JOIV : Int. J. Inform. Visualization, 7(2) - June 2023 487-493

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Pre-Trained CNN Architecture Analysis for Transformer-Based Indonesian Image Caption Generation Model

Rifqi Mulyawan [a], Andi Sunyoto [a,*], Alva Hendi Muhammad [a]

[a] *Post Graduate Program, Universitas Amikom Yogyakarta, Daerah Istimewa Yogyakarta 55283, Indonesia*

Corresponding author: [*]andi@amikom.ac.id

***Abstract*— Classification and object recognition in image processing has significantly improved computer vision tasks. The method is often used for visual problems, especially in picture classification utilizing the Convolutional Neural Network (CNN). In the popular state-of-the-art (SOTA) task of generating a caption on an image, the implementation is often used for feature extraction of an image as an encoder. Instead of performing direct classification, these extracted features are sent from the encoder to the decoder section to generate the sequence. So, some CNN layers related to the classification task are not required. This study aims to determine which CNN pre-trained architecture or model performs best in extracting image features using a state-of-the-art Transformer model as its decoder. Unlike the original Transformer's architecture, we implemented a vector-to-sequence way instead of sequence-to-sequence for the model. Indonesian Flickr8k and Flick30k datasets were used in this research. Evaluations were carried out using several pre-trained architectures, including ResNet18, ResNet34, ResNet50, ResNet101, VGG16, Efficientnet_b0, Efficientnet_b1, and Googlenet. The qualitative model inference results and quantitative evaluation scores were analyzed in this study. The test results show that the ResNet50 architecture can produce stable sequence generation with the highest accuracy value. With some experimentation, finetuning the encoder can significantly increase the model evaluation score. As for future work, further exploration with larger datasets like Flickr30k, MS COCO 14, MS COCO 17, and other image captioning datasets in Indonesian also implementing a new Transformers-based method can be used to get a better Indonesian automatic image captioning model.**

***Keywords*—Artificial intelligence; deep learning; convolutional neural network; indonesian image caption generation; transformer.**

Manuscript received 12 Nov. 2022; revised 31 Dec. 2022; accepted 8 Jan. 2023. Date of publication 30 Jun. 2023.

I. INTRODUCTION

Many currently available captioning algorithms to convey in words an essence of an image are based on the architecture of an encoder-decoder, in which a decoder infrastructure may anticipate words by using a function received from an encoder network through an attention approach. Studies on image subtitling have mainly concentrated on a translation approach consisting of a visual encoder and a language decoder [1].

Creating image captions may be utilized for various purposes, including automating the driving of autos, developing face recognition systems, characterizing individuals with visual impairments, enhancing the quality of photo queries, and many more. The difficult task of developing the natural language descriptions of the information in a picture resides within the computer vision (CV) interface for image feature extraction and generating the sequence using the natural language processing (NLP) technology.

The task of photo caption generation has already had a significant impact in several fields, such as image search also various disciplines, such as software development for people with disabilities, video surveillance and security, and the interface between humans and computers [2].

As a popular challenge involving sequence modeling, the state-of-the-art (SOTA) problem of photo caption generation uses various approaches. For example, the Convolutional Neural Network, ConvNet, known as the CNN, is applied with other language architecture, like the Recurrent Neural Network (RNN), as a CNN-RNN-based framework approach [3]. This work uses the standard encoder-decoder architecture using a pre-trained CNN model to build feature vectors, and they are then fed into an RNN as the decoder generates the language description.

The standard encoder-decoder model was also utilized to make subtitles from photographs [4], [5]. However, the recurrent structure of the enhanced RNN type, like the Long Short Term Memory (LSTM), makes it harder to train because

of its sequential nature, resulting in a lower evaluation score on the standard RNN-based model. However, the parallelism problem was finally overcome by the SOTA model, it is the Transformer [6]. Since the architecture is built on a context-aware attention mechanism, it can operate parallel throughout the training phase and does not require a certain order.

For image captioning in Indonesian, the GRU approach [7] generates Indonesian captions to overcome some problems in the RNN. However, as the model is still RNN-based, their finding shows that it lacks context understanding and stated that the need for SOTA research implementation for sequence generation in Indonesian is a must. Earlier research for Indonesian caption generation [8] also uses CNN with the pre-trained architecture of VGG-16 for the model's encoder with another RNN type, the LSTM, as the decoder, but without investigating feature extraction impact on measuring image text quality for the model's performance. Their finding shows that the model's result has a better evaluation score with BLEU 1-4 (50.00, 31.40, 23.90, 13.10, respectively). Previous studies have studied generating image captions in Indonesian leaves a space for exploring the effect of using another pre-trained CNN layer with the SOTA approach employing Transformer-based that is context-aware to get better model evaluation results [7], [8].

