



Image captioning using hybrid of VGG16 and bidirectional LSTM model

Yufis Azhar*¹, M. Randy Anugerah², Muhammad Al Reza Fahlopy³, Alfin Yusriansyah⁴

Informatics Department, Universitas Muhammadiyah Malang, Indonesia^{1,2,3,4}

Article Info

Keywords:

Image Captioning, VGG16, LSTM, LSTM Bidirectional, BLEU

Article history:

Received: October 11, 2022

Accepted: November 04, 2022

Published: November 30, 2022

Cite:

Y. Azhar, M. R. Anugerah, M. A. R. Fahlopy, and A. . Yusriansyah, "Image Captioning using Hybrid of VGG16 and Bidirectional LSTM Model", *KINETIK*, vol. 7, no. 4, Nov. 2022.

<https://doi.org/10.22219/kinetik.v7i4.1568>

*Corresponding author.

Yufis Azhar

E-mail address:

yufis@umm.ac.id

Abstract

Image captioning is one of the biggest challenges in the fields of computer vision and natural language processing. Many other studies have raised the topic of image captioning. However, the evaluation results from other studies are still low. Thus, this study focuses on improving the evaluation results from previous studies. In this study, we used the Flickr8k dataset and the VGG16 Convolutional Neural Networks (CNN) model as an encoder to generate feature extraction from images. Recurrent Neural Network (RNN) uses the Bidirectional Long-Short Term Memory (BiLSTM) method as a decoder. The results of the image feature extraction process in the form of feature vectors are then forwarded to Bidirectional LSTM to produce descriptions that match the input image or visual content. The captions provide information on the object's name, location, color, size, features of an object, and surroundings. A greedy Search algorithm with Argmax function and Beam-Search algorithm are used to calculate Bilingual Evaluation Understudy (BLEU) scores. The results of the evaluation of the best BLEU scores obtained from this study are the VGG16 model with Bidirectional LSTM using Beam Search with parameter $K = 3$ and the BLEU-1 score is 0.60593, so this score is superior to previous studies.

1. Introduction

In recent years, the processing of an image or object has undergone many very fast and significant advances. Examples of such advances are object detection and image classification [1][2]. Of the many developments and advances in the field of object detection and image classification like image captioning where these researchers use a combination of computer vision and natural language processing to make one or more sentences automatically and can explain the contents of the image or object [3][4].

The output of image captioning has a substantial positive impact on humans, such as in making titles for news images, descriptions of medical images, taking text-based images, and helping people who are blind to interact with images on online sites and social media [5]. In addition to the positive impact obtained from image captioning, there are challenges where a machine must understand the content or meaning of an object in an image that has a relationship with human language. Not only should a description generation model be able to identify the items in an image, but it should also be able to show their relationships with one another [6].

In image captioning, there are two main processes, namely encoder, and decoder. The definition of an encoder is an activity to extract features in an image or visual content and then produce a data set that represents every object in the picture. The notion of a decoder is a sentence reconstruction process based on a collection of previously identified objects and produces a text or sentence description of the image [7][8][9].

Over time, many other researchers have used various encoder and decoder methods to conduct image captioning research. Kumar developed image captioning in this research [10] using deep learning with VGG16 as a feature extractor model and Long-Short Term Memory (LSTM) to create a descriptive sentence that is syntactically and semantically appropriate. Kumar uses the Bilingual Evaluation Understudy (BLEU) as an evaluation model and gets a BLEU score of 0.50.

From a total of 32 experiments conducted by Hejazi [11], the best-proposed model is VGG16 and GRU without using dropout as a model for image captioning of Arabic datasets. Get The best BLEU scores were BLEU-1 = 36.5, BLEU-2 = 21.4, BLEU-3 = 12, and BLEU-4 = 6.6. This BLEU score is superior to the 31 proposed models using dropout or not using dropout.

Mulyanto in his research [12] uses the CNN-LSTM model to perform image captioning on images to produce Indonesian text that works well. The model proposed in the test set obtained a result of 50.0 for BLEU-1 and BLEU-3 of 23.9.

Nugraha [13] focuses on developing a generative model connecting machine translation and computer vision to generate image descriptions in Bahasa Indonesia. And the proposed model use Inception-V3 and 3 GRU layers.

