values to slowly converge. The two machines achieve stability accuracy for BLEU values when the epoch exceeds 100.



Prediction:
batupasir batuan sedimen klastik dan besar lapuk
BLEU-1: 0.7143; BLEU-2: 0.5976; BLEU-3: 0.5215; BLEU-4: 0.4347
Reference:
Singkapan batuan sedimen klastik dengan bidang perlapisan yang tidak tegas, batulumpur karbonatan, masif, retak-retak, sebagian hancur dan mulai lapuk



Prediction:
andesit berwarna coklat
BLEU-1: 1.00; BLEU-2: 1.00; BLEU-3: 1.00
Reference:
andesit berwarna coklat yang bertekstur porfiritik

FIGURE 6. Output caption in Bahasa from VaT and SeTrans.

TABLE 3. BLEU score from proposed model.

Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4
VaT (Xception) – LSTM-word2vec	0.93	0.84	0.74	0.54
VaT (Xception) – Trans-word2vec	0.91	0.88	0.75	0.51
VaT (Xception) – GRU-word2vec	0.92	0.88	0.79	0.58

With ADAM optimization, the gradient descent has been tuned to a learning rate of 0.00001 in Figs. 7 and 8. We see the differences between both models. For loss curves, the LSTM in Fig. 7 shows a smoother decline than transformers

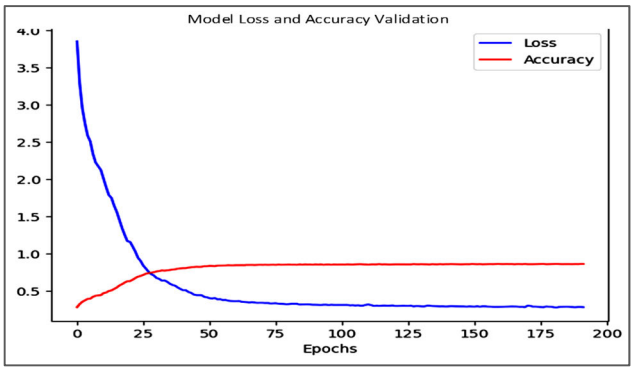


FIGURE 7. Loss and accuracy VaT-transformers curve.

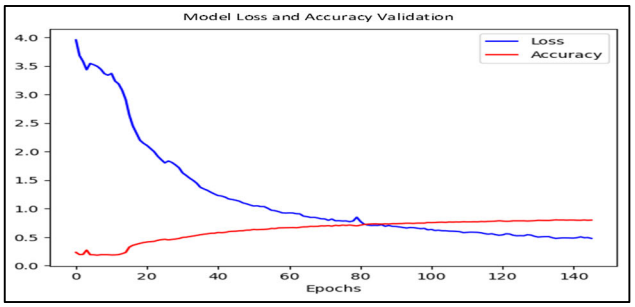


FIGURE 8. Loss and accuracy VaT-transformers curve.

in Fig. 8. The parallel process at Transformers results in a highly graduated change, creates barriers during word generation, and requires a large number of words. On the other hand, LSTM with sequential processing can perform well in generating words, as it has no constraints when selecting the most likely words.

Table 4 compares the use of techniques between VGG16 and the CNN architecture's proposed model. The VGG16 architecture does not use normalization to maintain the number of parameters processed during the extraction process. Table 4 shows that Xception has a longer extraction time than the CNN(3,3) architecture. ADAM optimizer helps speed up feature extraction. Table 4 confirms that the VaT-SeTrans model outperforms VGG16 in terms of extraction recovery time, starting from the extraction process and ending with the production of captions.

The loss calculation refers to (9) demonstrating how to use log likelihood optimization techniques for its parameters. The displayed loss indicates that there is an optimization difference between the parameter's argmax and the resulting model. This difference is based on the empirical distribution of the training set and the probability distribution of the resulting model.

The other results from the research are derived from text preprocessing. Text preprocessing is a task for manipulating the text and transforming it into a real number. The achievement of text processing is carrying out the real number that will be used for the embedding process in transformers, or LSTM.

TABLE 4. Comparison parameter tuning CNN In time.

Model	Layer	Filter	Parameters	Learning Rate	Time (minute)
Vgg16	16	1x1, 3x3, 5x5	138,357,544	ADAM (SGD) Optimizer, lr = 0.0001	243.73
CNN(3,3)	6	3x3	4,310,144	ADAM (SGD) Optimizer, lr = 0.0001	78.97
Xception	16	3x3	23,626,728	ADAM (SGD) Optimizer, lr = 0.0001	113.75

Fig. 9 shows a pipeline of text preprocessing. There are many steps to preprocessing text. Beginning the input text from the dictionary and continuing to clean text like punctuation, comma-delimited, space, and the others that are not needed in the caption. We used string manipulation facilities that the Python library provides. In the pipeline, we proposed word2vec as a word embedding that can produce output matrixes according to dimension requirements.

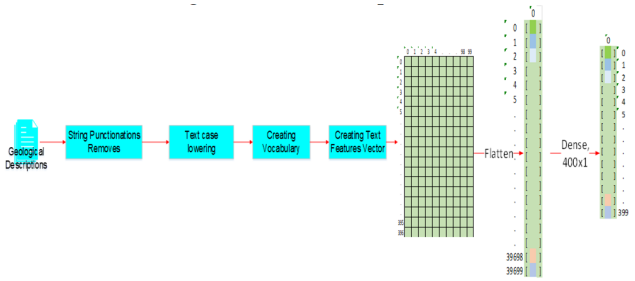


FIGURE 9. Pipeline model for word embedding.