Our contributions to this research are as follows:

- Create an Indonesian image captioning dataset based on the rules of the standard benchmark of Flickr8k and Flickr30k to train the model.
- Propose a Transformer-based model using CNN as the encoder to generate photo captions in Indonesian.
- Employ a context-aware using an attention-mechanism-based decoder.
- Compare eight different pre-trained CNN as photo feature extraction to the Transformer-based model.
- Compare the model performance to the previous approach in Indonesian image captioning.

In this study, we used Indonesian Flickr8k [9] and translated Flickr30k [10] to test our model's performance in the Indonesian language to produce an image captioning model in Indonesian, proposing SOTA Transformer-based architecture. Fig. 2 depicts the proposed Transformer-based model's approach to caption images in Indonesian.

This research explores which CNN architecture is the most effective at generating high-accuracy results by comparing and contrasting their respective performances on eight different pre-trained CNNs. This study also investigates the effect of varying CNN channel size (depth) on the Transformer-based model performance for image feature extraction**.**

A. Image Caption Generation in Another Language

Since most datasets are written in English, most of the study for caption generation was done in that language, whereas the attention-based mechanism is adapted for caption generation [11]. Most studies implement the VGG-16 for the encoder part of the captioning model, like the ConvNet [12]. However, several researchers also employed the pre-trained AlexNet [13], [4], or Residual Network (ResNet) for the visual feature and BiLSTM [13].

For other languages, other datasets like Chinese [14], [15], Japanese Yoshikawa [16], Arabic [17], Bahasa Indonesia in [18] (custom dataset that combines MS COCO and Flickr30k), Indonesian Flickr30k [7], and the FEEH-ID Flickr8k's dataset [8] also created besides English.

B. Image Caption Generation using Attention-Mechanism

A significant number of researchers in the past have made use of visual attention to English datasets. Encoder-decoder research has used two primary kinds of attention, namely for the purpose of captioning images or videos. The first sort of attention is called semantic attention, which refers to attention to words. The second one of attention is known as spatial attention, which relates to the focus placed on images. Research by Xu et al. [19] on photo captioning saw the introduction of a model for visual attention for the first time. They either applied "hard" pooling, which finds the region that is more likely to be attended, or "soft" pooling, which takes the average of the spatial qualities and assigns attentive weights to each variable.

Moreover, CNN's Channel-wise Attention and Spatial Attention were used when watching the network [20]. Chen et al. [21] also used visual attention when creating captions for the pictures. Also, a semantic attention model was used in RNNs to link the visual feature with the visual ideas to create the picture description [22].

C. Image Captioning using Transformer-Based Approach

Image captioning with Transformer as the model's decoder using an English dataset was used in previous research. Li et al. [23] studied a Transformer-based framework for sequence modeling in picture captioning. When it was initially developed, it included simply the attention and feed-forward layers.

In addition, the study presented by Herdade et al. [24] makes use of spatial object relationship modeling for picture caption generation. It is explicitly done inside the encoder-decoder architecture using the SOTA Transformer. It is done by implementing the object relation module to the encoder as the first step in developing image captions. Research in Atliha and Šešok [25] suggested that augmenting the photo captions in a dataset with additional information, such as employing BERT, might be an effective method for enhancing an existing solution to the problem of image captioning.

Research by Zhu et al. [26] used two different streams of architecture based on Transformers-one for the graphical component and another for the linguistic component. Zhu et al. [26] additionally utilized a CNN model for the encoding component, while a Transformer model was used for the decoding section of the model. Both the encoder and decoder models were utilized. The architecture was constructed using a Transformer, which consists of a model for both an encoder and a decoder. In addition, it employs a system for stacking its attention on top of itself. When CNN is employed as an encoder, as explored in Zhang et al. [27], image features may be obtained, and the encoder's output is a context vector containing the most significant picture information. After that, this vector is sent into Transformer, which creates the captions for the pictures based on those captions.

To put it into perspective, research by He et al. [28] presented the image Transformer as a tool for image captioning. Each layer of the Transformer implements several sub-Transformers that enable the encoding of spatial

relationships between picture portions and the decoding of the different forms of information within the image regions.

II. Materials and Method

A. Dataset

The dataset used for this analysis is the standard English Flickr8K [28]. We translated it to Indonesian using Google Translate and manually cross-checked the annotation. Named Flickr8k Bahasa [9], like the original Flickr8k, our dataset features 8,091 photos. There are 6,000 training photos, 1,000 validations, and 1,000 for testing.

