End-to-End Learning Using Cycle Consistency for Image-to-Caption Transformations

Keisuke Hagiwara¹, Yusuke Mukuta¹,², and Tatsuya Harada¹,²

¹The University of Tokyo
²RIKEN AIP
{hagiwara, mukuta, harada}@mi.t.u-tokyo.ac.jp

Abstract

So far, research to generate captions from images has been carried out from the viewpoint that a caption holds sufficient information for an image. If it is possible to generate an image that is close to the input image from a generated caption, i.e., if it is possible to generate a natural language caption containing sufficient information to reproduce the image, then the caption is considered to be faithful to the image. To make such regeneration possible, learning using the cycle-consistency loss is effective. In this study, we propose a method of generating captions by learning end-to-end mutual transformations between images and texts. To evaluate our method, we perform comparative experiments with and without the cycle consistency. The results are evaluated by an automatic evaluation and crowdsourcing, demonstrating that our proposed method is effective.

1. Introduction

Generating captions from images is useful, and research has been conducted to solve this problem using machine learning. For example, there are many situations where it is difficult for blind people to correctly understand images on the web. During Internet shopping, it is necessary to obtain information on images of products in audio or text form to shop as expected. In addition, image captioning is required for non-blind users. For example, when the only information a user possesses concerning the content they are interested in is an image, it would be helpful for obtaining information via a search if it is possible to input this image and acquire text for the search. Moreover, when it is necessary to utilize words only to describe a captured scene, it is necessary to convey the situation captured by the camera in text form. In addition, applications for providing assistance based on the road condition while driving a car or understanding the situation in a room can also be considered.

A caption should faithfully describe an image. In this study, if the image recalled from a generated caption is close to the original image, then the caption is considered to represent the image faithfully. Thus, we propose a method to generate captions by learning to regenerate images using mutual transformations between images and texts. Previous research has been conducted to make it possible to search an original image from a generated sentence [14]. However, in that case if only the minimum information necessary to describe the differences from other images for identification is available, then image retrieval can be considered possible. On the other hand, to generate an image from a caption generated from the input image, the caption must include information that is essential to convey the input image. In addition, the caption must make it possible to correctly understand the information when recalling an image from a caption. Therefore, an approach that attempts to generate captions from images by extracting the relationships between images and texts using a mutual transformation through machine learning can constitute a useful contribution.

In this paper, we aimed to improve the performance of captions generated from images by incorporating a learning method using mutual transformations. A generated caption must be grammatically correct, such that it is easy for humans to read, and we assume that color information is
not described for each pixel. To realize such captions, we design a generator using generative adversarial networks (GANs). In addition, mutual transformations are performed between images and texts using a cycle-consistency loss. A simple diagram of the system of our model is presented in Figure 1. There are two generators and two discriminators in this model.

The contributions of this research are summarized as follows.

- We propose a method for image captioning through training to regenerate the input image.
- The effectiveness of the proposed method is demonstrated by an automatic evaluation, and crowdsourcing to evaluate the degree to which images are described.

2. Related Work

Here, we introduce related research on image-to-text generation, text-to-image generation, and mutual transformations between different domains using GANs.

2.1. Image Captioning

Object recognition in images, understanding the relationships between objects, and the generation of grammatically correct sentences are required to generate captions from images, and research to make this possible has been conducted in recent years using machine learning. We refer the reader to the representative study of Vinyals et al. [27]. This represents an application of the encoder–decoder model, which achieves a strong performance in machine translation and uses a convolutional neural network (CNN) to encode feature from images and decode captions using a recurrent neural network (RNN). Here, long short-term memory (LSTM) [8] is employed for the RNN. In addition, methods incorporating attention have recently become mainstream [29]. When generating a caption, the first word is acquired by performing discrete sampling, such as searching for the maximum value of the appearance probability of the next word generated from the image features. Then, the next word is acquired from that word in the same manner, and so on.

In the method of Dai et al. [3], the authors combine the image captioning network of Vinyals et al. with a network for evaluating the generated captions. In the generator, the latent variable connected to an image feature extracted by the CNN is passed to LSTM. The generated caption and image are input into a network for evaluation, and the image features extracted by the CNN are multiplied by the text features extracted by LSTM to determine whether the caption matches the image. In the unsupervised image captioning method of Feng et al. [5], the authors proposed an extension to unpaired data.

