INTERNATIONAL JOURNAL
ON INFORMATICS VISUALIZATION

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

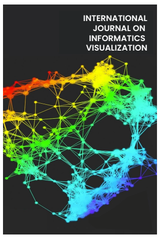


Image Captioning with Style Using Generative Adversarial Networks

Dennis Setiawan [a,*], Maria Astrid Coenradina Saffachrissa [a], Shintia Tamara [a], Derwin Suhartono [a]

[a] *Computer Science Department, School of Computer Science, Bina Nusantara University, Palmerah, Jakarta 11480, Indonesia*
*Corresponding author: *dennis.setiawan@binus.ac.id*

***Abstract*— Image captioning research, which initially focused on describing images factually, is currently being developed in the direction of incorporating sentiments or styles to produce natural captions that reflect human-generated captions. The problem this research tries to solve the problem that captions produced by existing models are rigid and unnatural due to the lack of sentiment. The purpose of this research is to design a reliable image captioning model that incorporates style based on state-of-the-art SeqCapsGAN architecture. The materials needed are MS COCO and SentiCaps datasets. Research methods are done through literature studies and experiments. While many previous studies compare their works without considering the differences in components and parameters being used, this research proposes a different approach to find more reliable configurations and provide more detailed insights into models' behavior. This research also does further experiments on the generator part that have not been thoroughly investigated. Experiments are done on the combinations of feature extractor (VGG-19 and ResNet-50), discriminator model (CNN and Capsule), optimizer (Adam, Nadam, and SGD), batch size (8, 16, 32, and 64), and learning rate (0.001 and 0.0001) by doing a grid search. In conclusion, more insights into the models' behavior can be drawn, and better configuration and result than the baseline can be achieved. Our research implies that research in comparative studies of image recognition models in image captioning context, automated metrics, and larger datasets suited for stylized image captioning might be needed for furthering the research in this field.**

***Keywords*— Stylized image captioning; SeqCapsGAN; sentiments or styles; Generative Adversarial Network (GAN); capsule; discriminator; generator.**

Manuscript received 31 Oct. 2021; revised 20 Dec. 2021; accepted 5 Jan. 2022. Date of publication 31 Mar. 2022.

I. Introduction

The advancement of the computer vision domain in recent years has led to image classification and object detection. These advancements allow the development of image captioning, where one could automatically generate one or more descriptive sentences of visual contents in an image. The applications of image captioning are proven to be beneficial.

Automatic image captioning provides convenience for many different fields. For example, generating captions for news images, generating descriptions for medical images, providing information for visually impaired people, and developing human-machine interactions. Specifically, image captioning in image indexing can also be utilized for Content-Based Image Retrieval (CBIR), a text-based image search [1]. Also, the advancement in research allows future researchers to continue or develop new image captioning models that would be able to complete more complex tasks more effectively and efficiently.

Image captioning has been a fundamental proposition in machine learning that connects the computer vision domain with natural language processing (NLP). The popularity of image captioning rises along with humans' desire to create a machine that can replicate the ability to identify and describe an image in more detail. However, compared to a machine-generated caption that describes an image factually, human-generated captions are relatively better in describing the image content. Humans can adjust the sentence's sentiment to match certain requirements, which results in better understanding. Meanwhile, image captioning models have yet to replicate human communications, resulting in rigid and unnatural captions. Therefore, for machines to generate human-like captions, it needs a semantic meaning in the form of style.

The style allows an image description to be clearer, more attractive and has the right sentiment [2]. It is also useful to reflect characters, improve marketing communication, increase social interaction, and improve user engagement [3] [4]. Style can also be utilized for an application that aids visually impaired people in accessing more engaging and natural visual information. Realizing the importance of style, this research developed captioning image models that can generate captions with style.

Several studies have been carried out in a research effort to produce a varying, natural, and flexible image caption. SentiCap research [5], which initiated the use of style in image captioning, used a switching RNN with word-level supervision that allows the model to learn effectively with a small amount of data.

The first research that emphasizes aspects of naturalness and diversity in image descriptions [6] proposes Conditional GAN to be used to generate image descriptions, thus creating more varying captions that are not only focused on the ground truth. The research by Dognin [7] was based on the Conditional GAN (CGAN) framework using context-aware captioning and Self-Critical Sequence Training (SCST) with semantical gap and dataset bias as the main concern. It improved the CGAN algorithm framework by adding the attention mechanism at generator & discriminator elements.