Nugraha in this research used the Flickr 30k dataset translated into Indonesian that obtained BLEU-1, BLEU-2, BLEU-3, and BLEU-4 of 36, 17, 6, 2 respectively.

Wang in his research [14] used the AlexNet and Bidirectional LSTM models with an additional method, namely Multi task learning to perform image captioning. The proposed model yields 58.4, 42.1, 28.6, and 18.2 for BLEU-1, BLEU-2, BLEU-3, BLEU-4 respectively.

From previous research, we can conclude that the evaluation results obtained are still low because when the bleu score is far from 1.0, the caption results are bad. We aim to determine the best algorithm as a method for generating predictive sentences and calculating BLEU scores at the evaluation stage. In order to improve the evaluation results of previous studies, the contribution made in our research is to combine the VGG 16 model and the Bidirectional LSTM approach with the Greedy search argmax function and Beam search algorithm.

2. Research Method

This study has a design system consisting of 4 main processes preprocessing (image & caption), image feature extraction using the VGG16 model, caption generation using Bidirectional LSTM, and evaluation or scoring model using Greedy search algorithm with Argmax function and Beam search algorithm with parameter $K = 3$. The design system illustrated in Figure 1.

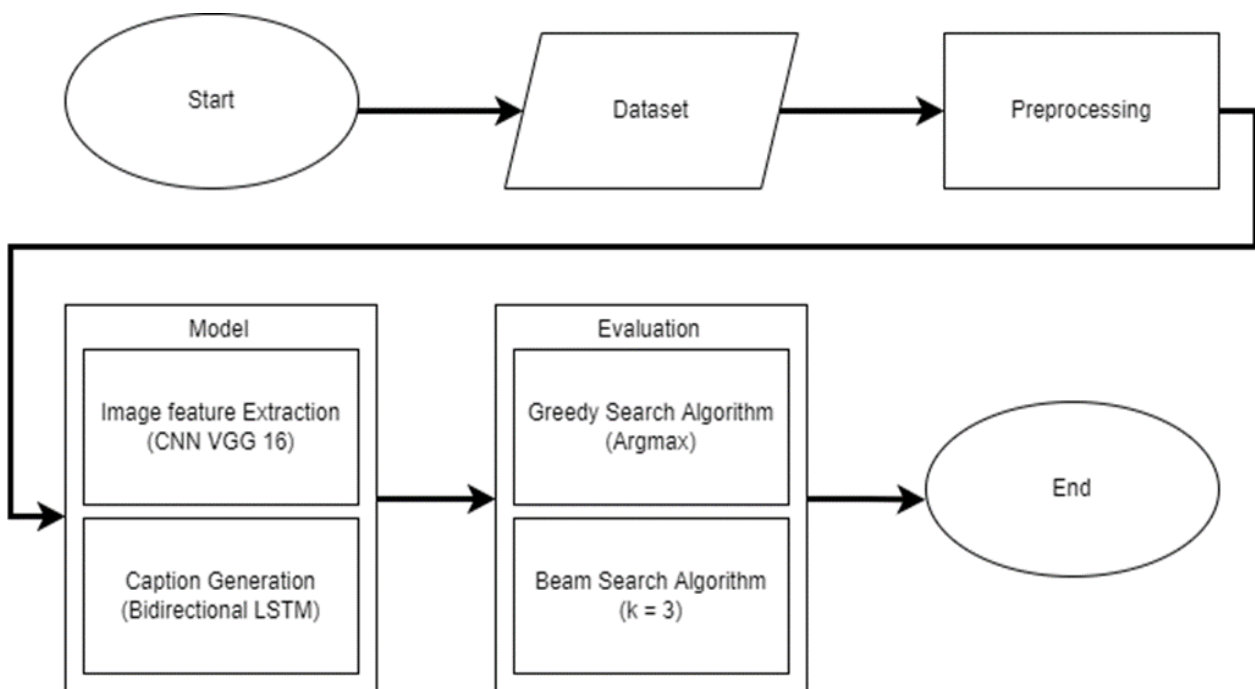


Figure 1. System Design of Image Captioning

2.1 Dataset Collection

The data used for this research is Flickr8k, and this dataset is obtained from the Kaggle website (<https://www.kaggle.com/datasets/sayanf/flickr8k>). Flickr8k is a dataset consisting of eight thousand (8000) images with five target captions (provided by humans). This dataset is divided into two folders, namely text and image folders. Each image has a caption stored together with its respective ID and has a unique ID.