2.2. GANs for Text Generation

GANs [6] consist of two parts: a generator and discriminator. In GANs, parameters of a generator are updated using gradients from a discriminator. However, because text generation usually involves indifferentiable discrete sampling, as in the study of Vinyals et al., it is not possible to propagate gradients from discriminators and train networks. With this motivation, SeqGAN was proposed by Yu et al. [30] as a method enabling the generation of text from latent variables using GANs. As for ordinary GANs, the discriminator determines whether the input text is real or fake. An RNN is employed as the generator, but reinforcement learning is utilized for training. The parameter of the generator is utilized as the policy, the result of the discriminator acts as the reward, and the parameter is updated such that the reward is maximized using the policy gradient method. Because the discriminator can only calculate the reward for data series for which generation has been completed, the remainder of a partially generated series is generated using a Monte Carlo search, and the reward is calculated approximately. Using this method, it is difficult to start learning from random parameters, owing to the use of the policy gradient method, and pre-training is required.

On the other hand, GANs for sequences of discrete elements with the Gumbel-softmax distribution (GSGAN), proposed by Kusner et al. [11], utilizes the Gumbel-softmax [9] as a generator, which is differentiable. Here, the Gumbel-softmax is as shown in Equation 1, \( \pi_1, \pi_2, ..., \pi_k \) is the probability of each class before putting it in gumbel-softmax, \( \tau \) is the temperature parameter. As \( \tau \) is smaller, the output is closer to the one-hot vector, and as \( \tau \) is larger, the output is closer to uniform distribution. Even if the text is generated in one go from the latent variable to the end, such as in the method of generating the first word from the latent variable and then generating each following word from the current word, parameter updating by error backpropagation is enabled. Because the Gumbel-softmax can output the approximate one-hot vector, it can obtain the next predicted word by approximation of the one-hot expression without performing the process of searching for the maximum value. This achieves differential sampling, and trains a generator without using reinforcement learning. However, in this study only data sequences of short length, such as lists of symbols, are generated. An additional study employs the Gumbel-softmax to generate image descriptions using GANs [23].

\[
\begin{align*}
  u & \sim \text{Uniform}(0, 1) \\
  g & = -\log(-\log(u)) \\
  z_i & = \frac{\exp((\log(\pi_i) + g_i)/\tau)}{\sum_{j=0}^{k} \exp((\log(\pi_j) + g_j)/\tau)} & \text{for } i = 1, ..., k
\end{align*}
\]
2.3. Text-to-Image Generation

The method AlignDRAW [15] was proposed by Munsimov et al. to generate images from text. This approach estimates the relationship between text features and a generated image using an RNN, but produces a blurred image that only captures rough features.

Reed et al. [22] generated images from text using GANs. The DCGAN [20] method, proposed by Radford et al., is a mainstream approach using GANs to generate images from latent variables, and employs a CNN. In addition, CGAN [16] makes it possible to generate an image that matches a condition label. Using the same network as CGAN, in the method of Reed et al. the text feature vector extracted from a caption is connected to the latent variable as a condition, and input into the generator. The discriminator determines whether the generated image is valid for the caption. A method [21] combining a CNN and RNN is also employed to extract features from captions. The model trained using this method is utilized as a text encoder, but this component is employed without updates when training GANs.

The StackGAN method of Zhang et al. [31] is another approach for generating high-resolution images. Like that of Reed et al., this method is based on CGAN, but the training is divided into two stages. An image satisfying the conditions is generated in the first stage, and a high-resolution image is generated in the second stage. Here, if the text feature of the caption is employed, then the input vector is biased, and learning becomes unstable. To circumvent this issue, to maintain diversity the authors employ a condition sampled randomly from the normal distribution, whose average and diagonal covariance matrix are calculated from the text feature. In addition, to avoid overfitting the Kullback–Leibler divergence is incorporated into the loss function, representing the distance between the standard normal distribution and the normal distribution obtained from the feature. An additional study enables the generation of high-resolution images by end-to-end learning without division into two stages [32].