In addition, five human-created reference captions are linked to each image, meaning that for every image in our training set, there are 40,460 corresponding caption samples. We also prepared Indonesian Flickr30k's Bahasa, comprising 158,915 captions to test our final model performance. This translated dataset contains 31,783 photos, including a caption file comprising five types of sentences, 29,000 used for training, 1,000 used for testing, also validations.

B. System Design

Fig. 1, which can be seen further down this page, is a flow or process that describes in detail the experiments carried out to determine how the different ConvNet or CNN's pre-trained model methods perform in generating and evaluating image caption problems in Indonesian. It provides an easy-to-follow visual representation of the entire procedure. The first step is to preprocess the caption text and the input image. The caption text from the dataset is tokenized to ensure we have a unique vocabulary. At this stage, each image in our Indonesian dataset changed to less than the original size. Then the dataset was prepared for training, validation, and testing, resulting in the input data for the training process using the CNN method with transfer learning techniques.

Eight different CNN pre-trained architectures are used at this stage, namely ResNet18, ResNet34, ResNet50, ResNet101, VGG16, Efficientnet_b0, Efficientnet_b1, and Googlenet. Another system output is the prediction or the inferences of the CNN-Transformer model.

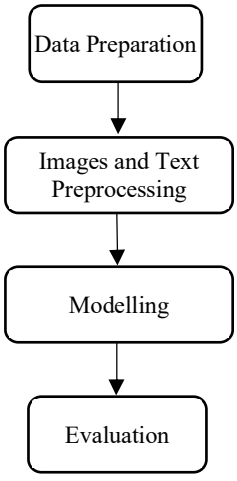

Fig. 1 Indonesian Image Captioning Model Analysis Flowchart

C. CNN-Transformer

The ResNet CNN model was utilized as our choice for the encoding algorithm baseline. Vectors of fixed-length feature representation that CNN extracts are called encoder's hidden states, which are then used as the basis for the attention mechanism alongside the annotation vectors. Various networks, including ResNet18, ResNet34, ResNet50, ResNet101, VGG16, Efficientnet b0, Efficientnet b1, and Googlenet, were used in our tests. Since we are not interested in classifying the input, the last pooling and softmax layer are unneeded and retrieved annotation vectors from the last convolutional layer instead. Here, the output is of the size that can be expressed with $x * y,\ n$, where n is the CNN feature channels that vary with the particular encoder employed and x, y represents the shape of the feature map.

Afterward, n number of decoder layers was applied to the summed-up output. Each decoder layer comprised three further layers:

- A sub-layer of masked multi-head attention that includes both a padding mask and a look-ahead mask.
- An attention sub-layer with many heads with a padding mask that accepts the encoder output as inputs (with two inputs).
- A masked multi-head attention sub-layer that has an output query.

Look-ahead and the padding mask of the Transformer were multi-head attention sub-layers that were disguised. Within this specific architectural design context, the third layer was made up of feed-forward networks. Then, the information that the Transformer decoder produced was sent to the linear layer so that it could be utilized as input there. In the end, probabilistic SoftMax predictions are constructed in a serial way, and the output generated up to this point is employed to determine the subsequent step that must be done to complete the process. Fig. 2, which can be seen below, is the image for our proposed architecture.

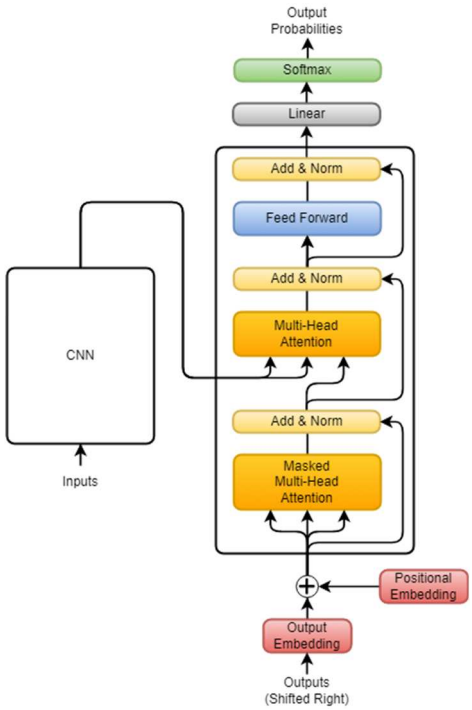

Fig. 2 CNN and Transformer-Based Decoder Model Architecture

Unlike RNN, where we send the words of a sentence one by one into the model, we send the whole sentence to the decoder simultaneously. This parallelization is the main benefit of why the architecture is faster to train compared to the previous one, like RNN/LSTM and GRU.