The datasets in our study were divided into training, test, and validation parts. The training section has (6000) images, the testing section has (1000) images, and the validation section also has (1000) images.

2.2 Preprocessing on Image

In this study, images served as input for a CNN model neural network. Therefore, image preprocessing converts the image object into an RGB array. Then the array is resized to (224, 224, 3).

From our dataset, we use transfer learning to extract the image features. For this purpose, we use a pre-trained model and input the results of extracting features from the image into the neural network of the RNN model as an interpretation of the images. The CNN and RNN-LSTM training flow diagrams are illustrated in Figure 2.

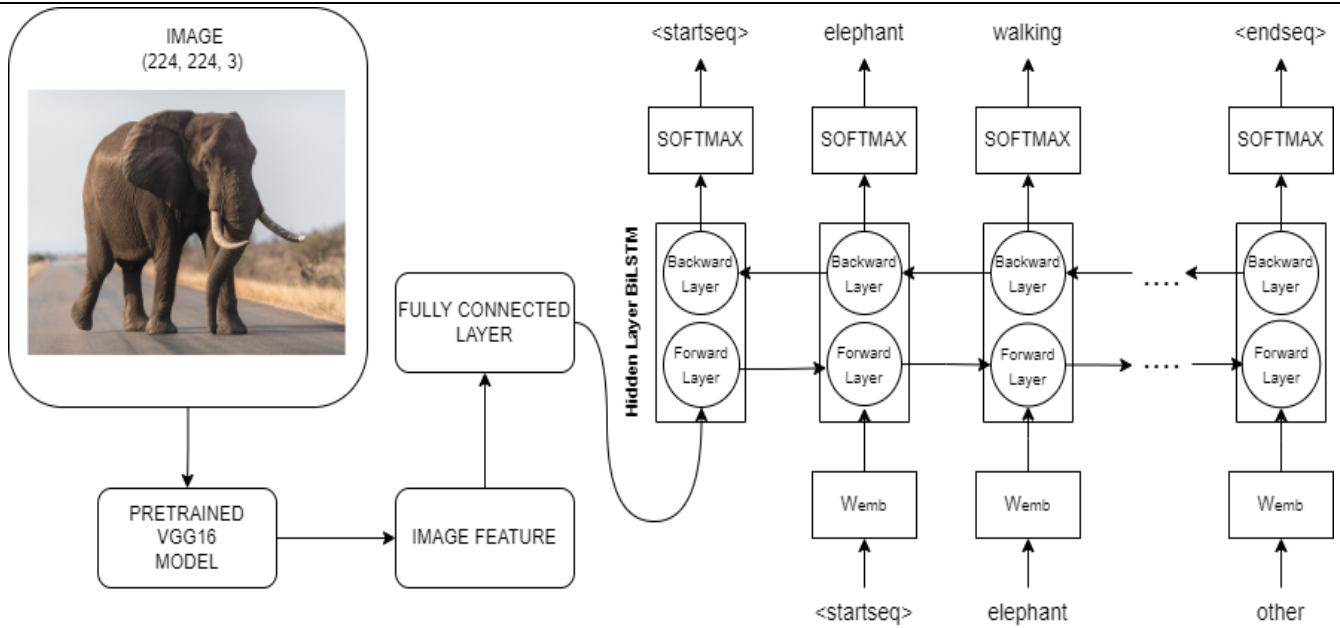


Figure. 2 Training CNN and Bidirectional LSTM (BiLSTM) Flow Diagram

2.3 Preprocessing on Caption

Before preprocessing the caption, we fetch the file containing the text with the appropriate image ID. Then, we perform a looping operation to create a python dictionary that contains a key in the form of an image ID and the value contains a list of image descriptions.

Preprocessing a caption is used to clean data, such as removing punctuation marks, converting tokens to lowercase, removing tokens containing numbers, separating each token with a space, and giving the token “<startseq>” at the beginning of each description as a sign to start the system. generate sentences. And stop producing sentences when given the token “<endseq>” at the end of each description. The flow of the system diagram in determining the next word is shown in Table 1.