Further research developments [8] used the two-stage ATTEND-GAN architecture, which trains the model on a large factual dataset and a small stylistic dataset. After that, an adversarial training mechanism is applied to direct the generator to produce sentimental captions. The latest stylized image captioning domain research resulted in a new model called SeqCapsGAN [9]. The SeqCapsGAN model uses the Generative Adversarial Network (GAN) framework and capsule component to alleviate missing information caused by pooling layers in the NN-based discriminator [10].

This study aims to implement, evaluate, and improve the performance of the stylized image captioning model based on the SeqCapsGAN model through experiments of the generator component, which the previous study has not further explored. The objectives are to find more reliable configurations and provide the models' behavior in more detail for future references in optimizing and developing the image captioning model. This research compared the VGG-19 used in SeqCapsGAN with the ResNet-50 model with higher Top-1 accuracy in image recognition tasks [11]. It also compares the optimizer's performance of Adam used in the baseline with Nadam, which has better performance in reducing training & validation loss [12], and SGD could give better generalization in image features [13]. The batch size and learning rate hyperparameters are also experimented with in several studies [14] [15] to find out the impact of bigger and smaller values to produce a more reliable model and experimental data that might be useful for further research [16].

II. Materials and Method

A. Datasets

This research uses MSCOCO and SentiCap Dataset, also used in our SeqCapsGAN model [9]. The MSCOCO dataset is used in generator and discriminator pre-training to learn how to map visual features extracted from images to factual captions. SentiCap dataset is used at the training stage of the GAN framework to generate & evaluate captions with styles in the form of positive and negative sentiments.

The MSCOCO dataset [17] is a dataset for image recognition provided by Microsoft. This dataset provides a group of pictures of everyday objects consisting of 82,783 images with 413,915 captions in the training set and 40,504 images with 202,520 captions in the validation set used in this research.

The SentiCap dataset [5] is a dataset derived from the MSCOCO dataset with additional captions containing positive and negative sentiments. It was obtained from research conducted by members of the Computational Media Lab. This dataset is used to add a sentiment to the resulting caption. The sentiment in question comes from the point of view of an objective observer. It is subject to an observer who does not know the background and actual events that occurred from the photo through crowd-sourcing Amazon mTurk. SentiCap dataset used in this research are as follows:

- Training Set: 998 images with 2,873 positive sentiment captions and 997 images with 2,468 negative sentiment captions.
- Validation Set: 174 images with 409 positive captions and 174 images with 429 negative sentiment captions.

B. Pre-processing

Pre-processing is the initial operation stage performed on the input before processing the captioning model. This stage is divided into two; namely, image pre-processing and caption pre-processing. Image pre-processing is cropped to maintain a 1:1 aspect ratio and resized to 224x224. The aim is to normalize the captions and build the vocabulary of this research. There are several steps in the normalization process: delete symbols, change '&' to 'and', delete multiple spaces, lowercase captions, delete captions with more than 25 words. The vocabularies are built as a mapping of word to index of all the words in the datasets with 3 special tokens: '<NULL>', '<START>', and '<END>'. The last two tokens indicate the starting and ending of the caption vector, while the first token is used to fill in the gap words to make the caption a static 25 words vector. This process results in 28,773 words, with the special tokens excluded.

C. Experiments

To achieve our goals described in Section 1, this research does several experiments for these components in the Generative Adversarial framework:

1) Generator:

- Feature Extractor: VGG-19 and ResNet-50
- Optimizer: Adaptive Moment Estimation (Adam), Stochastic Gradient Descent (SGD), and Nesterov-accelerated Adaptive Moment Estimation (Nadam).

2) Discriminator Model: CNN (neuron) and Capsule.

3) Hyperparameters:

- Learning Rate: 0.001 (1e-3) dan 0.0001 (1e-4)
- Batch Size: 8, 16, 32, 64

This research does a grid search in executing the experiments resulting in 96 experiments in total. Each combination is trained for 10 epochs each for generator pre-training, discriminator pre-training, and GAN training.