D. Model Evaluation Metrics

When assessing the quality of automatically generated captions, we make use of BLEU [29], METEOR [30], ROUGE [31], and CIDEr [32]. Utilizing n-grams, BLEU [29] determines the degree of similarity between a collection of reference texts and the text created by a computer. The word-to-word matching algorithm METEOR [30] uses equivalent word stems and synonyms to find straight matches between words. ROUGE [31] measures sentence similarity using word pairings, n-grams, and word sequences, whereas extant research on picture captioning makes considerable use of different metrics like BLEU, METEOR, and ROUGE. In addition, CIDEr [32] is also utilized to quantify the similarity between reference texts and predicted text for every n-gram. On the other hand, it has been discovered that CIDEr has a stronger correlation with human evaluation [33]. As a result, we concluded that including CIDEr would provide a more accurate depiction of the caption quality.

III. Results and Discussion

Three Transformer layers using A ResNet50 model as the encoder was our basic configuration for the ResNet-Transformer architecture, where one head is used for each SOTA Transformer layer. Here, we carried out some experiments: one in which we varied the encoder pre-trained model type; another in which we used the inference. Fig. 3 shows the qualitative model inference comparison, where the ResNet50 generates the Indonesian caption with a stably generated prediction (translated caption can be seen below each generated caption) and the detail of the experiment's quantitative test results in Table I.

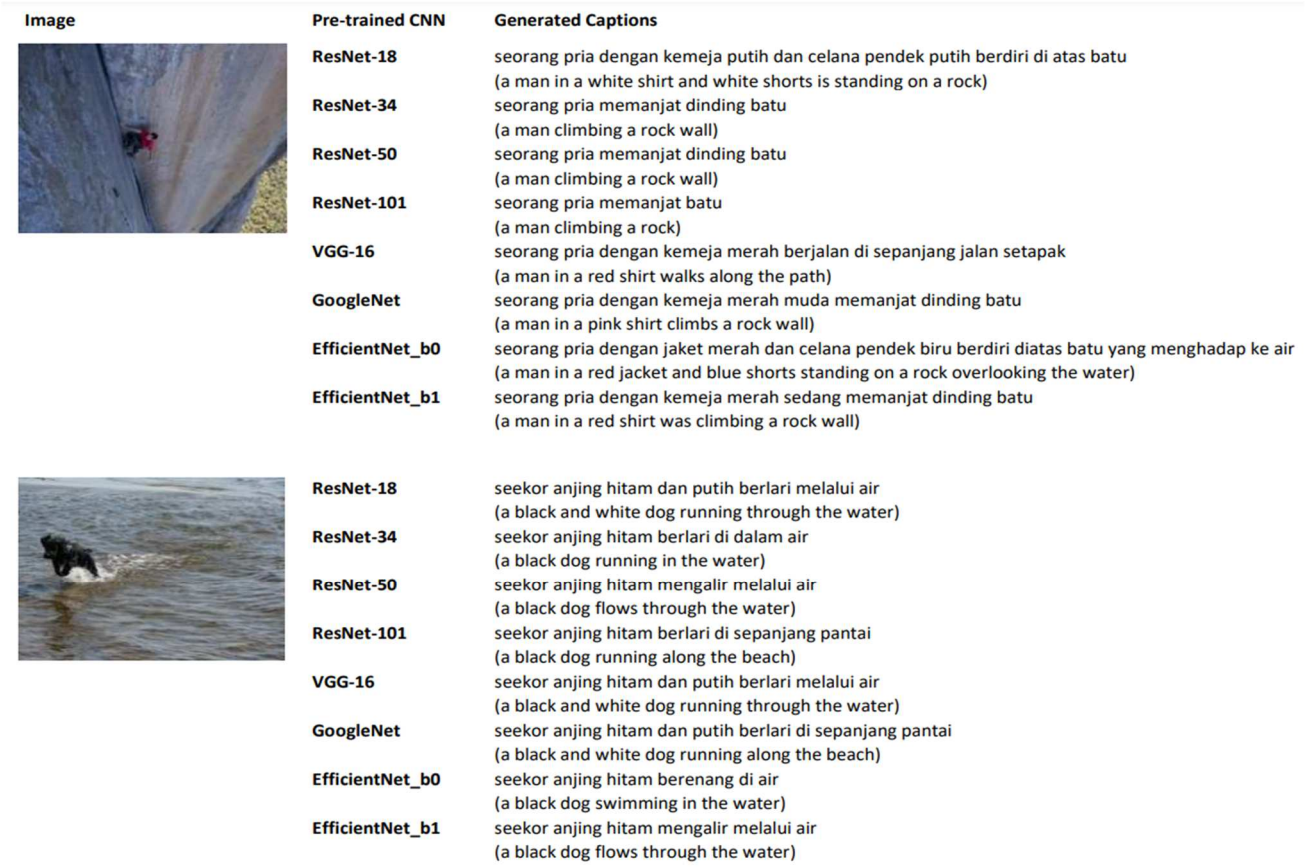

Fig. 3 CNN-Transformer Model Inference Qualitative Results Comparison With Different Pre-trained CNN Architecture