Table 1. The Next Sequence of Text and Word Input in the Sequence for the Training Phase

Input Sequence	Next Word
<startseq>	elephant
<startseq>, elephant	walking
<startseq>, elephant, walking	on
<startseq>, elephant, walking, on	the
<startseq>, elephant, walking, on, the	street.
<startseq>, elephant, walking, on, the, street.	<endseq>

2.4 Convolutional Neural Networks Model: Encoders

This section discusses the Convolutional Neural Network (CNN) model used for the encoder process. Convolutional Neural Network is a Deep Learning algorithm usually used for image processing [15][16]. Convolutional Neural Network in image captioning functions as a feature extraction process on the input image to produce a feature vector consisting of a matrix with several pixels that match the input and image filters [17].

In this study using the VGG16 encoder model [18][19]. The VGG16 network has 16 layers total, of which two are dense layers and the other 15 are convolution layers. Feature extraction performed on image input must have dimensions of 224 * 224. Each Convolutional block layer has a filter size of 3 x 3. Then the output of the resulting Convolutional is forwarded to the max pooling layer using dimensions of 2 * 2 pixels with a stride length of 2 for the first to third Conv Blocks, then Conv Blocks four to five have a stride length of 1. Then enter the fully connected hidden layer with 4096 units. The architecture of the VGG16 model is shown in Figure 3.

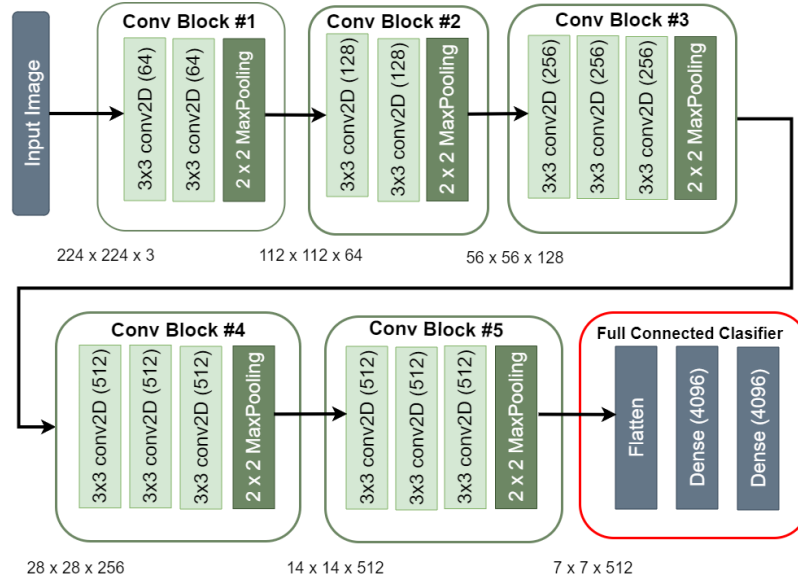


Figure 3. Model Architecture VGG16

2.5 Recurrent Neural Networks Model: Decoders

In this study, we use the Bidirectional LSTM (BiLSTM) model as a decoder which aims to string words to describe an image. This model is usually used in Natural Language Processing (NLP). BiLSTM has 2 layers of memory blocks, namely a forward hidden layer and a backward hidden layer. BiLSTM is capable of receiving input from both directions (forward and backward). Thus, more information will be provided, and better prediction results. Then, the LSTM forward hidden layer and backward hidden layer are combined using concatenation before being sent to the next layer [20]. The intuition behind the LSTM is illustrated in Figure 5.

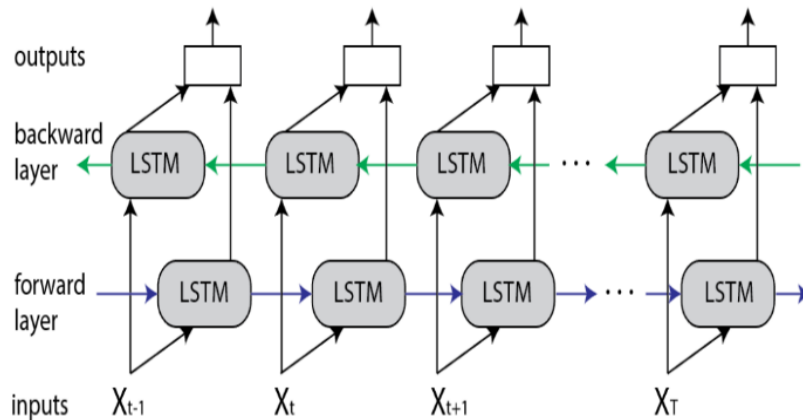


Figure 5. Intuition Behind Bidirectional LSTM

In Figure 5, we illustrate with an example. Suppose there is a missing sentence “I am very _____. I want to eat ten servings of fried chicken”. With the backward hidden layer, the BiLSTM concept reverses the input information, so that it can get information from the next (future) time step to predict the word "hungry" in the gap sentence.