2.4. Transformation Using the Cycle Consistency

The CycleGAN [33] method, proposed by Zhu et al., employs GANs for mutual transformations between images. The authors combined two GANs, and enabled color conversions of images such as horses and zebras, and summer and winter landscapes. In CycleGAN, let X and Y represent the image domains, such as summer and winter. Then, we have four networks: the X to Y generator $G_Y$, discriminator of Y $D_Y$, Y to X generator $G_X$, and discriminator of X $D_X$. In this model, the data of the domain Y is generated from that of the domain X, and the data of X is regenerant from the generated data. The data of Y is generated from that of Y, and the data of Y is regenerant from the generated data in the same manner. CycleGAN utilizes two types of loss function for learning. The reason for employing two is that they combine two GANs. One is the same as for regular GANs. The other is called the cycle-consistency loss, and represents the error between the original data and the result obtained by passing this through the two generators. This is treated as a loss function for the generators, and is not used to train the discriminators. In this manner, CycleGAN enables mutual transformations of images, even for datasets that do not contain pair data. Text2image2textGAN, proposed by Gorti et al. [7], is an example of using the cycle-consistency to generate images from text. Here, an image is generated from a text, a text is generated from the resulting image, and learning is performed using the consistency loss between the regenerated and original texts. The model of Vinyals et al. is employed in the image-to-text generator, and so differentiable sampling as in GSGAN is not performed.

3. Proposed Model

Our model performs mutual transformations between images and captions with reference to CycleGAN. The entire pipeline is shown in Figure 2. Unlike CycleGAN, which performs transformations between images, pre-training is performed to handle texts consisting of series data, so that a grammatically correct description and an image without noise can be generated. We explain each procedure in detail below.

3.1. Image-to-Caption GAN

In GANs that generate captions from images, two types of generator are prepared. These generators share the same network parameters.

One is employed to generate captions from images in the dataset. This generator is shown in Figure 2(a). Incorporating the method of GSGAN, we utilized the Gumbel-softmax to generate the text from an image feature. This is because a further image is generated from the generated caption, and error backpropagation can be performed during learning.

The other generator is employed to generate a caption from a generated image. This generator is shown in Figure 2(b), and is based on the method of Vinyals et al. Here, the trained model of VGG16 [24] was employed as a CNN for extracting features of images. An image is input into the trained model, and pooling is performed in the fifth layer. The output of this is then input into LSTM through the linear layer, and the features are extracted by the hidden vector of LSTM. The features are utilized as the first LSTM hidden vector of the caption generation component, the word representing the beginning of the sentence is employed as the first input, and the cross entropy of the probability distribution of the next word to be predicted and the ground truth are calculated. The first word of the ground truth is adopted
as the next input, and cross entropy of the probability distribution of the next word to be predicted and the ground truth are again calculated. The generator is pre-trained by calculating the sum of the cross entropies until the end of the sentence in the same manner.

A caption and an image are input into the discriminator. The captions generated by the generator are approximated as one-hot expressions by the Gumbel-softmax, and the sentences of the ground truth are represented by one-hot expressions. By multiplying the image features extracted by the trained VGG16 model with the caption features extracted by LSTM, it is determined whether captions match images. To prevent the discriminator from becoming overly strong, it is not pre-trained.

When training both generators and classifiers, the parameters of the VGG16 component are not updated, as these are contained in the trained model.

3.2. Caption-to-Image GAN

We designed the training of GANs that generate images from captions based on the first stage generator and discriminator of StackGAN. The loss function of the generator is the same as that of StackGAN, such that the amount of Kullback–Leibler information is added to the loss function of the normal generator of GANs. As with StackGAN, the discriminator also employs either the generated image or correct image from the dataset and the correct text features as input to determine whether an image matches the text. Here, the convolution and linear combination layers from the SNGAN [17] method are adopted, such that spectral normalization is performed in the discriminator to stabilize the learning.

Next, the resulting caption is input into a text encoder to extract features and generate an image from these features. In the method of Reed et al. and StackGAN, the text encoder generates features for captions in the first stage of the StackGAN model. These features are upsampled to the size of the input image. The patch features are extracted from the input image and the features of the input image are extracted by the text encoder. These are then processed using the generator and discriminator of the first stage of StackGAN, respectively. The loss of the first stage is determined by calculating the sum of the cross entropies until the end of the sentence in the same manner.
encoder employs a learned model, and parameters of the model are not updated while training for image generation. In this study, bi-directional LSTM [2] is employed as the text encoder, to extract the forward and backward features of the caption and connect them. While the effects of words increase at the end of the text in ordinary LSTM, the effects of words at the beginning of text can also be increased by employing bi-directional LSTM. This text encoder was trained by the generator using its loss function.