We set the matrix to 400 × 100, following the state-of-the-art approach of Thomas Mikolov [41]. We aligned the dimensions of the matrix with unique words from a dictionary, up to a total of 400 words. Another set with 100 columns; we prefer this size to collect the various words in the dictionary. The process results in a dense 40000 × 1 dimension, which is then applied to LSTM or transformers as an input. The construction of Word2vec, using Bag of Word (BoW) and skip-gram, encourages the creation of a numeric value with a 400 × 100 dimension.

Fig. 10 shows the standard text from geological descriptions. The structure of the description aligns with that of MS COCO [1]. Construction: The text is developed using a streamword that contains the file name and description.

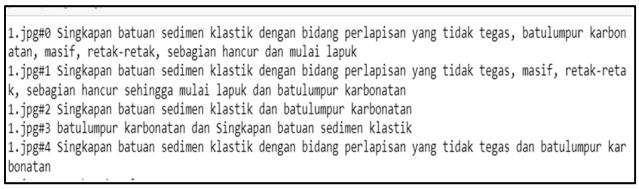


FIGURE 10. Collecting text in Bahasa from annotator.

Fig. 11 appears to be the result of string manipulation. The results are displayed in the lower text. The lower text represents a prerequisite process that must be completed before proceeding to the word embedding process. We must ensure that all process sequences follow the same steps.

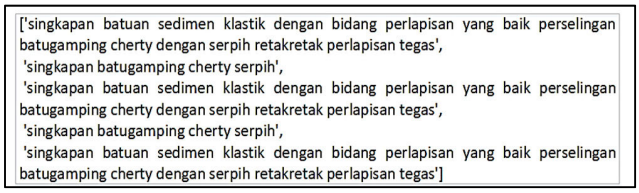


FIGURE 11. The output text in Bahasa from lowering text process.

Fig. 12 shows the matrix output by text preprocessing. Every single value in the matrix corresponds to word2vec results. Word2Vec outputs are the values calculated by calculating the distance of every word that is surrounding the word target. After comparing the value with the other values that have a minimum distance from targeting words, Word2vec stated the value. In the dictionary, we have collected 400 unique words [41].

	1	2	3	4	…	96
1 breksi	-0.013008724	0.09091062	0.017409598	-0.1067254	…	0.10084084
2 abuabu	-0.027804952	0.10471311	0.01711094	-0.11959923	…	0.10930428
3 batupasir	-0.030819757	0.16631457	0.011403339	-0.1913099	…	0.16003962
4 basal	-0.024758015	0.087285034	-0.002059478	-0.10553074	…	0.08015941
5 putih	-0.004322943	0.09487546	0.00035906926	-0.11555044	…	0.10107599
6 coklat	-0.004739953	0.09201679	0.022705337	-0.11300297	…	0.100627735
7 andesit	-0.029149808	0.12189302	0.00908405	-0.12889118	…	0.121709816
… …	…	…	…	…	…	…
… …	…	…	…	…	…	…
… …	…	…	…	…	…	…

FIGURE 12. Matrix 400 × 100 dimension every word in Bahasa on the dictionary. This result gets from Fig. 4, word2vec process.

IV. DISCUSSION

The discussion encompasses the CNN architecture in each model, such as VGG16, Xception, and SeTrans, as well as the word embedding of the resulting geological image description. We consider the VGG16 model as the foundational model for caption development [1], [14]. Some new models frequently compare results with VGG16 [1], [3]. We have developed an architectural model for tasks like image feature extraction using ResNet50, and we are considering using InceptionV3 as an engineering model. This can be attributed to the stability of the extraction process. Additionally, the speed enhances accuracy and minimizes loss during the extraction of image features [13], [16], [29].

Table 5 displays column BLEU 1–4 calculations from the validation dataset, which pertain to section II. The BLEU calculations focus on the precision of the language model used [32]. The use of Transformers and attention as language models for caption generation in the validation dataset has reached values between 40% and 60% for word embedding using word2vec. The model that utilizes word2vec and attention yields BLEU values that surpass the baseline model. The caption process prioritizes the regularity of the generated words, ensuring a significant proximity between the predicted results and the reference captions.

The Xception model approach confirmed that image feature extraction performed better than the proposed machine model. The VaT and SeTrans models confirmed that the obtained BLEU-4 value outperformed the baseline model. This event causes the BLEU score to decrease; the VaT and SeTrans models yield BLEU values of BLEU-1 = 0.91, BLEU-2 = 0.88, BLEU-3 = 0.75, and BLEU-4 = 0.51. On the other hand, the evaluation uses RougeL and Meteor [33], [34].

The comparison between evaluators aims to quantify the characteristics of each model. BLEU and RougeL are evaluators who focus more on the model's precision. These evaluators have gathered data on the word similarity between the predictions and the references. The key to calculations is how many words sequentially appear to have a similar position on the caption [32], [34].

At Table 5, the meteor will count how much ability the dictionary provides the word to serve the prediction word. Another term, it said recall [33]. We employ three evaluators to provide insights into the effectiveness of dictionary collections, enabling us to determine if the collection word is adequate for captioning or not. Language models, such as GRU, LSTM, or Transformers, both provide result support with quality BLEU value achievements [18], [28], [44]. RougeL's results also demonstrate the production of quality sentences, with a rate exceeding 50%. The proposed architecture successfully predicts a caption with precision and word availability [34].