TABLE I
PRE-TRAINED CNN MODEL RESULTS

Encoder	BLEU 1-4				METEOR	ROUGE_L	CIDER
ResNet18	54.21	39.40	28.26	19.98	19.68	43.39	54.78
ResNet34	55.96	41.24	29.41	20.85	20.24	45.05	59.09
ResNet50	**56.57**	**42.16**	**30.57**	**21.82**	20.32	45.21	60.72
ResNet101	55.55	40.45	28.69	20.16	**21.11**	**45.81**	**62.63**
VGG16	55.68	40.61	28.77	20.19	20.29	44.57	59.40
GoogleNet	52.18	37.54	26.69	18.90	18.79	41.89	52.46
EfficientNetb0	52.98	37.55	26.15	18.31	19.09	42.82	53.52
EfficientNetb1	52.80	37.76	26.31	18.20	19.36	43.05	55.06

On the graphics processing unit (GPU) of a Google Colab Pro, each experiment was trained at a constant learning rate of 0.00004 using the Adam optimizer. It is done within fifty epochs and stopped if there has been no improvement in BLEU-4 throughout the most recent 10 epochs (the halting training criteria), where the overall training process is done in 5-12 hours on each pre-trained CNN architecture. Python with PyTorch's library is the performance analysis environment for each CNN model that includes three phases: (1) Training phase. (2) Validation. (3) Testing. In other words, we implement the parallelization to it, as the Transformer's architecture supports the simultaneous process.

As seen in Fig. 2, we changed the model's encoder part of the Transformer with a CNN. Instead of modeling sequence-to-sequence, like in the original Transformers, the modeling is done in a vector-to-sequence way. The input is the image we send into the CNN as the backbone. A Transformer decoder can handle the sequences generation part, which can generate the next word of a sentence. The decoder accepts these input features that extract input images from the CNN backbone as the visual backbone, where they predict the caption generation token by token. The generated captions are formulated as $Caps = (C_0, C_1, C_2, C_3, \dots, C_{token}, C_{token+1})$. The first generated caption $C_0 = < SOS >$ where the "SOS" stands for the start of a sentence, and the $C_{token+1} = < EOS >$ where "EOS" is the unique token meant as the last of the sentence. In short, this model architecture has two different sources of input: (1) The image we want to caption. (2) The very sentence we want it to generate but shifted one word to the left.

To begin, we use trained tokens and positional embeddings to transform the tokens that make up the caption into vectors. After that, we perform the vector's element-wise sum, layer normalization, and drop out. Next, these vectors are processed into a series of transformation layers. As seen in the proposed model architecture, the model uses the decoder component from the original Transformer. In addition to conducting masked multi-head self-attention on the token vectors, image vectors in each layer implement a two-layer fully connected network for every vector in turn.

The third step, layer normalization, comes after these three operations and is preceded by a dropout wrapped in a residual connection. Through their attention, token vectors interact with one another token. The masking that occurs throughout this procedure keeps the final predictions' causal structure intact. After applying the last Transformer layer, the unnormalized log probabilities throughout the token vocabulary are predicted by applying a linear layer to each vector that occurs after the application of the end of the Transformer layer. The pre-trained ResNet50 network, after the last convolutional layer, takes an image with dimensions of 224 by 224. It generates a 7 by 7 grid of features with a total of 2048 dimensions.

Because of the unique nature of the pre-training architecture, the CNN channel must be changed to each different model. 512 CNN channel for ResNet18 and ResNet34, 1024 for GoogleNet, 1280 for Efficientnet, 2048 for ResNet50, and ResNet101. The learning rate and epoch values implemented during the training phase were also consistent throughout the experiments.