2.6 Training and Evaluation Model

We used the Google Colaboratory Notebook Pro for model training with specifications 36 GB ram and 226 GB memory. We trained the RNN model with embedding size 256, batch size 64, Adam optimizer, and categorical cross entropy loss function.

The validation loss value is a metric that is used evaluation the model for every epoch returned by the loss function. Throughout the training phase, we monitor the value of model validation loss. We perform checkpoints to store the best results from each epoch in the model as the validation loss value increases. In the model training phase, we use the model with the highest validation accuracy value in the data set.

2.7 Performance Measurement of the Model

In measuring the performance of the model, we use two algorithms Greedy search and Beam search. This algorithm serves to generate image descriptions of the VGG16 and BiLSTM models. Greedy search algorithm with Argmax function: In each search step, Greedy Search selects the token with the highest potential to generate tokens [21].

Implementing the argmax function will generate an image description of the best set of tokens. We predict the sequence output from the token sequence "<startseq>" to the end of the predicted token sequence "<endseq>". An illustration of the Greedy search algorithm using the Argmax function can be seen in Figure 6.

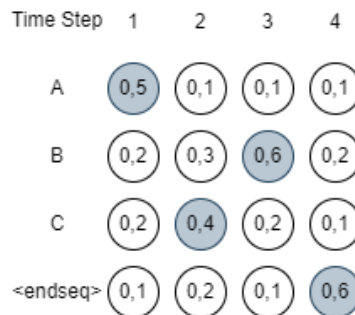


Figure 6. The Process of Greedy Search using Argmax Function

In Figure 6, we illustrate with an example. Suppose there are four tokens "A", "B", "C", and "<endseq>". In the third step, token "B" is chosen because it has the highest probability. The best candidate sequence using this algorithm produces A, C, B, and <endseq> in describing the image that is input to the system. This algorithm has the advantage of choosing tokens, but it may not be the best choice for determining the complete statement.

Beam search algorithm is the best version of the heuristic algorithm in Greedy search. This algorithm yields impressive performance in the search for the best tokens. This makes Beam search a frequently used algorithm in many State-of-the-Art Natural Language Processing systems [22][23].

The advantage of Beam search is that it has a parameter called beam size, denoted by the letter K [24]. Beam size is an algorithm that selects a large number of alternative tokens with the highest vocabulary probability in each search step according to the specified parameters. Illustration of the Beam search algorithm can be seen in Figure 7.

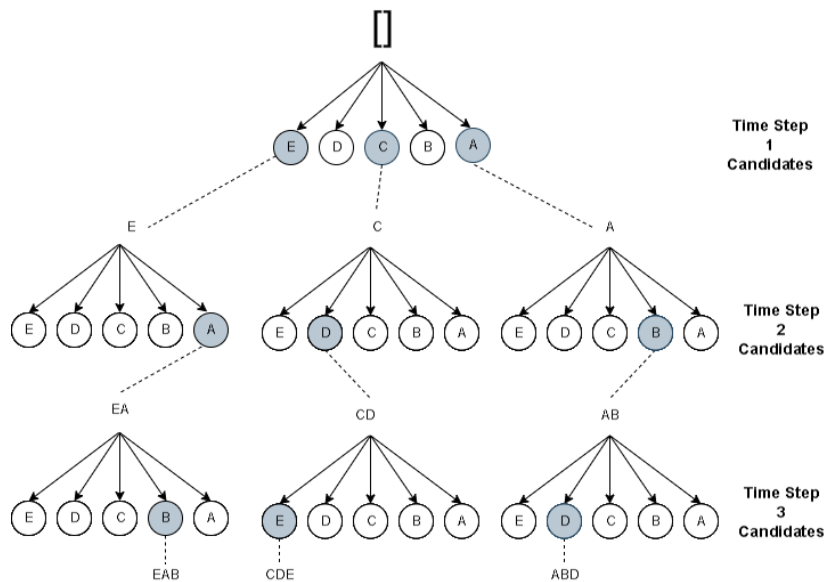


Figure 7. The Process of Beam Search (k = 3)

In Figure 7, we illustrate this with an example. Suppose there are three Beam sizes "K=3" with five tokens "A", "B", "C", "D", and "E", one of which is <endseq>. The best candidate sequence using this algorithm produces A, C, E, AB, CD, EA, ABD, CDE, and EAB. This algorithm will choose which sentence has the highest probability to describe the image input to the system.