D. Training

The model's training process in more detail is shown in Figure 1. This block diagram contains the stages of the process carried out, as well as the inputs used, and the outputs generated from each process. The process begins by loading

MSCOCO, vocabulary, and SentiCap data obtained from the pre-processing stage, namely *train/val_coco_data.pkl*, *word_to_idx.pkl*, and *train/val_senticap_data.pkl*. MS COCO data and vocabulary was used for the generator pre-training process and produce a generator model that can generate factual captions.

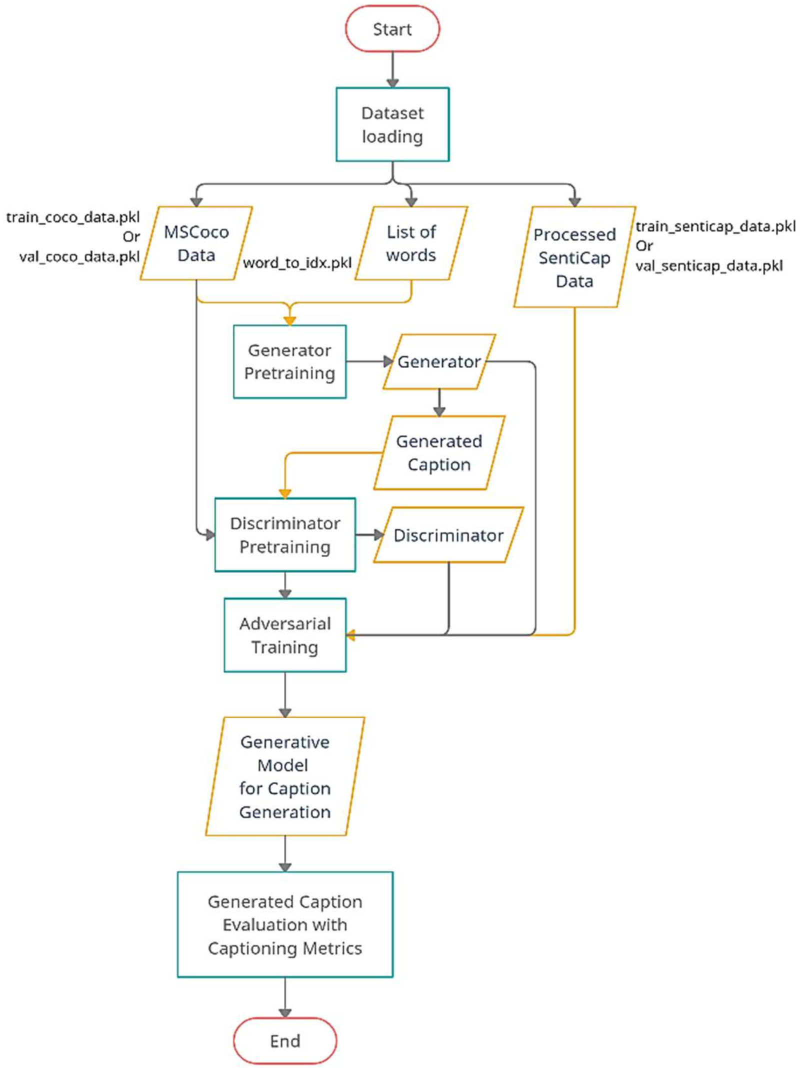


Fig. 1 Training Activity Flowchart

The resulting generator model and MSCOCO data was used for the stages of the discriminator pre-training process, which produces the discriminator model. Generator, the discriminator, and SentiCap data was used in the adversarial training process and produce other model generators that can generate captions with styles. The generator maximizes the chance of the discriminator misclassifying the caption, while the discriminator improves accuracy to correctly classify captions into real and fake classes by optimizing the loss function according to WGAN settings [18]. Finally, the generator model that can generate captions with these styles was evaluated using the evaluation metric.

E. Evaluation Method

Evaluation is done by processing captions using several automated evaluation metrics. The evaluation metrics used were BLEU (with n-grams 1 to 4), ROUGE-L, and CIDEr. Evaluation metrics could produce a score that could be used as a benchmark for comparing the accuracy of each model. In addition to the automated image captioning metrics mentioned, this research also uses loss and qualitative sampling evaluation of the captions produced by the model.