In Table 5, the meteor will measure the extent to which the dictionary can provide a word that matches the prediction word. Another term is recall. We employ three evaluators to gather data on the advantages of dictionary collections, enabling us to determine if each word in the collection is adequate for captioning or not. Language models, such as GRU, LSTM, or Transformers, both provide result support with quality BLEU value achievements [18], [28], [44]. RougeL's results also demonstrate the production of quality sentences, with a rate exceeding 50%. The proposed architecture successfully predicts a caption with precision and word availability [34].

Fig. 13 shows a caption that corresponds to the reference. The VaT and SeTrans models have also succeeded in creating captions that are close to reference. The VGG16-Transformers model confirms this. These results were confirmed to have a low BLEU score compared to other

TABLE 5. Comparison metric evaluation for each models.

Encoder	Word Embedding	Decoder	Att.	BLEU-1	BLEU-2	BLEU-3	BLEU-4	RougeL	Meteor
Vgg16 (Baseline)	One Hot Vector	LSTM	-	0.53	0.49	0.49	0.44	-	-
Vgg16	Word2vec	LSTM	-	0.89	0.81	0.70	0.52	-	-
Vgg16	Word2vec	LSTM	Bahdanau	0.89	0.78	0.66	0.48	-	-
Vgg16	Word2vec	LSTM	Luong	0.85	0.73	0.61	0.46	-	-
Vgg16	Word2vec	Transformers	-	0.86	0.81	0.69	0.52	-	-
Resnet50	One Hot Vector	LSTM	-	0.70	0.68	0.68	0.66	-	-
InceptionV3	Word2vec	LSTM	-	0.66	0.64	0.63	0.60	-	-
Xception	Word2vec	GRU		0.92	0.88	0.79	0.58	0.68	0.64
Xception	Word2vec	LSTM	-	0.93	0.84	0.74	0.54	0.67	0.62
Xception	Word2vec	Transformers	-	0.91	0.88	0.75	0.51	0.67	0.61

models. Some of the results obtained from experimenting with VGG16 include training parameters that reach up to 134 million parameters.

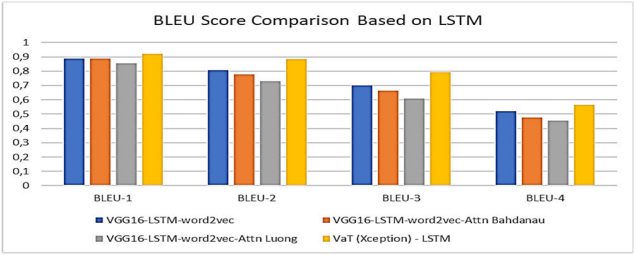


FIGURE 13. BLEU score comparison between VGG16 and VaT. Meanwhile, the language generator using LSTM and Attention [8], [9], [10].

Fig. 14 shows caption overfitting; almost all models produced incorrect captions. Fig. 14 reveals a low precision in the calculated BLEU score.

The VaT and SemAtt models demonstrate that the proposed captioning model for rock geology consistently outperforms the baseline in caption production. Furthermore, the VaT and SeTrans proposals outperform the VaT and SemAtt proposals. When constructing a word in the decoder, the attention values, such as query, value, and key, should be limited to searches for the probability word prediction [20]. We understand that applying LSTM to a small vocabulary can enhance its effectiveness in generating words that align with the feature map area.

Figs. 15 and 14 clearly show that using LSTM as a word generator can provide the best trend epoch when it reaches a turning point below epoch-100. In Fig. 15, see BLEU-4 bars. VaT, a new model of captioning, has confirmed shifting the captions and increased the BLEU-4 score by more than 50%. This indicates that the proposed hypothesis holds importance in the caption results.

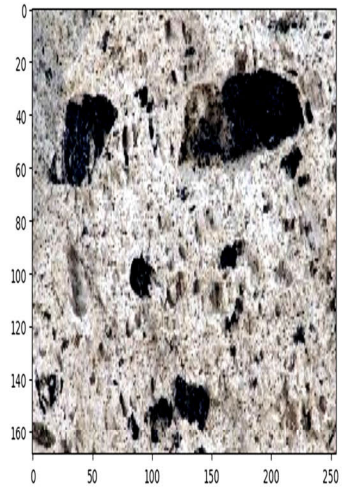
After conducting a series of experiments, it becomes important to prove whether the hypothesis proposed impacts the research objectives. We should test this impact to determine the significance of the effect. Section I's contribution informs the hypothesis. In light of the hypothesis, we must demonstrate that we can discern the results of our experiments by adjusting the parameters of the models. We experimented with the baseline models that the author recommended. After that, we treated the parameters with the new construct of the architectures with tuning parameters and different models, and we found the shifting of captions.

The curve in Figs. 16 and 17 shows that an LSTM consistently generates a better caption than Transformers. The comparison among BLEU-N scores demonstrates this. LSTM leverages the formulas ((10), (11)), (12), (13), (14), and (15) to generate better captions than Transformers. Transformers are better than LSTM in terms of time. Nevertheless, if we attend to the results, providing words in the dictionary is very important to generate a word.

Figs. 17(d) and 17(f) have difficulty reaching the gradient descent point smoothly. When forming a caption, the curve does not smoothly decline to achieve the best accuracy. Transformers imposed a restriction on the ability to search for a specific word. The process in Transformers involves searching for a word in a previously defined query. This has resulted in restrictions on the search for matching words within the specified area.