2.8 BLEU Score

The bilingual Evaluation Understudy (BLEU) score is a metric used to measure the accuracy and quality of the sentences generated by the model [13][25]. The values of the BLEU score are always between 0-1. If the sentence generated by machine translation matches the actual description, then the BLEU value is 1.0. On the other hand, if the resulting sentence is much different from the actual description, the BLEU value given is 0.0 [26][23]. The result of image captioning is shown in Figure 8. The smaller the value of the BLEU score, the less output from the machine will be irrelevant (Hallucination).



Figure 8. Example of Image Captioning Results and Explanations

3. Results and Discussion

In this study, we trained 6000 training images and 1000 validation images from our dataset using VGG16 as encoder, BiLSTM as decoder, with Adam optimizer, batch size 64, epoch 50, embedding size 256, and categorical cross entropy as loss function, in each scenario. We took 15 images from Google and used them as test data to test the model. This model has never studied these images, and we can find out whether the model is good in learning or not.

The evaluation results were obtained by using qualitative and quantitative measurements. The results of the qualitative measurement, obtained from several images with the information generated by the model, are shown in Table 2. Most of the predicted sentences are grammatically correct. However, there are some results from the resulting image captions that do not match the context of the given image. In this case, it is caused by the choice of words to be predicted is limited to five target texts (provided by humans). This problem can be solved by taking advantage of subword units. As in the research [27], which proves that texts that use sub-word units are able to provide better performance.

Quantitative results it is measured in the form of BLEU scores. Each caption of the predicted image generated by the model is compared with five target captions (provided by humans) which serve as a reference for calculating the BLEU score.

Table 2. Model Bidirectional Generated Captions



Image from Google	Caption Result from Model BiLSTM	
	Greedy Search (Argmax)	Beam Search
	A person is riding a bike on a dirt hill	A person is riding dirt bike on a dirt hill
	A man in a blue uniform is kicking a soccer ball	A soccer player in a blue uniform is kicking the ball

Table 3 illustrates the performance of the proposed model using the BLEU score. We conducted four experiments to test the LSTM model in making a sentence. First, we use Unidirectional LSTM with a Greedy search algorithm (Argmax). The second uses Unidirectional LSTM with a Beam search algorithm (k = 3). Third, we use Bidirectional LSTM with a Greedy search algorithm (Argmax). Lastly, we use Bidirectional LSTM with a Beam search algorithm (k =

3). With a set of specifications determined, we get the results that the Bidirectional LSTM with Beam search outperformed the Unidirectional LSTM.

Table 3. BLEU Scores

Model	BLEU Scores	
	Greedy Search (Argmax)	Beam Search with K = 3 (beam width)
VGG16 + Unidirectional LSTM	BLEU-1 = 0.578005 BLEU-2 = 0.331570 BLEU-3 = 0.224491 BLEU-4 = 0.109854	BLEU-1 = 0.579857 BLEU-2 = 0.342115 BLEU-3 = 0.239830 BLEU-4 = 0.122404
VGG16 + Bidirectional LSTM	BLEU-1 = 0.596393 BLEU-2 = 0.347015 BLEU-3 = 0.242815 BLEU-4 = 0.123927	BLEU-1 = 0.605931 BLEU-2 = 0.355516 BLEU-3 = 0.251673 BLEU-4 = 0.131444

However, each type of model has its advantages and limitations, such as the performance of Bidirectional LSTM is better than Unidirectional LSTM. This is because the Unidirectional LSTM only performs one-way input and only stores the past context. On the other hand, Bidirectional LSTM runs inputs in both forward and backward directions to retain information from the past and future, helping us better understand the context. In addition, the hidden layer of Bidirectional LSTM is more complex than Unidirectional LSTM.