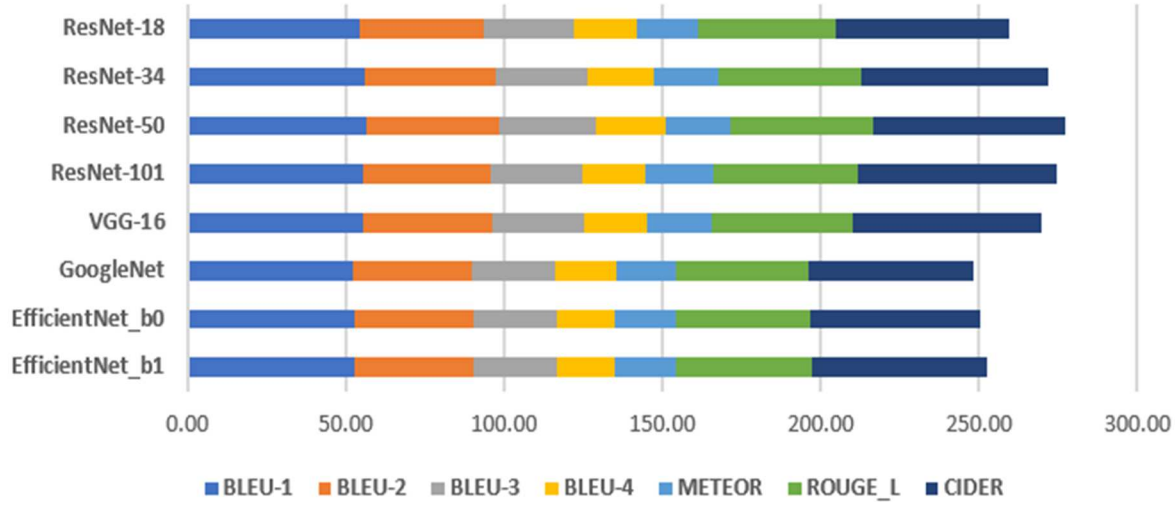

Fig. 4 Pre-trained CNN Architecture Results Comparison

Using the eight's different pre-trained CNN architecture in Fig. 4 shows that the ResNet50-based also has the best overall evaluation result. As shown in Fig. 4 above, the difference in CNN's channel size or depth affects the prediction results produced. The larger the size, the higher the accuracy value obtained. Based on the visualization of the test results, this increased accuracy value applies to all tested CNN models except for the ResNet101 CNN pre-trained model type with a 2048 channel size. We expect this to occur because our small Flickr8k's Bahasa dataset is underfitting.

We also examined the effect of finetuning the encoder and the model's performance after finetuning. It is accomplished by prohibiting gradient computation for the encoder's second blocks through the fourth convolutional as if we used zero learning rate for these parts. The validation results can be seen in Table II below.

TABLE II
FINETUNED PRE-TRAINED CNN MODEL RESULTS

Encoder	BLEU 1-4				METEOR	ROUGE_L	CIDER
ResNet18	52.91	37.53	25.70	17.39	18.79	42.31	49.85
ResNet34	55.73	40.38	28.41	19.57	19.29	43.53	53.17
ResNet50	**58.10**	**42.91**	**30.40**	21.13	20.12	45.32	60.80
ResNet101	56.24	41.46	29.44	20.28	**20.45**	**45.88**	**61.15**
VGG16	53.70	38.96	26.86	18.00	18.88	42.83	50.97
GoogleNet	52.39	37.12	25.54	17.20	18.49	41.16	48.43
EfficientNetb0	57.73	42.54	30.33	21.10	20.62	45.73	62.57
EfficientNetb1	56.84	42.07	30.24	**21.24**	20.40	45.59	61.09

TABLE III
STATE-OF-THE-ART RESULTS COMPARISON ON INDONESIAN IMAGE CAPTIONING

Model	Dataset	BLEU 1-4				METEOR	ROUGE_L	CIDER
CNN + GRU [7]	FLICKR30K INDONESIAN	36.70	17.80	06.70	02.00	-	-	-
CNN + LSTM [8]	FLICKR8K FEEH-ID	50.00	31.40	23.90	13.10	-	-	-
CNN + LSTM with Adaptive Attention [18]	MS COCO + FLICKR30K	67.80	51.20	37.50	27.40	-	-	99.00
Ours (CNN + Transformer)	FLICKR8K BAHASA	58.10	42.91	30.40	21.13	20.12	45.32	60.80
Ours (CNN + Transformer)	**FLICKR30K BAHASA**	**75.34**	**62.84**	**50.58**	**40.04**	**27.52**	**58.52**	**110.28**

The finetuned model's results in Table II effectively increase the overall model's result score evaluation except for ResNet18 as it seems other parameters like learning rate or Transformer's layer for the ResNet18-based model need to be readjusted.With some experimentation, we tested our Transformer-based finetuned model with the larger Flickr30k Bahasa dataset that has been prepared for experimental work. As we expected, the validation results were outstanding, as the Transformer-based model works better with larger training data. Based on the results, we can now compare the model with other previous approaches in Indonesian image captioning. Here, Transformer's context-aware attention mechanism as the model's decoder proved to be better than the previous types that used an RNN-type approach like GRU or LSTM as the model's decoder resulting better evaluation score, as shown in Table III.