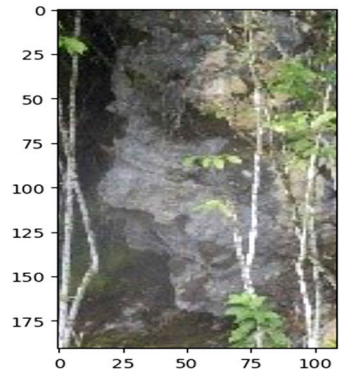
The ANOVA approach is used to test the accuracy of the hypothesis presented in the introductory chapter. Analysis of variance (ANOVA) is the first step in analyzing the factors that affect a given data set. The analyst uses the ANOVA in the f-test to generate additional data aligned with the proposed regression model. The ANOVA test allows comparison of more than two groups at the same time to determine if there is a relationship between the variables. The result of the ANOVA, F statistical (also called the F ratio), allows the analysis of multiple groups of data to determine variability between and within samples.

Fig. 17 presents the trending curve for each model. To observe the models, we create six curves for loss and accuracy. We labeled the curve with the x-axis, representing



```
VaT-SeTrans : andesit berwarna abuabu terang yang lava gunung lava
BLEU-1: 0.7722; BLEU-2: 0.6977; BLEU-3: 0.6061; BLEU-4: 0.5247

VaT-LSTM    : andesit berwarna abuabu terang dengan hornblende berwarna
              hitam dari lava gunung api
BLEU-1: 1.00; BLEU-2: 1.00; BLEU-3: 1.00; BLEU-4: 1.00

VGG16-Trans : andesit dari lava gunung api mengandung hornblende berwarna
              hitam dari lava gunung api
BLEU-1: 0.6923; BLEU-2: 0.7206; BLEU-3: 0.7735; BLEU-4: 0.7864

VGG16-LSTM  : andesit mengandung hornblende berwarna hitam dari lava
              gunung api
BLEU-1: 1.00; BLEU-2: 1.00; BLEU-3: 1.00; BLEU-4: 1.00

VGG16-LSTM-Luong    : andesit mengandung hornblende berwarna hitam dari
                      lava gunung api
BLEU-1: 1.00; BLEU-2: 1.00; BLEU-3: 1.00; BLEU-4: 1.00

VGG16-LSTM-Bahdanau : andesit berwarna abuabu terang yang mengandung
                      hornblende berwarna hitam dari lava gunung api
BLEU-1: 1.00; BLEU-2: 1.00; BLEU-3: 1.00; BLEU-4: 1.00

Ground-Truth :
andesit berwarna abuabu terang yang mengandung hornblende berwarna hitam dari lava gunung api
andesit mengandung hornblende berwarna hitam dari lava gunung api
andesit berwarna abuabu terang dengan hornblende berwarna hitam dari lava gunung api
andesit berwarna abuabu terang dari lava gunung api mengandung hornblende berwarna hitam
andesit dari lava gunung api mengandung hornblende berwarna hitam
```

FIGURE 14. Comparison of caption in Bahasa from various image captioning model.



```
VaT-SeTrans : batupasir dengan
BLEU-1: 0.1116; BLEU-2: 0.00; BLEU-3: 0.00; BLEU-4: 0.00

VaT-LSTM    : batugamping
BLEU-1: 0.0183; BLEU-2: 0.00 BLEU-3: 0.00; BLEU-4: 0.00

VGG16-Trans : intrusi batulanau abuabu napal dan serpih
BLEU-1: 0.1667; BLEU-2: 0.00; BLEU-3: 0.00; BLEU-4: 0.00

VGG16-LSTM  : batupasir berselingan dengan batulempung pasir
BLEU-1: 0.20; BLEU-2: 0.00; BLEU-3: 0.00; BLEU-4: 0.00

VGG16-LSTM-Luong : batupasir coklat
BLEU-1: 0.1116; BLEU-2: 0.00; BLEU-3: 0.00; BLEU-4: 0.00

VGG16-LSTM-Bahdanau : singkapan batupasir dan breksi atau lava butir
                      piroksen olivin plagioklas yang relatif hancur dan serpih
BLEU-1: 0.333; BLEU-2: 0.00; BLEU-3: 0.00; BLEU-4: 0.00

Ground-Truth :
singkapan batugamping klastik berukuran butir lempungan lanaulempung berwarna segar coklat tua
singkapan batugamping berwarna segar coklat tua
singkapan batugamping klastik dengan berukuran butir lempungan
singkapan batugamping klastik berukuran butiran lempungan dan berwarna segar coklat tua
batugamping berwarna segar coklat tua
```

FIGURE 15. Captions show overfitting in Bahasa.

an epoch, and the y-axis, representing a loss or an accuracy value. The legend presents two different colors: red and blue. We noticed that the red is an accuracy curve, and the blue is a loss on the validation dataset. The red-colored vertical line represents the threshold epoch boundary for each model. The models use an LSTM as a language generator and have a smooth decline when comparing the prediction and reference. This illustrates the formula that is referred to in equation (9). Equation (9) has consistently analyzed the differences between the predictions and the references.

The ANOVA test is a way to find out if the results of a survey or experiment are significant. In other words, it aids the research in determining whether to reject or accept the proposed hypothesis. ANOVA solely establishes the scientific validity of the proposed hypothesis through experimentation. The hypothesis suggests that alterations in the architecture of the captioning model will lead to variations in the caption outcomes.