Furthermore, based on Table 3, it can be seen that the Beam search algorithm is able to produce better information evaluation than the Greedy search algorithm. This is because Beam search can use a parameter named Beam size (K=3), which makes it possible to select more than one alternative token with the highest probability vocabulary at each search step. Meanwhile, Greedy search only selects one token as its default value. However, the estimated time required for BLEU scoring in beam search is longer. So, we can see that the beam search algorithm with parameter Beam size (K=3) managed to get the best performance with a BLEU-1 score of 0.605931, BLEU-2 of 0.355516, BLEU-3 of 0.251673, and BLEU-4 of 0.131444.

Table 4. Comparative Best Results with Other Research

Research Work	CNN	RNN	BLEU-1
Research [10]	VGG16	LSTM	0.50
Research [11]	VGG16	GRU	0.37
Research [12]	CNN	LSTM	0.50
Research [13]	InceptionV3	GRU	0.36
Research [14]	AlexNet	BiLSTM (Multi Task Learning)	0.58
Best Proposed Our Model	VGG16	BiLSTM	0.60

4. Conclusion

In this study, we use the Flickr8k dataset with proposed model for image captioning. Based on the results it can be concluded that using the VGG16 Bidirectional LSTM model and beam search algorithm obtained, the BLEU-1 score of 0.60 is higher than from previous studies. This shows that the model can produce grammatically correct sentence predictions. However, some sentence predictions are still wrong and cannot be matched with the context in some images.

For further development of this research, the researcher gives suggestions to use datasets that have a larger capacity, such as the Flickr30k dataset or MS COCO. By using other CNN methods such as InceptionNet and GoogleNet. In additional, also can use other RNN models such as GRU by adding Attention, Glove, FastText, etc.

References

- [1] J. Zhang, K. Li, Z. Wang, X. Zhao, and Z. Wang, "Visual enhanced gLSTM for image captioning," *Expert Syst. Appl.*, vol. 184, no. June, p. 115462, 2021. <https://doi.org/10.1016/j.eswa.2021.115462>
- [2] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, "Image captioning: Transforming objects into words," *Adv. Neural Inf. Process. Syst.*, vol. 32, no. NeurIPS, pp. 1–11, 2019. <https://doi.org/10.48550/arXiv.1906.05963>
- [3] Z. Deng, Z. Jiang, R. Lan, W. Huang, and X. Luo, "Image captioning using DenseNet network and adaptive attention," *Signal Process. Image Commun.*, vol. 85, p. 115836, 2020. <https://doi.org/10.1016/j.image.2020.115836>