The loss function in this study is the Mean Squared Error (MSE) which measures the penalty for bad predictions. BLEU (Bilingual Evaluation Understudy Score) is a precision-based metric used to measure the similarity of words and phrases (2-4 words) that the model has successfully learned that refers fully to the reference caption [17]. ROUGE-L stands for Recall-Oriented Understudy for Gisting Evaluation, and L represents Longest Common Subsequence (LCS) that reflects the order of words at the sentence level so that this metric can assess the suitability of the sentence structure/grammatical of the resulting caption [19]. CIDEr (Consensus-based Image Description Evaluation metric) is a metric used to capture human consensus. This metric can measure the similarity of the resulting caption to how most people would describe the image [20]. One of the significant things that CIDEr does is to consider the word saliency, or the importance of a word based on the TF-IDF (Term Frequency Inverse Document Frequency) method. This method gives great weight to important words because they often appear in a set of reference captions, so they are considered significant. Then, a small weight was given for words that appear in general, such as conjunctions and particles in each dataset. So, CIDEr can provide a consensus assessment based on keywords that provide significant visual information from an image.

III. Results and Discussion

As described in section 1, one of this research aims to design an image captioning model that is reliable in generating stylized captions based on the SeqCapsGAN architecture. For this reason, in this study, replication of the original parameter configuration from the research conducted by Bibi, Abidi, and Dhaouadi [9] was also carried out as an equivalent comparison in this study. The configurations used in this research are feature extractor VGG-19, batch size 32, learning rate 0.001, and Adam optimizer. The following are the metrics and losses that are used as the baseline model based on the original research architecture and parameters in the experiments carried out:

TABLE I
Experimental Data Baseline Model Replication Results

Component	Generator Loss	BLEU-1	BLEU-2	BLEU-3	BLEU-4	ROUGE-L	CIDEr
Generator Pre-training	31.81	0.617	0.386	0.247	0.163	0.476	0.437
GAN Training with CNN	21.957	0.479	0.245	0.128	0.07	0.377	0.321
GAN Training with Capsule	20.97	0.492	0.268	0.151	0.088	0.389	0.356

A. *Feature Extractor*

Figure 2 shows the model with feature extractor VGG-19 has the smallest loss value, both at the generator pre-training and GAN training stages using CNN and capsule architecture. The smallest loss value from VGG-19 is 17.82826 with a combination of batch size 32, learning rate 0.001, and the Nadam optimizer in the GAN training stage with capsules. With this data, it can be indicated through experiments conducted that the model using the VGG-19 feature extractor on the SeqCapsGAN architecture has better learning capabilities than ResNet-50.

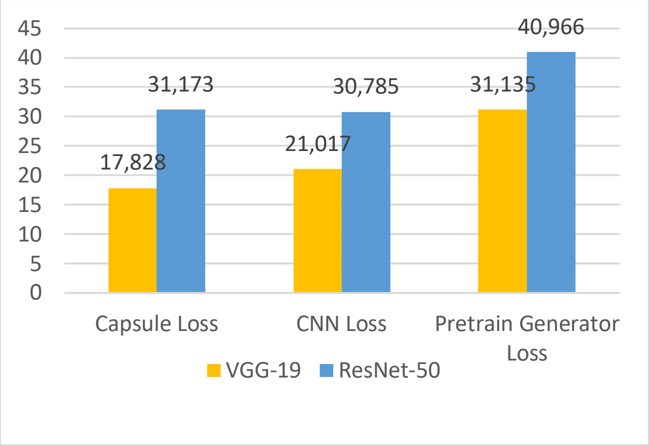


Fig. 2 Minimum Loss Feature Extractor

Based on the evaluation results of the metrics in Figure 3, VGG-19 has a higher evaluation value than ResNet-50. These results indicate that the VGG-19 feature extractor can produce captions that are both precise with BLEU metric measurements and structurally with ROUGE and CIDEr metric measurements. We also conducted a qualitative test on the captions generated by the best metric models of the two feature extractors. Captions generated by the ResNet model do not vary widely where all captions are generated the same. This indicates a mode collapse, where the generator continues to produce the same output. In comparison, the VGG-19 model provided a better caption in context and object.