The generator, discriminator, and text encoder were pre-trained.

3.3. Image-to-Caption Mutual Transformation

For the mutual transformation, it is necessary to calculate the cycle-consistency loss from the results through the two generators and the original input, and to utilize the loss for learning. We used two types of L1 loss, for the cycle-consistency loss between images (Figure 2 (a)) and the cross entropy for the cycle-consistency loss between captions (Figure 2 (b)).

Images should not only be directly compared on a pixel-by-pixel basis, but the contents of the images should also be compared. Therefore, the L1 loss calculated from a direct image and that calculated from the value of the intermediate layer passing the image through VGG16 were incorporated. Specifically, we employed the calculation results from up to the fifth pooling layer. In addition, weighting was performed using hyper parameters, so that the effect of the L1 loss calculated through VGG16 was greater than that of the directly calculated L1 loss. The technique of comparing images through the image encoder and calculating the loss is also employed in SRGAN [12], making it possible to prevent blurry outputs, such as those occurring when averaging pixel values when comparing images directly.

For the cycle-consistency loss between captions, we did not employ the generated text as it was, but rather we used the vector before passing the Gumbel-softmax, which represented the probability distribution of the predicted word, and calculated the cross entropy using the captions of the dataset.

3.4. Objective Function

Equation 2 defines the objective function of GANs that generates captions from images. Here, the domain X is an image and Y is a text. Furthermore, $G_Y$ and $D_Y$ are the generator and discriminator of captions, respectively.

$$\max_{D_Y} L_{D_Y} = E_{(x,y) \sim p_{data}(x,y)}[\log D_Y(y,x)] + E_{z \sim p_z, y \sim p_{data}(y)}[\log(1 - D_Y(G_Y(z), x))]$$

(2)

Equation 3 defines the objective function of GANs that generates images from captions. Here, $G_X$ and $D_X$ are the generator and discriminator of images, respectively. Furthermore, $\varphi_Y$ denotes features extracted by the text encoder. The text encoder component is trained using only the generator’s loss function. Here, the hyper parameter was set as $\lambda_{KL} = 2.0$.

$$\max_{D_X} L_{D_X} = E_{(x,y) \sim p_{data}(x,y)}[\log D_X(x, \varphi_Y)] + E_{z \sim p_z, y \sim p_{data}(y)}[\log(1 - D_X(G_X(z, \varphi_y), \varphi_y))]$$

$$\min_{G_X} L_{G_X} = E_{z \sim p_z, y \sim p_{data}(y)}[\log(1 - D_X(G_X(z, \varphi_y), \varphi_y))] + \lambda_{KL} KL(D_{cyc}(N(\mu(\varphi_y), \Sigma(\varphi_y)||N(0, 1)))$$

(3)

Finally, Equation 4 defines the objective function of the cycle-consistency loss. Here, $F_{IE}$ is the image encoder, and we employed VGG16. Furthermore, $y_t$ is the t-th word in a caption and $T$ is the length of a caption. Here, the hyper parameters were set as $\lambda_1 = 1.0, \lambda_2 = 1.000$, and $\lambda_3 = 0.01$.

$$\min_{G_Y, G_X} L_{cyc} = E_{x \sim p_{data}(x)}[\lambda_1 \| G_X(G_Y(x)) - x \|_1] + E_{x \sim p_{data}(x)}[\lambda_2 \| F_{IE}(G_X(G_Y(x))) - F_{IE}(x) \|_1]$$

$$+ E_{y \sim p_{data}(y)}[-\lambda_3 \sum_{t=0}^{T} y_t \log(G_Y(G_X(y_t)))]$$

(4)

In summary, the objective function of the proposed method is defined as in Equation 5.

$$\max_{D_Y, D_X} V_{D_Y, D_X} = L_{D_Y} + L_{D_X}$$

$$\min_{G_Y, G_X} V_{G_Y, G_X} = L_{G_Y} + L_{G_X} + L_{cyc}$$

(5)

4. Experiments

Here, we describe an experiment performed to generate a caption from an image, and the generated caption is evaluated.