IV. CONCLUSION

This study uses several CNN models, namely ResNet18, ResNet34, ResNet50, ResNet101, VGG16, Efficientnet_b0, Efficientnet_b1, and Googlenet, to obtain a CNN model that can produce the best performance as a feature extractor for predicting text sequences performed by the Transformer decoder.

The test is carried out using different sizes of the CNN Channel, where the best model was acquired using ResNet50 and proved that the model could generate grammatically correct Indonesian captions. Experiments indicate that finetuning the encoder model nearly always enhances the decoder model's output, producing a better evaluation score of about 2% than other CNN models.

The ResNet50 model is recommended for using CNN-based systems as the backbone and Transformer as the decoder, where the quantitative results are slightly better than earlier caption generation approaches using the Indonesian dataset. A sensibility analysis on a variety of CNN pre-trained architectures and implementing finetuning to the encoder improve the output of the Transformer-based decoder model for every different pre-trained encoder architecture with BLEU 1-4, METEOR, ROUGE_L, CIDEr of 58.10, 42.91, 30.40, 21.13, 20.12, 45.32, 60.80 respectively for Flickr8k Bahasa and BLEU 1-4, METEOR, ROUGE_L, CIDEr of 75.34, 62.84, 50.58, 40.04, 27.52, 58.52, 110.28 for the final validated model on Flickr30k Bahasa dataset.

As for future work, as our computational resources are platform-limited, further exploration of larger datasets such as Flickr30k, MS COCO 14, MS COCO 17, and other datasets related to image captioning undoubtedly improves the model's performance. Hopefully, as this finding only focuses on the encoder part of the model, it would be fascinating to test the impact of employing pre-trained word embeddings for the decoder part, mainly in Indonesian, as well as a more complex Transformers-based model.

REFERENCES

[1] R. Subash, R. Jebakumar, Y. Kamdar, and N. Bhatt, "Automatic image captioning using convolution neural networks and LSTM," *J. Phys. Conf. Ser.*, vol. 1362, no. 1, 2019, doi: 10.1088/1742-6596/1362/1/012096.

[2] M. D. Zakir Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, "A comprehensive survey of deep learning for image captioning," *ACM Comput. Surv.*, vol. 51, no. 6, 2019, doi: 10.1145/3295748.

[3] K. C. Nithya and V. V. Kumar, "A Review on Automatic Image Captioning Techniques," *Proc. 2020 IEEE Int. Conf. Commun. Signal Process. ICCSP 2020*, pp. 432–437, 2020, doi: 10.1109/ICCSP48568.2020.9182105.

[4] C. Wang, H. Yang, and C. Meinel, "Image Captioning with Deep Bidirectional LSTMs and Multi-Task Learning," *ACM Trans. Multimed. Comput. Commun. Appl.*, vol. 14, no. 2s, 2018, doi: 10.1145/3115432.

[5] C. Amritkar and V. Jabade, "Image Caption Generation Using Deep Learning Technique," *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2018*, pp. 1–4, 2018, doi: 10.1109/ICCUBEA.2018.8697360.

[6] A. Vaswani *et al.*, "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 2017-Decem, no. Nips, pp. 5999–6009, 2017.

[7] A. A. Nugraha, A. Arifianto, and Suyanto, "Generating image description on Indonesian language using convolutional neural network and gated recurrent unit," *2019 7th Int. Conf. Inf. Commun. Technol. ICoICT 2019*, pp. 1–6, 2019, doi: 10.1109/ICoICT.2019.8835370.

[8] E. Mulyanto, E. I. Setiawan, E. M. Yuniarno, and M. H. Purnomo, "Automatic Indonesian Image Caption Generation using CNN-LSTM Model and FEEH-ID Dataset," *2019 IEEE Int. Conf. Comput. Intell. Virtual Environ. Meas. Syst. Appl. CIVEMSA 2019 - Proc.*, 2019, doi: 10.1109/CIVEMSA45640.2019.9071632.

[9] R. Mulyawan, A. Sunyoto, and A. H. Muhammad, "Automatic Indonesian Image Captioning using CNN and Transformer-Based Model Approach," in *2022 5th International Conference on*

Information and Communications Technology (ICOIACT), 2022, pp. 355–360, doi: 10.1109/ICOIACT55506.2022.9971855.

[10] B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2015 Inter, pp. 2641–2649, 2015, doi: 10.1109/ICCV.2015.303.

[11] J. Aneja, A. Deshpande, and A. G. Schwing, "Convolutional Image Captioning," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 5561–5570, 2018, doi: 10.1109/CVPR.2018.00583.