In Table 6 displays the results of BLEU-4 in each captioning model. BLEU-4 samples were taken for only

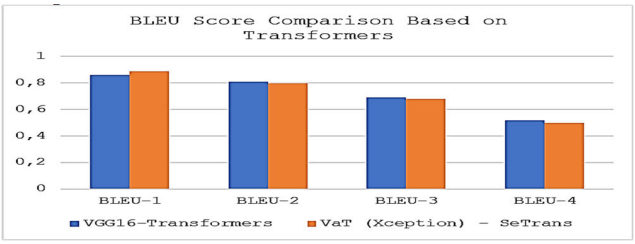


FIGURE 16. BLEU score comparison between VGG16 and VaT. Meanwhile, the language generator using transformers [7].

145 images The reason for taking the BLEU-4 is because captions are more visible with a 4-gram.

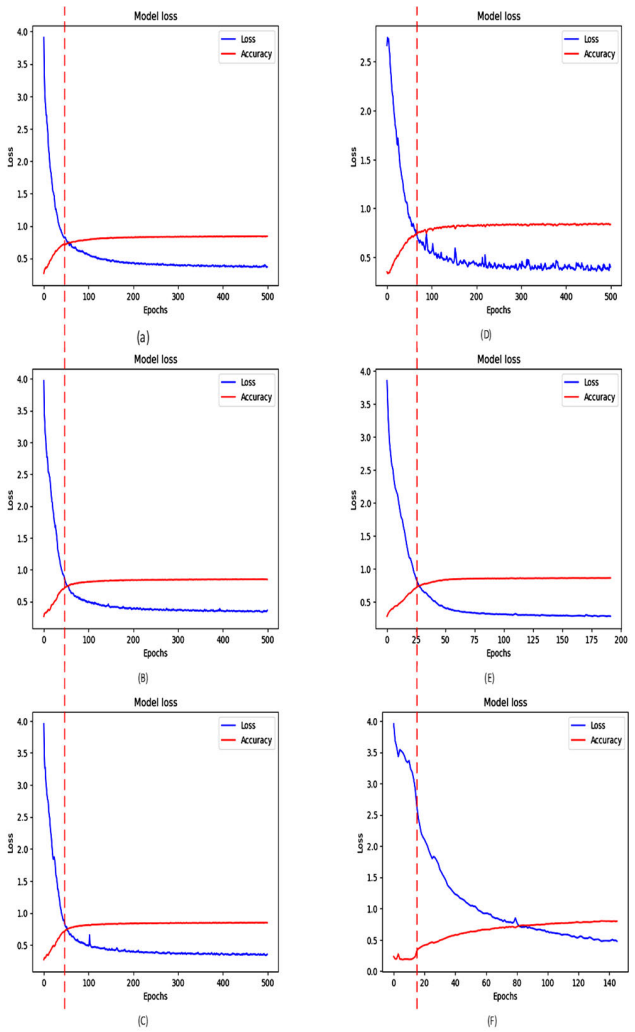


FIGURE 17. Curve trend comparison by epoch (a) VGG16-LSTM, (b) VGG16-LSTM-Bahdanau, (c) VGG16-LSTM-Luong, (d) VGG16-Transformers, (e) VaT-LSTM, (f) VaT-SeTrans.

In Fig. 18, we obtained the results of hypothesis measurement using ANOVA. The measurement results confirmed that the calculated P-value (P) was obtained at P-value = 0.28. When considering P-value values greater than 0.05 in statistical trials, the combined BLEU-4 cannot be used as a basis for statistical tests. In other calculations involving the F-test, it is established that the F-test is greater than the F-crit. F-test calculations for combined BLEU-4 values are not suitable for use as hypothesis tests. We can conclude from the ANOVA calculation that the hypothesis based on the four BLEU-4 values in each model is not suitable for hypothesis testing. Based on the results of the combined statistical test BLEU-4, we conducted an independent statistical test to test another hypothesis. Self-statistical testing exclusively utilizes one captioning model and depends on the values of BLEU-1, BLEU-2, BLEU-3, and BLEU-4.

TABLE 6. Bleu-4 from Vgg16, separable Cnn.

Image No	VGG16L	VGG16B	VaT-SeMatt	VaT-SeTrans
1	0,00	0,00	0,00	0,44
2	0,00	0,00	0,00	0,00
3	0,00	0,00	1,00	0,00
4	1,00	1,00	1,00	0,53
5	0,00	1,00	0,00	0,00
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…
141	0,00	0,00	0,00	0,00
142	0,00	0,00	0,00	0,00
143	1,00	1,00	1,00	0,00
144	0,00	0,00	0,00	0,00
145	0,00	0,00	0,00	0,00

Anova: Single Factor

SUMMARY

Groups	*Count*	*Sum*	*Average*	*Variance*
VGG16L	145,00	66,00	0,46	0,25
VGG16B	145,00	69,00	0,48	0,25
Xcep+LSTM	145,00	81,66	0,56	0,25
Xcep+trans	145,00	72,42	0,50	0,24

ANOVA

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	0,95	3,00	0,32	1,29	0,28	2,62
Within Groups	142,27	576,00	0,25			
Total	143,22	579,00				

FIGURE 18. ANOVA score for model VGG16L, VGG16B, VaT-SemAtt, VaT-SeTrans.