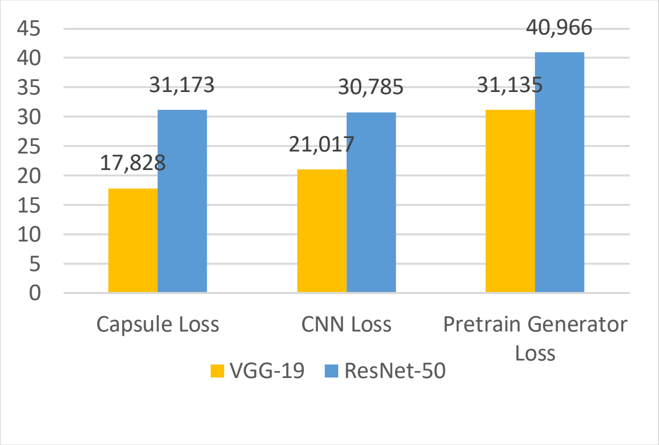


Fig. 3 Best Evaluation Metric Score per Feature Extractor GAN Capsule

B. *Optimizer*

Optimize aims to determine the effect of modifying the algorithm replacing the optimizer from the baseline model. Comparisons are made to the final value of the smallest loss generator produced by each optimizer in each process, as shown in Figure 4.

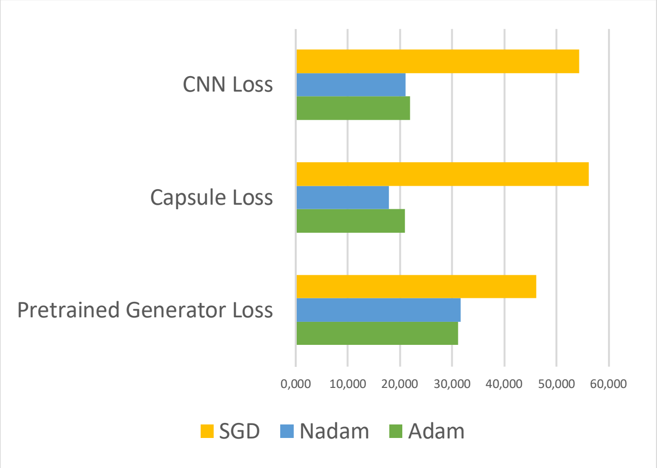


Fig. 4 Best Evaluation Metric Score per Feature Extractor GAN Capsule

From the data, out of the three optimizers used in the pre-training stage, SGD has the largest minimum loss value, with a minimum loss of only 46,126 and even increased after adversarial training. The Nadam optimizer has a better minimum loss with a significant difference, especially in adversarial training with a discriminator using a capsule network, with a loss value of 17.828. The model with the smallest loss value is the VGG-19 Nadam model with a batch size of 32 and a learning rate of 0.001. According to the initial hypothesis, the Nadam optimizer has better learning capabilities than SGD and Adam's baseline model optimizer.

Figure 4 shows that Nadam obtains a fairly low maximum value on BLEU scores. So, based on the BLEU-3 and BLEU-4 scores. It indicates that by using Nadam, the model could have a more diverse caption in terms of the number and arrangement of words in the sentence. This does not mean that the use of the Nadam optimizer makes the model unable to produce captions properly because high ROUGE-L and CIDEr values are still obtained, which represent the grammatical and semantics of the sentences.