4.1. Datasets

We employed a dataset combining Caltech-UCSD Bird [28], consisting of bird images, and Oxford-102 Flower [13], consisting of flower images. The dataset contains attached captions, and was provided by Reed et al. [21]. Unlike in CycleGAN, this is a paired dataset. Because this dataset is limited to birds and flowers, it is often employed to study image generation from text. Therefore, we considered it to be appropriate for this study. Some examples are presented in Figure 3.
Birds

the medium sized bird has a dark grey color, a black downward curved beak, and long wings.
the bird is dark grey brown with a thick curved bill and a flat shaped tail.
bird has brown body feathers, white breast feathers and black beak
this bird has a dark brown overall body color, with a small white patch around the base of the bill.
the bird has very long and large brown wings, as well as a black body and a long black beak.
it is a type of albatross with black wings, tail, back and beak, and has a white ring at the base of its beak.
this bird has brown plumage and a white ring at the base of its long, curved brown beak.
the entire body is dark brown, as is the bill, with a white band encircling where the bill meets the head.
this bird is gray in color, with a large curved beak.
a large gray bird with a long wingspan and a long black beak.

Flowers

this flower is blue and green in color, with petals that are oval shaped.
the petals of the flower are bright blue with white lines, and the stamen is bright green with black dots.
the flower has stamen and the petals are green in color.
this flower has green petals and purple and green stamen.
the flower shown has green pollen tubes with green sepal and blue petals
a large flower with neon colors and a large green stigma.
this blue and white flower has pointed petals and green sepal.
this flower has a flat row of pointed white petals and a flat row of thin blue filament on top of that.
this flower has petals that are green and has purple stringy stamen.
this flower has large green petals under a fringed set of purple and white quills.

Figure 3. Some examples of the dataset provided by Reed et al.

Here, 10 captions were attached to each image using Amazon Mechanical Turk (AMT) [1]. Each of these 10 sentences is independent. Workers were instructed not to mention background, so the information that said it was in the sea or forest was not included in the captions.

There are a total of 11,788 images of birds in 200 classes, and a total of 8,188 images of flowers in 102 classes. Of these, 90% were utilized as training data and 10% as test data. Specifically, 10,609 training and 1,179 test data items were from the bird dataset, and 7,369 training and 819 test data items were from the flower dataset.

Here, the lengths of the captions included in each dataset are shown in Figure 4. The bird and flower datasets vary in length.

Therefore, to make the lengths of the utilized captions uniform, one of the 10 captions for each image was selected to be utilized according to the following method.

- Remove periods, commas, and semicolons from the captions.
- Align all captions to 20 words in length by filling the end of the caption with a symbol indicating the end of the sentence or deleting the 21st and later words.
- During training, select one of the 10 captions at random for each iterator (however, the caption shall be limited to five or more words except for the sentence-ending symbol).

The resulting captions of the dataset were utilized in the experiment. In addition, because the bird dataset was provided with bounding-box information indicating the part of the image where the bird appears, this part was cropped and utilized. The images were RGB representations of size $64 \times 64$.

4.2. Experimental Setup

We generated captions of 20 words in length from $64 \times 64$-sized images. When a caption was input into the text encoder, consisting of bi-directional LSTM, the text features were extracted as a 1,024-dimensional vector, in which forward vector and backward vectors of 512 dimensions were connected, and size-$64 \times 64$ image was generated from the text features and a 100-dimensional latent variable according to the normal distribution. The training consisted of 500 epochs in pre-training and 200 in subsequent training. Adam [10] and weight decay were employed for optimization during the whole training period, including pre-training.

As a comparison method, image captioning and image regeneration were performed by combining $G_X$ and $G_Y$, which were trained the same number of times as the proposed method, without using the cycle-consistency loss.

4.3. Qualitative Results

Some successful examples from this experiment are presented in Figure 5. The captions were generated using the proposed method from images in the dataset for test. When the symbol indicating the end of the sentence was output, words after that were deleted. They correctly describe the colors of birds or flowers overall, and also mention the colors of beaks or shapes of petals. Furthermore, the captions
Figure 4. Lengths of captions in the dataset. The upper and lower parts represent the bird and flower data, respectively.