Meanwhile, for the statistical examination of the variable BLEU, we have switched to the VaT-SeTrans model. We will perform ANOVA measurements on the results of the statistical test using the VaT-SeTrans model selected for BLEU-4, as shown in Fig. 1 We utilize Table 7 for statistical evaluation. n. This measurement is intended to determine whether the VaT-SeTrans Model has an impact on caption success.

Fig. 19 displays statistical parameters that indicate the level of significance of the BLEU evaluation. The BLEU score was influenced by both the baseline model and the proposed models that treated the CNN layer and its parameters.

TABLE 7. BLEU-N scores for VaT-SeTrans.

Image	BLEU-1	BLEU-2	BLEU-3	BLEU-4
1	0,71	0,60	0,52	0,43
2	1,00	1,00	1,00	0,00
3	0,21	0,13	0,00	0,00
4	0,77	0,70	0,61	0,52
…	…	…	…	…
…	…	…	…	…
…	…	…	…	…
143	0,89	0,33	0,00	0,00
144	0,00	0,00	0,00	0,00
145	0,00	0,00	0,00	0,00

Anova: Single Factor

SUMMARY

Groups	*Count*	*Sum*	*Average*	*Variance*
BLEU-1	145	128,46	0,89	0,07
BLEU-2	145	115,55	0,80	0,13
BLEU-3	145	98,35	0,68	0,20
BLEU-4	145	72,42	0,50	0,24

ANOVA

Source of Variation	*SS*	*df*	*MS*	*F*	*P-value*	*F crit*
Between Groups	12,14	3,00	4,05	24,94	3,49E-15	2,62
Within Groups	93,47	576,00	0,16			
Total	105,61	579,00				

FIGURE 19. ANOVA for VaT-SeTrans models.

Fig. 19 writes P-Value $= 3.49 \times 10^{-15}$. It means the models have an impact on the shifting captions. The P-value gives a guarantee to the proposed hypothesis. In statistical terms, the thresholds for the P-value do not exceed 0.05. In Fig. 18, the ANOVA indicates that the P-value is less than the threshold of 0.05. The F measure is another important value. The F measure must exceed the F threshold. In Fig. 18, the F threshold, or F crit, is 2.6203, while the F measure is 24.94. The P-value and F-crit values provide a guarantee of significance for the hypothesis. We conclude that the proposed model has significant potential to generate the caption. The improved caption, compared to the baseline caption, validates the model's success in creating a better caption and causing the shift.

The conducted experiments yielded several observations about the proposed model. We observed that numerical values in the proposed word embedding method, word2vec, provide more support for vector operationalization calculations. The dimension size of 400×100 provides flexibility in translating each word, so that the combination of result values is more precise than each word translation.

The specified image inputs of 224×224 and 299×299 provide support for the separable CNN method, so that image extraction can provide the expected object detection [45]. The residual value present on the separable CNN helps to direct the results to the expected features. Determining the effective number of CNN layers for the geological imagery domain requires separate research. The text generation method, LSTM, supports word prediction for small word counts in dictionaries [46], [47]. LSTM is confirmed to be more suitable for geological imagery of rocks than the attention method. Adding imagery to the dataset is more likely to improve object detection accuracy. Variations in color and texture of rock imagery will provide support in terms of characterizing the specifications of a particular rock.

The loss calculation refers to (9) has shown how to use the log likelihood optimization technique for the parameters. The loss displayed indicates that there is a difference in optimization between the argmax of the parameters and the resulting model. This distinction is based on the empirical distribution defined by the training set and the probability distribution of the resulting model. The difference for each loss on each machine is not very significant.

V. CONCLUSION

Several findings from this study align with the objectives of the proposed model, including the implementation of a backbone with a separable CNN. The model has proven to produce better output than the baseline model. The incorporation of a residual value into the architecture of the separable CNN can enhance the accuracy of image extraction. The weight at the fully connected layer (FC) reinforces image object identification. The use of FC layers 2048 and 4096 is an important output in geological image extraction, as it strengthens the prediction of words that intersect with image features. LSTM word generators significantly outperform Transformers in producing captions. Word generation with LSTM provides sentences that approach geologists with BLEU score values reaching 40% and RougeL reaching above 50%. Transformers help to produce simpler captions than LSTM. We confirmed that the proposed image captioning model has a BLEU score above the model baseline. We can test the model's hypothesis by using BLEU-1, BLEU-2, BLEU-3, and BLEU-4 as statistical test variables. The ANOVA results refer to the values of F-Test $= 24.94$ and P-Value $= 3.49 \times 10^{-15}$. The hypothesis, which states that the CNN architectural model and text generator can provide captions close to references, holds true in the context of our domain problem. This test was measured by F-Test $>$ F-crit and P-value $<$ (P-test $= 0.05$). Using evaluation methods such as BLEU, RougeL, and METEOR can significantly enhance the validity of model proposals. The evaluation of captions using the BLEU and METEOR methods demonstrates their effectiveness in generating accurate and available captions. RougeL ensures the production of words with sufficient availability.

With exposure to the results and records obtained, further research can be conducted in the field of geological images. We can still utilize the most recent word embedding technique or adjust the word2vec method's number of dimensions. On the encoder side, it is still possible to find more efficient models for object detection, such as the effectiveness of the number of CNN layers, residual values, separable techniques, and other parameters. The CNN method can still be used with

YOLO, which can improve object detection precision. In a single geological image, the YOLO results can identify variations in rocks, allowing multiple rock names to appear. On the text generation side, there is still room for improvement in terms of geological grammatical arrangement to ensure it meets the standard. The resulting caption results can be formally recognized for word arrangement and grammar.