[12] S. Liu, L. Bai, Y. Hu, and H. Wang, "Image Captioning Based on Deep Neural Networks," *MATEC Web Conf.*, vol. 232, pp. 1–7, 2018, doi: 10.1051/matecconf/201823201052.

[13] H. Shi, P. Li, B. Wang, and Z. Wang, "Image captioning based on deep reinforcement learning," *ACM Int. Conf. Proceeding Ser.*, vol. 01052, pp. 1–7, 2018, doi: 10.1145/3240876.3240900.

[14] W. Lan, X. Li, and J. Dong, "Fluency-guided cross-lingual image captioning," *MM 2017 - Proc. 2017 ACM Multimed. Conf.*, pp. 1549–1557, 2017, doi: 10.1145/3123266.3123366.

[15] X. Li, W. Lan, J. Dong, and H. Liu, "Adding Chinese captions to images," *ICMR 2016 - Proc. 2016 ACM Int. Conf. Multimed. Retr.*, pp. 271–275, 2016, doi: 10.1145/2911996.2912049.

[16] Y. Yoshikawa, Y. Shigeto, and A. Takeuchi, "STAIR captions: Constructing a large-scale Japanese image caption dataset," *ACL 2017 - 55th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf. (Long Pap.*, vol. 2, pp. 417–421, 2017, doi: 10.18653/v1/P17-2066.

[17] H. A. Al-muzaini, T. N. Al-yahya, and H. Benhidour, "Automatic Arabic image captioning using RNN-LSTM-based language model and CNN," *Int. J. Adv. Comput. Sci. Appl.*, vol. 9, no. 6, pp. 67–73, 2018, doi: 10.14569/IJACSA.2018.090610.

[18] M. R. S. Mahadi, A. Arifianto, and K. N. Ramadhani, "Adaptive Attention Generation for Indonesian Image Captioning," *2020 8th Int. Conf. Inf. Commun. Technol. ICoICT 2020*, 2020, doi: 10.1109/ICoICT49345.2020.9166244.

[19] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," *32nd Int. Conf. Mach. Learn. ICML 2015*, vol. 3, pp. 2048–2057, 2015.

[20] L. Chen *et al.*, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 6298–6306, 2017, doi: 10.1109/CVPR.2017.667.

[21] H. Chen, G. Ding, Z. Lin, S. Zhao, and J. Han, "Show, observe and tell: Attribute-driven attention model for image captioning," *IJCAI Int. Jt. Conf. Artif. Intell.*, vol. 2018-July, pp. 606–612, 2018, doi: 10.24963/ijcai.2018/84.

[22] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 4651–4659, 2016, doi: 10.1109/CVPR.2016.503.

[23] G. Li, L. Zhu, P. Liu, and Y. Yang, "Entangled transformer for image captioning," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2019-Octob, no. c, pp. 8927–8936, 2019, doi: 10.1109/ICCV.2019.00902.

[24] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, "Image captioning: Transforming objects into words," *Adv. Neural Inf. Process. Syst.*, vol. 32, pp. 1–11, 2019.

[25] V. Atliha and D. Šešok, "Text augmentation using BERT for image captioning," *Appl. Sci.*, vol. 10, no. 17, 2020, doi: 10.3390/app10175978.

[26] X. Zhu, L. Li, J. Liu, H. Peng, and X. Niu, "Captioning transformer with stacked attention modules," *Appl. Sci.*, vol. 8, no. 5, 2018, doi: 10.3390/app8050739.

[27] W. Zhang, W. Nie, X. Li, and Y. Yu, "I m age Caption Generation With Adaptive Transfor m er," pp. 521–526, 2019.

[28] S. He, W. Liao, H. R. Tavakoli, M. Yang, B. Rosenhahn, and N. Pugeault, "Image Captioning Through Image Transformer," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12625 LNCS, pp. 153–169, 2021, doi: 10.1007/978-3-030-69538-5_10.

[29] C. Cormier, "Bleu," *Landscapes*, vol. 7, no. 1, pp. 16–17, 2005, doi: 10.3917/chev.030.0107.

[30] S. Banerjee and A. Lavie, "METEOR: An automatic metric for mt evaluation with improved correlation with human judgments," *Intrinsic Extrinsic Eval. Meas. Mach. Transl. and/or Summ. Proc. Work. ACL 2005*, no. June, pp. 65–72, 2005.

[31] C. Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Proc. Work. text Summ. branches out (WAS 2004)*, no. 1, pp. 25–26, 2004.

[32] R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07-12-June, pp. 4566–4575, 2015, doi: 10.1109/CVPR.2015.7299087.

[33] R. Staniute and D. Šešok, "A systematic literature review on image captioning," *Appl. Sci.*, vol. 9, no. 10, 2019, doi: 10.3390/app9102024.