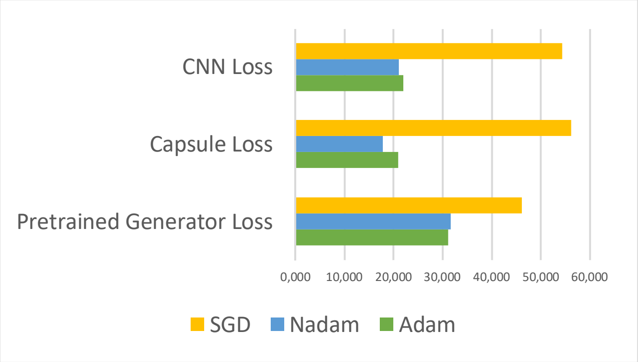


Fig. 5 Maximum Metrics for Optimizer Experiments in GAN Capsule

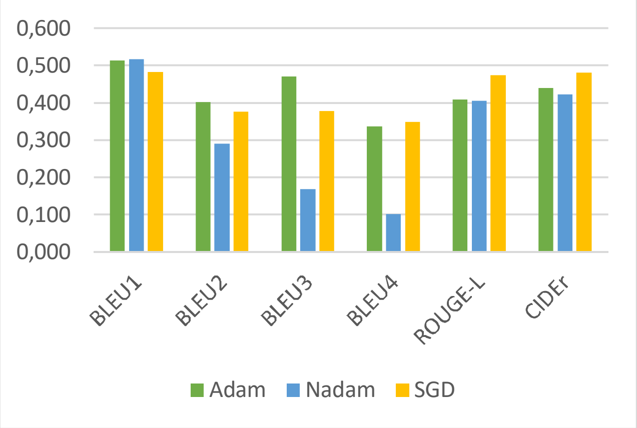


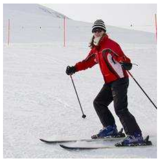
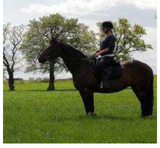
Fig. 6 Maximum Metrics for Optimizer Experiments in GAN CNN

Meanwhile, Figure 6 shows that with the CNN discriminator, the model that uses SGD as the optimizer has a slightly higher maximum metric value than Nadam & Adam, on the BLEU-4, ROUGE-L, and CIDEr metrics. This means that compared to the other two optimizers, SGD succeeded in producing captions that have a structure resembling a reference caption with a good grammatical arrangement and high similarities to the consensus reference caption. It shows that SGD, despite having a simpler method of optimizing, can produce a model with a better caption generalization at a certain score in GAN CNN settings.

C. Learning Rate

The training process results carried out with ten epochs illustrate that the model with a learning rate of 0.001 produces a smaller loss than the model with a learning rate of 0.0001. This proves that a model with a large learning rate requires fewer epochs to converge than a small learning rate. The smallest loss value of a learning rate of 0.001 is at 17,828 with a combination of batch size 32, feature extractor VGG-19, and the Nadam optimizer in the GAN training stage with capsules. On the other hand, the learning rate of 0.0001 could only reach the smallest loss value of 26,058 with a combination of batch size 8, feature extractor VGG-19, and optimizer Nadam.

TABLE II
CAPTION MODEL LR 0.001 AND 0.0001

Image	VGG19_Nadam_0.001_64	VGG19_Nadam_0.0001_64
	*Positive: a man is skiing down a rough hill in a bad parking spot* *Negative: a nice man is skiing down a beautiful mountain*	*Positive: a man in **a red jacket** is skiing down a beautiful mountain* *Negative: a man is skiing down a rough hill*
	*Positive: a man is riding a horse in the beautiful snow* *Negative: a man is riding a horse in the **cold snow***	*Positive: a man is riding a horse in a **sunny field*** *Negative: a man is riding a horse in a **sunny field***

However, the results of the metric evaluation show that a learning rate of 0.0001 provides a greater metric value than a learning rate of 0.001. The results of qualitative caption in Table 2 also show that the use of a smaller learning rate, which is 0.0001, can provide more accurate captions where the captions are more detailed and contextually precise.

D. Batch Size

The experiment results take the best model from the generator pre-training stage. It shows that 75% (6 of 8 models) of the best loss models in Table 3 are models with large batch sizes (64). This shows that a model with a large batch size can minimize loss better than a model with a small batch size.

TABLE III
BEST LOSS MODEL ON GENERATOR PRETRAINING

Feature Extractor	Optimizer	Learning Rate	Best Batch Size	Loss
VGG	Adam	0.001	64	31.135
		0.0001	8	31.258
	Nadam	0.001	64	32.178
		0.0001	8	31.621
ResNet	Adam	0.001	64	40.966
		0.0001	64	41.401
	Nadam	0.001	64	41.576
		0.0001	64	41.781

We also conduct qualitative testing, wherein the comparison of captions in Table 4 example (a), the model with a smaller batch size (batch size 16) has difficulty recognizing the main object of the image which consists of a dog and several pairs of footwear. On batch size 16, the model detects "paint" and on model 32 it gets a little better by recognizing the color "brown teddy bear" even though the object is not quite right. Then, batch size 64 was successfully recognized the object as a "brown dog". The larger the batch size, the more information the model can recognize, such as the species, colors, and objects in its environment. Another example in Figure (b) shows that the largest batch size can recognize that the train is stopping, not moving. From this, it can be seen that according to the initial hypothesis, the model with a larger batch size has higher accuracy in the ability to recognize objects in the image and their relationships with other objects in the image.