ACKNOWLEDGMENT

The authors would like to thank Dr. Joko Wahyudiono and Fitriani Agustin who help to collect and annotate the geological rock imagery and also would like to thank their willingness to freely donate their time in order to provide the image materials and geological rock annotations.

REFERENCES

[1] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 664–676, Apr. 2017, doi: 10.1109/TPAMI.2016.2598339.

[2] R. Li, H. Liang, Y. Shi, F. Feng, and X. Wang, "Dual-CNN: A convolutional language decoder for paragraph image captioning," *Neurocomputing*, vol. 396, pp. 92–101, Jul. 2020, doi: 10.1016/j.neucom.2020.02.041.

[3] V. Mullachery and V. Motwani, "Image captioning," 2018, *arXiv:1805.09137*.

[4] E. Tan and L. Sharma, "Neural image captioning," 2019, *arXiv:1907.02065*.

[5] M. Soh, "Learning CNN-LSTM architectures for image caption generation," Dept. Comput. Sci., Stanford Univ., Jane Stanford Way, Stanford, CA, USA, Tech. Rep., pp. 1–9, 2016. [Online]. Available: https://cs224d.stanford.edu/reports/msoh.pdf

[6] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," 2017, *arXiv:1703.06870*.

[7] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1440–1448.

[8] Y. Zhu, X. Li, X. Li, J. Sun, X. Song, and S. Jiang, "Joint learning of CNN and LSTM for image captioning," in *Proc. CEUR Workshop*, 2016, pp. 421–427.

[9] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 1–10.

[10] W. Ren, M. Zhang, S. Zhang, J. Qiao, and J. Huang, "Identifying rock thin section based on convolutional neural networks," *Proceedings 9th Int. Workshop Comput. Sci. Eng. (WCSE)*, vol. 52, 2020, pp. 345–351, doi: 10.18178/wcse.2019.06.052.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25, 2012, pp. 1–9, doi: 10.1201/9781420010749.

[12] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.

[13] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.

[14] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[15] H. Kaiming, Z. Xiangyu, R. Shaoqing, and S. Jian, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 1–9.

[16] Y. Bhatia, A. Bajpayee, D. Raghuvanshi, and H. Mittal, "Image captioning using Google's inception-ResNet-V2 and recurrent neural network," in *Proc. 12th Int. Conf. Contemp. Comput. (IC3)*, Noida, India, Aug. 2019, pp. 1–6.

[17] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–15.

[18] N. Li and Z. Chen, "Image cationing with visual-semantic LSTM," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Stockholm, Sweden, 2018, pp. 793–799. [Online]. Available: https://www.ijcai-18.org/

[19] M. T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2015, pp. 1412–1421, doi: 10.18653/v1/d15-1166.

[20] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017, *arXiv:1706.03762*.

[21] M. Junhua, X. Wei, Y. Yi, W. Jiang, H. Zhiheng, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (M-RNN)," in *Proc. ICLR*, 2015, pp. 1–17.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, Long Beach, CA, USA, 2017, pp. 1–11.

[23] H. Lee, S. Yoon, F. Dernoncourt, D. S. Kim, T. Bui, and K. Jung, "ViLBERTScore: Evaluating image caption using vision-and-language BERT," in *Proc. 1st Workshop Eval. Comparison NLP Syst. (Eval4NLP)*, 2020, pp. 34–39. [Online]. Available: https://aclanthology.org/2020.eval4nlp-1.4.pdf

[24] J. Li, P. Yao, L. Guo, and W. Zhang, "Boosted transformer for image captioning," *Appl. Sci.*, vol. 9, no. 16, p. 3260, Aug. 2019, doi: 10.3390/app9163260.

[25] S. He, W. Liao, H. R. Tavakoli, M. Yang, B. Rosenhahn, and N. Pugeault, "Image captioning through image transformer," 2020, *arXiv:2004.14231*.

[26] J. Su, J. Tang, Z. Lu, X. Han, and H. Zhang, "A neural image captioning model with caption-to-images semantic constructor," *Neurocomputing*, vol. 367, pp. 144–151, Nov. 2019, doi: 10.1016/j.neucom.2019.08.012.

[27] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3156–3164.

[28] P. Chun, T. Yamane, and Y. Maemura, "A deep learning-based image captioning method to automatically generate comprehensive explanations of bridge damage," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 37, no. 11, pp. 1387–1401, Sep. 2022, doi: 10.1111/mice.12793.

[29] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807. Accessed: Feb. 8, 2023. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2017/papers/Chollet_Xception_Deep_Learning_CVPR_2017_paper.pdf

[30] W. Joko, S. Aris, A. Ryandi, and S. Bisma, "Penelitian geologi dan geofisika untuk pengusulan wilayah kerja migas seram onshore," Geological Res. Center, Ministry Energy Mineral Resources, Bandung, Indonesia, Tech. Rep. 2017, 2017.

[31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[32] K. Papineni, S. Roukos, T. Ward, and Z. Wei-Jing, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Philadelphia, PA, USA, 2002, pp. 311–318. [Online]. Available: https://dl.acm.org/citation.cfm?id=1073135

[33] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proc. ACL Workshop Intrinsic Extrinsic Eval. Measures Mach. Transl. Summarization*, vol. 29, 2005, pp. 65–72.