Table 1. Automatic evaluation of generated captions

|                  | BLEU-4 | ROUGE | Meteor | CIDEr |
|------------------|--------|-------|--------|-------|
| Ours(bird)       | 0.057  | 0.304 | 0.146  | 0.218 |
| Comp.(bird)      | 0.047  | 0.304 | 0.147  | 0.207 |
| Ours(flower)     | 0.070  | 0.294 | 0.146  | 0.212 |
| Comp.(flower)    | 0.062  | 0.287 | 0.141  | 0.190 |

are also grammatically correct, at a level that can be understood by humans.

4.4. Caption Evaluation

As a quantitative evaluation method for the captions generated from the test data, automatic metrics such as BLEU [19], ROUGE [13], Meteor [4], and CIDEr [26] are employed. The results are shown in Table 1. For both the bird and flower datasets, the proposed method achieved a comparable or better performance than the comparison method. In particular, the score improvement in BLEU-4 and CIDEr was remarkable.

4.5. Human Evaluation

To determine whether faithful captions were generated, a manual evaluation was also performed by crowdsourcing using AMT. Given generated caption pairs (from the proposed and comparison methods) together with an image from the dataset, workers were asked to select which caption better represents the image. Here, we instructed the workers to give priority to judging the extent to which the caption refers to the image, rather than focusing on grammatical order. The data employed here consisted of 100 randomly selected items from the test data, and five workers completed the same task. The workers were all certified as Master by AMT. The points obtained by summing the selections of each option for the proposed and comparison methods were used for the evaluation. In addition, binomial tests were performed on the evaluation results, to determine whether the differences between the proposed and comparison methods were significant. Here, it was tested whether the proportion selecting the proposed method was significantly different from 0.5, with a significance level of 0.05. The results are shown in Table 2. For both datasets, the proposed method achieved higher points, and the statistical tests show a significant difference between the proposed and comparison methods. (Each p value was less than 0.05.)

4.6. Regenerated Image Evaluation

As a quantitative evaluation method for regenerated images, the fine-tuned inception model [31] of StackGAN was employed, and the inception score was calculated. Specifically, we employed the fine-tuned inception model of Zhang et al. [25] for Caltech-UCSD Bird and Oxford-102 flower. The results are shown in Table 3. The overall results are not considerably different, and for the flower dataset the score was lower for the proposed method than for the comparison method.

4.7. Extension to unpaired data

The dataset used in this experiment is used as a non-pair data set by randomly selecting image and description pairs. Learning was also performed using non-paired data sets in
this flower is pink in color and has petals that are oval shaped and lighter at the edges

this bird is yellow in color with a black beak and black eye rings

dark gray bird with black strips on its nape and a white belly

this bird is brown with white and has a very short beak

this flower is white and purple in color with petals that are thin and spotted

Figure 5. Examples of successes. Each caption on the right is generated from the image on the left.

|                          | mean | std. |
|--------------------------|------|------|
| **Table 3. Inception scores of regenerated images** |      |      |
| Ours(bird)               | 1.46 | 0.14 |
| Comp.(bird)              | 1.38 | 0.10 |
| Ours(flower)             | 1.26 | 0.07 |
| Comp.(flower)            | 1.34 | 0.08 |

|                          | BLEU-4 | ROUGE | Meteor | CIDEr |
|--------------------------|--------|-------|--------|-------|
| **Table 4. Automatic evaluation of generated captions(unpaired)** |       |       |        |       |
| Ours(bird)               | 0.056  | 0.308 | 0.149  | 0.216 |
| Comp.(bird)              | 0.048  | 0.299 | 0.144  | 0.197 |
| Ours(flower)             | 0.065  | 0.295 | 0.142  | 0.185 |
| Comp.(flower)            | 0.064  | 0.282 | 0.145  | 0.181 |

In this study, we proposed a learning method for image captioning using mutual transformations between images and texts. We demonstrated that mutual transformations are possible using the cycle-consistency loss. Furthermore, automatic evaluations demonstrated that a comparable or better performance was achieved by the proposed method with paired dataset. In addition, crowdsourcing evaluations also demonstrated that our method is more effective than one without the cycle-consistency loss.