- [4] O. Sargar and S. Kinger, "Image captioning methods and metrics," *2021 Int. Conf. Emerg. Smart Comput. Informatics, ESCI 2021*, pp. 522–526, 2021. <https://doi.org/10.1109/ESCI50559.2021.9396839>
- [5] I. Hrga and M. Ivašić-Kos, "Deep image captioning: An overview," *2019 42nd Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2019 - Proc.*, pp. 995–1000, 2019. <https://doi.org/10.23919/MIPRO.2019.8756821>
- [6] A. Nursikuwagus, R. Munir, and M. L. Khodra, "Image Captioning menurut Scientific Revolution Kuhn dan Popper," *J. Manaj. Inform.*, vol. 10, no. 2, pp. 110–121, 2020. <https://doi.org/10.34010/jamika.v10i2.2630>
- [7] Y. Pan, T. Yao, Y. Li, and T. Mei, "X-Linear Attention Networks for Image Captioning," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 10968–10977, 2020. <https://doi.org/10.1109/CVPR42600.2020.01098>
- [8] M. A. Al-Malla, M. A. Al-Malla, A. Jafar, and N. Ghneim, "Pre-trained CNNs as Feature-Extraction Modules for Image Captioning," *ELCVIA Electron. Lett. Comput. Vis. Image Anal.*, vol. 21, no. 1, pp. 1–16, 2022. <https://doi.org/10.5565/rev/elcvia.1436>
- [9] K. Chandhar, C. H. Sandeep, M. Akarapu, K. R. Chythanya, and V. Thirupathi, "Deep learning model for automatic image captioning," *Int. Conf. Res. Sci. Eng. Technol.*, vol. 2418, no. May, p. 020074, 2022. <https://doi.org/10.1063/5.0081847>
- [10] A. Kumar, "Image Captioning and Image Retrieval," vol. 4, no. 4, pp. 909–912, 2019.
- [11] H. Hejazi and K. Shaalan, "Deep Learning for Arabic Image Captioning: A Comparative Study of Main Factors and Preprocessing Recommendations," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 11, pp. 37–44, 2021. <https://dx.doi.org/10.14569/IJACSA.2021.0121105>
- [12] E. Mulyanto, E. I. Setiawan, E. M. Yuniarno, and M. H. Purnomo, "Automatic Indonesian Image Caption Generation using CNN-LSTM Model and FEED-ID Dataset," *2019 IEEE Int. Conf. Comput. Intell. Virtual Environ. Meas. Syst. Appl. CIVEMSA 2019 - Proc.*, 2019. <https://doi.org/10.1109/CIVEMSA45640.2019.9071632>
- [13] 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. <https://doi.org/10.1109/ICoICT.2019.8835370>
- [14] 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. <https://doi.org/10.1145/3115432>
- [15] M. Chohan, A. Khan, M. S. Mahar, S. Hassan, A. Ghafoor, and M. Khan, "Image captioning using deep learning: A systematic literature review," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 5, pp. 278–286, 2020. <https://dx.doi.org/10.14569/IJACSA.2020.0110537>
- [16] Y. Azhar, M. C. Mustaqim, and A. E. Minarno, "Ensemble convolutional neural network for robust batik classification," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1077, no. 1, p. 012053, 2021. <https://doi.org/10.1088/1757-899X/1077/1/012053>
- [17] S. S. Rawat, K. S. Rawat, and R. Nijhawan, "A novel convolutional neural network-gated recurrent unit approach for image captioning," *Proc. 3rd Int. Conf. Smart Syst. Inven. Technol. ICSSIT 2020*, no. IcSSIT, pp. 704–708, 2020. <https://doi.org/10.1109/ICSSIT48917.2020.9214109>
- [18] S. Tammina, "Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images," *Int. J. Sci. Res. Publ.*, vol. 9, no. 10, p. p9420, 2019. <http://dx.doi.org/10.29322/IJSRP.9.10.2019.p9420>
- [19] B. I. S. L. Nalbalwar, *Advances in Intelligent Systems and Computing 810 Computing, Communication and Signal Processing*, vol. 1. 2018.
- [20] Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection," *Expert Syst. Appl.*, vol. 185, no. June, p. 115524, 2021. <https://doi.org/10.1016/j.eswa.2021.115524>
- [21] S. Welleck, I. Kulikov, S. Roller, E. Dinan, K. Cho, and J. Weston, "Neural Text Generation with Unlikelihood Training," no. i, pp. 1–17, 2019. <https://doi.org/10.48550/arXiv.1908.04319>
- [22] C. Meister, T. Vieira, and R. Cotterell, "Best-first beam search," *Trans. Assoc. Comput. Linguist.*, vol. 8, pp. 795–809, 2020. https://doi.org/10.1162/tacl_a_00346
- [23] A. Deshpande, J. Aneja, L. Wang, A. G. Schwing, and D. Forsyth, "Fast, diverse and accurate image captioning guided by part-of-speech," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 10687–10696, 2019. <https://doi.org/10.1109/CVPR.2019.01095>
- [24] J. M. Czum, "Dive Into Deep Learning," *J. Am. Coll. Radiol.*, vol. 17, no. 5, pp. 637–638, 2020. <https://doi.org/10.1016/j.jacr.2020.02.005>
- [25] D. H. Fudholi, A. Zahra, and R. A. N. Nayoan, "A Study on Visual Understanding Image Captioning using Different Word Embeddings and CNN-Based Feature Extractions," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, no. 1, pp. 91–98, 2022. <https://doi.org/10.22219/kinetik.v7i1.1394>
- [26] 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. <https://doi.org/10.1109/ICoICT49345.2020.9166244>
- [27] M. Kuyu, A. Erdem, and E. Erdem, "Altsözcük Ö ğ eleri ile Türkçe Görüntü Altyazılama Image Captioning in Turkish with Subword Units," *2018 26th Signal Process. Commun. Appl. Conf.*, pp. 1–4. <https://doi.org/10.1109/SIU.2018.8404431>