[34] C.-Y. Lin. (2004). *ROUGE: A Package for Automatic Evaluation of Summaries*. Accessed: Jul. 2, 2023. [Online]. Available: https://aclanthology.org/W04-1013

[35] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998. [Online]. Available: http://ieeexplore.ieee.org/document/726791/#full-text-section

[36] X. Ran, L. Xue, Y. Zhang, Z. Liu, X. Sang, and J. He, "Rock classification from field image patches analyzed using a deep convolutional neural network," *Mathematics*, vol. 7, no. 8, p. 755, Aug. 2019, doi: 10.3390/math7080755.

[37] B. Liu, Y. Zhang, D. He, and Y. Li, "Identification of apple leaf diseases based on deep convolutional neural networks," *Symmetry*, vol. 10, no. 1, p. 11, Dec. 2017, doi: 10.3390/sym10010011.

[38] Y. Zhang, M. Li, S. Han, Q. Ren, and J. Shi, "Intelligent identification for rock-mineral microscopic images using ensemble machine learning algorithms," *Sensors*, vol. 19, no. 18, p. 3914, Sep. 2019, doi: 10.3390/s19183914.

[39] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016, doi: 10.1016/j.neucom.2015.09.116.
[40] Y. Boureau, J. Ponce, J. P. Fr, and Y. Lecun, "A theoretical analysis of feature pooling in visual recognition," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 111–118. [Online]. Available: https://www.di.ens.fr/sierra/pdfs/icml2010b.pdf
[41] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proc. 1st Int. Conf. Learn. Represent. (ICLR)*, 2013, pp. 1–12.
[42] R. C. Staudemeyer and E. Rothstein Morris, "Understanding LSTM—A tutorial into long short-term memory recurrent neural networks," 2019, *arXiv:1909.09586*.
[43] S. Weijie, Z. Xizhou, C. Yue, L. Bin, and L. Lewei, "VL-BERT: Pre-training of generic visual," in *Proc. ICLR*, 2020, pp. 1–16.
[44] I. Shivhare, J. Purohit, V. Jogani, P. M. Chawan, and B. Tech Student, "Image captioning using transformer: Visionaid," *Int. Res. J. Eng. Tech.*, vol. 9, no. 10, pp. 567–575, 2022. [Online]. Available: https://www.irjet.net
[45] Z. Momynkulov, Z. Dosbayev, A. Suliman, B. Abduraimova, N. Smailov, M. Zhekambayeva, and D. Zhamangarin, "Fast detection and classification of dangerous urban sounds using deep learning," *Comput., Mater. Continua*, vol. 75, no. 1, pp. 2191–2208, 2023, doi: 10.32604/cmc.2023.036205.
[46] B. Omarov, O. Auelbekov, A. Suliman, and A. Zhaxanova, "CNN-BiLSTM hybrid model for network anomaly detection in Internet of Things," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 3, pp. 436–444, 2023, doi: 10.14569/IJACSA.2023.0140349.
[47] A. Mukhametkaly, Z. Momynkulov, N. Kurmanbekkyzy, and B. Omarov, "Deep Conv-LSTM network for arrhythmia detection using ECG data," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 9, pp. 698–707, 2023, doi: 10.14569/IJACSA.2023.0140973.

AGUS NURSIKUWAGUS (Member, IEEE) was born in Jakarta, Indonesia, in 1975. He received the master's degree from Bandung Institute of Technology, in 2005, and the Ph.D. degree in Indonesia, in 2023. He has a major field in computer vision and natural language processing. He is currently a Senior Lecturer with Indonesia Computer University (UNIKOM). He has published numerous articles in the Scopus index and WoS. One of the topics, he has published is related to data mining and fuzzy systems. He has authored over 25 articles and has invented five software applications. He has been a member of IAENG, since 2019; and has joined the informatics associations APTIKOM and IAII. His research interests include artificial intelligence, machine learning, and fuzzy systems. He has been receiving scholarships and grants from the Ministry of Culture and Education Indonesia.

RINALDI MUNIR (Member, IEEE) was born in Padang, Indonesia, in 1965. He received the B.S. and M.S. degrees, in 1992 and 1999, respectively, and the Doctor degree in informatics engineering from Bandung Institute of Technology, in 2010.

He has been an Associate Professor. Since 1994, he has been lecturing with the Informatics Department, Bandung Institute of Technology. He has written two books, algorithms, and data structures and discrete mathematics and more than 30 articles. His research interests include encryptions and computer vision. He received a research grant from the Ministry of Culture and Education and was recognized as the Best Teaching Lecturer in the School of Electrical Engineering and Informatics.

MASAYU L. KHODRA received the graduate degree in informatics engineering from Bandung Institute of Technology, Indonesia, in 2004, the M.S. degree, and the Doctor degree in informatics engineering from Bandung Institute of Technology, Indonesia, in 2006.

Since 2008, she has been an Associate Professor with Bandung Institute of Technology. Her research interests include the natural language processing and artificial intelligence.

DESHINTA ARROVA DEWI received the Ph.D. degree in computer science from the National University of Malaysia (UKM), Malaysia, in 2019. She joined INTI International University, in 2010, where she is actively conducting teaching, learning, and research. Her research interests include artificial intelligence, software engineering, data science, and education. She was appointed as an Associate Professor, in 2023, and a Managing Editor of the *Journal of Data Science*.