TABLE IV
LOSS MODEL ON GENERATOR PRETRAINING

Image	VGG19_Adam _0.001_16	VGG19_Adam _0.001_32	VGG19_Adam _0.001_64
a)	*a close up of a* ***cat*** *on a table*	*a large* ***brown teddy bear*** *sitting on top of a wooden bench*	*a* ***large brown dog*** *sitting on top of a wooden bench*
b)	*a train is* ***going down*** *the track near a building*	*a train is* ***traveling*** *down the tracks near a building*	*a train is* ***stopped*** *at a train station*

E. Discriminator Model

Based on the experimental results, it was found that the average loss produced by the CNN model and the capsule model did not have a significant difference, as shown in Table 5. However, the capsule model has the advantage that the minimum loss produced by the capsule model is smaller than that produced by the CNN model.

TABLE V
DISCRIMINATOR MODEL LOSS

Discriminator Model	Min Loss
Capsule	17.828
CNN	21.017

TABLE VI
COMPARISON OF CNN MODEL AND CAPSULE MODEL RESULTS CAPTION

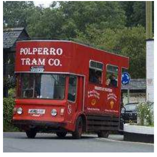
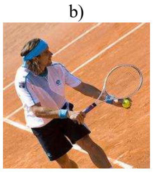
Image	VGG19_Nadam_0.001_32 _Caps	VGG19_Nadam_0.001_32 _CNN
a) 	***Positive:*** *a red and yellow bus* ***drives*** *down a pleasant street* ***Negative:*** *a red and yellow bus is* ***driving*** *down a road past a stop sign*	***Positive:*** *a red and yellow sign* ***sitting*** *on the side of a beautiful street* ***Negative:*** *a red and yellow sign* ***sitting*** *on the side of a lonely road*
b) 	***Positive:*** *a man holding a tennis racquet in front of a good crowd* ***Negative:*** *a dead man* ***holding*** *a tennis racquet on a court*	***Positive:*** *a nice man* ***swinging*** *a bat during a baseball game* ***Negative:*** *a dead man swinging a bat during a baseball game*

Table 6 shows that both models are good enough to produce captions with the right sentence structure. However, there is a slight difference in terms of object recognition, where in some cases, models with a capsule discriminator can recognize objects better. For example, in example (a), where the capsule model can recognize that the bus is moving, by considering the position of the bus in the middle of the road. This is better than the caption produced by the CNN model, which describes the parked bus. In addition, it is also seen in example (b) also shows that the use of a discriminator capsule can provide a more precise description of "holding" while the model with a CNN component identifies it as "swinging".

IV. CONCLUSION

This study implemented the Generative Adversarial Network (GAN) for image captioning using the SeqCapsGan architecture. The Capsule Network architecture has a smaller loss value and more detailed positional accurate captions than the Convolutional Neural Network (CNN) architecture used in the discriminator component. Batch size 64 can produce smaller losses compared to batch sizes 8, 16, and 32 on pre-training generators. A larger learning rate (0.001) provides a smaller loss value compared to a smaller learning rate (0.0001).

Otherwise, a smaller learning rate (0.0001) provides a more detailed and context-appropriate caption than a higher learning rate (0.001). The Nadam optimizer outperforms Adam and SGD in the success of the training and learning process, while SGD could give better generalization in BLEU-4, ROUGE-L and CIDEr metrics. The VGG-19 feature extractor used in the model generator is proven to provide smaller loss and larger metric values than the ResNet-50 feature extractor. The best combination of image captioning models in loss is the configuration of the feature extractor VGG-19, the Nadam optimizer, batch size 32, and a learning rate of 0.001. Meanwhile, the best image captioning model metrically is in the configuration of the VGG-19 feature extractor, Nadam optimizer, batch size 64, and learning rate 0.0001.

Furthermore, this research implies that further research in image recognition models used as feature extractors in image captioning context is needed. A comparative study of other state-of-the-art models and variations could be potential research to further the progress. In addition to that, automated metrics in measuring stylized image captioning are also needed to accelerate the existing evaluation method. Lastly, larger datasets suited for stylized image captioning might also be needed to create more natural and varying captions.

REFERENCES

[1] M. Z. Hossain, F. Sohel, M. F. Shiratuddin and H. Laga, "A Comprehensive Survey of Deep Learning for Image," ACM Computing Surveys, vol. 51, no. 118, pp. 1-36, Nov. 2019. Accessed on: October, 27, 2021, DOI: 10.1145/3295748, [Online].