In order to improve performance more, it is necessary to increase the accuracy of GANs used for mutual conversion. In particular, GANs that generate images matching text from text are considered to be difficult to achieve higher accuracy than GANs that generate text from images in terms of the amount of information. Also, in this paper, we did not consider where to pay attention to images and captions. During learning, two types of loss function can interfere with parameter updates of each other. In order to avoid that, it is effective to find something better than the one used in this paper as a cycle-consistency loss between images.

In order to know the effect on non-paired data sets, it is considered that datasets with more domains to handle are required.

6. Acknowledgement

This work was partially supported by JST CREST Grant Number JPMJCR1403, Japan. We would like to thank Mikihiro Tanaka, Atsuhiro Noguchi, Hiroaki Yamane for helpful discussions.
References

[1] https://www.mturk.com.

[2] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In International Conference on Learning Representations (ICLR), 2015.

[3] B. Dai, S. Fidler, R. Urtasun, and D. Lin. Towards diverse and natural image descriptions via a conditional gan. In the IEEE International Conference on Computer Vision (ICCV), 2017.

[4] M. Denkowski and A. Lavie. Meteor universal: Language specific translation evaluation for any target language. In the ninth workshop on statistical machine translation, 2014.

[5] Y. Feng, L. Ma, W. Liu, and J. Luo. Unsupervised image captioning. arXiv preprint arXiv:1811.10787, 2018.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Neural Information Processing Systems (NeurIPS), 2014.

[7] S. K. Gorti and J. Ma. Text-to-image-to-text translation using cycle consistent adversarial networks. arXiv preprint arXiv:1808.04538, 2018.

[8] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

[9] E. Jang, S. Gu, and B. Poole. Categorical reparameterization with gumbel-softmax. arXiv preprint arXiv:1611.01144, 2016.

[10] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In International Conference on Learning Representations (ICLR), 2015.

[11] M. J. Kusner and J. M. Hernández-Lobato. Gans for sequences of discrete elements with the gumbel-softmax distribution. arXiv preprint arXiv:1611.04051, 2016.

[12] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[13] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In Text Summarization Branches Out: Proceedings of the ACL-04 Work- shop, 2004.

[14] R. Luo, B. Price, S. Cohen, and G. Shakharovich. Discriminability objective for training descriptive captions. In the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[15] E. Mansimov, E. Parisotto, J. L. Ba, and R. Salakhutdinov. Generating images from captions with attention. In International Conference on Learning Representations (ICLR), 2016.

[16] M. Mirza and S. Osindero. Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784, 2014.

[17] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In International Conference on Learning Representations (ICLR), 2018.

[18] M.-E. Nilsback and A. Zisserman. A visual vocabulary for flower classification. In the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006.

[19] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In the 40th annual meeting on association for computational linguistics (ACL), 2002.

[20] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In International Conference on Learning Representations (ICLR), 2016.

[21] S. Reed, Z. Akata, H. Lee, and B. Schiele. Learning deep representations of fine-grained visual descriptions. In the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[22] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee. Generative adversarial text to image synthesis. In International Conference on Machine Learning (ICML), 2016.

[23] R. Shetty, M. Rohrbach, L. Anne Hendricks, M. Fritz, and B. Schiele. Speaking the same language: Matching machine to human captions by adversarial training. In the IEEE International Conference on Computer Vision (ICCV), 2017.

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In International Conference on Learning Representations (ICLR), 2014.

[25] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In the IEEE conference on Computer Vision and Pattern Recognition (CVPR), 2016.

[26] R. Vedantam, C. Lawrence Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In the IEEE conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[27] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In the IEEE conference on Computer Vision and Pattern Recognition (CVPR), 2015.

[28] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[29] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In International Conference on Machine Learning (ICML), 2015.

[30] L. Yu, W. Zhang, J. Wang, and Y. Yu. Seggan: Sequence generative adversarial nets with policy gradient. In Association for the Advancement of Artificial Intelligence (AAAI), 2017.

[31] H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In the IEEE International Conference on Computer Vision (ICCV), 2017.

[32] Z. Zhang, Y. Xie, and L. Yang. Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

[33] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial net-
works. In *the IEEE International Conference on Computer Vision (ICCV)*, 2017.