[2] C. Gan, Z. Gan, X. He, J. Gao and L. Deng, "StyleNet: Generating Attractive Visual Captions with Styles," in Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 955-964.

[3] A. Mathews, L. Xie and X. He, "SemStyle: Learning to Generate Stylised Image Captions using Unaligned Text," in Proc. IEEE CVPR, Salt Lake City, UT, USA, 2018, pp. 8591-8600.

[4] J. D. Lannoy, "The effect of chatbot personality on emotional connection and customer satisfaction," M.Sc. Thesis Business Administration, Dept. Behav. Mgmt. Soc. Sci., Univ. Twente, Enschede, NL, 2017.

[5] A. Matthews, L. Xie and X. He, "SentiCap: Generating Image Descriptions with Sentiments," in Proc. AAAI Conf. on AI, Phoenix, AZ, USA, 2016, pp. 3574-3580.

[6] B. Dai, S. Fidler, R. Urtasun and D. Lin, "Towards Diverse and Natural Image Descriptions via a Conditional GAN," in Proc. IEEE ICCV, Venice, Italy, 2017, pp. 2989-2998.

[7] P. Dognin, I. Melnyk, Y. Mroueh, J. Ross and T. Sercu, "Adversarial Semantic Alignment for Improved Image Captions," in Proc. IEEE CVPR, Long Beach, CA, USA, 2019, pp. 10455-10463.

[8] O. M. Nezami, M. Dras, S. Wan, C. Paris and L. Hamey, "Towards Generating Stylized Image Captions via Adversarial Training" in Proc. PRICAI, Cuvu, Yanuca Isl., Fiji, 2019, pp. 270-284.

[9] A. Bibi, H. Abidi and O. Dhaouadi, "SeqCapsGAN: Generating Stylized Image Captions," 2020. Accessed on: November, 19, 2020, [Online].

[10] S. Sabour, N. Frosst and G. E. Hinton, "Dynamic Routing Between Capsules," in Proc. NIPS, Long Beach, CA, USA, 2017, pp. 3859-3869.

[11] A. Shah, E. Kadam, H. Shah, S. Shinde and S. Shingade, "Deep Residual Networks with Exponential Linear Unit," in Proc. VisionNet, Jaipur, RJ, India, 2016, pp. 59-65.

[12] T. Dozat. (May, 2016). Incorporating Nesterov Momentum Into Adam. Presented at ICLR. [Online]. Available: https://openreview.net/pdf/OM0jvwB8jIp57ZJjtNEZ.pdf

[13] P. Zhou, J. Feng, C. Ma, C. Xiong, S. HOI and W. E, "Towards Theoretically Understanding Why SGD Generalizes Better Than Adam in Deep Learning," in Proc. NEURIPS, 2020, pp. 21285-21296.

[14] D. Masters and C. Luschi, "Revisiting Small Batch Training for Deep Neural Networks," arXiv preprint arXiv:1804.07612, 2018. Accessed on: January, 8, 2021, [Online].

[15] P. M. Radiuk, "Impact of Training Set Batch Size on the Performance of Convolutional Neural Networks for Diverse Datasets," Inf. Tech. and Mgmt. Sci., vol. 20, pp. 20-24, Dec. 2017. Accessed on: December, 21, 2021, DOI: 10.1515/itms-2017-0003, [Online].

[16] Q. Fu, Y. Liu and Z. Xie, "EECS442 Final Project Report," Univ. Michigan, Ann Arbor, MI, USA, Rep. WI2020, 2019.

[17] M. Maire, S. Belongie, B. Lubomir, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, P. Dollar and T. Y. Lin, "Microsoft COCO: Common Objects in Context," in Proc. ECCV, Zurich, ZU, Switzerland, 2014, pp. 740-755.

[18] M. Arjovsky, S. Chintala and L. Bottou, "Wasserstein GAN," in Proc. ICML, Sydney, NSW, Australia, 2017, pp. 214-223.

[19] C. Lin, "Rouge: A package for automatic evaluation of summaries," in Proc. ACL Workshop, Barcelona, Spain, 2004, pp. 78-81.

[20] R. Vedantam, C. Lawrence Zitnick and D. Parikh, "Cider: Consensus-based image description," in Proc. IEEE CVPR, Boston, MA, USA, 2015, p. 4566–4575.