Article

Realistic Image Generation from Text by Using BERT-Based Embedding

Sanghyuck Na ¹ , Mirae Do ², Kyeonah Yu ² and Juntae Kim ^{1,*} ¹ Department of Computer Science and Engineering, Dongguk University, Pildong-ro 1-gil, Jung-gu, Seoul 04620, Korea; shna@dongguk.edu² Department of Computer Engineering, Duksung Women's University, Samyang-ro 144-3gil, Dobong-gu, Seoul 01369, Korea; wwwdo109@naver.com (M.D.); kyeonah@duksung.ac.kr (K.Y.)

* Correspondence: jkim@dongguk.edu; Tel.: +82-2-2260-3712

Abstract: Recently, in the field of artificial intelligence, multimodal learning has received a lot of attention due to expectations for the enhancement of AI performance and potential applications. Text-to-image generation, which is one of the multimodal tasks, is a challenging topic in computer vision and natural language processing. The text-to-image generation model based on generative adversarial network (GAN) utilizes a text encoder pre-trained with image-text pairs. However, text encoders pre-trained with image-text pairs cannot obtain rich information about texts not seen during pre-training, thus it is hard to generate an image that semantically matches a given text description. In this paper, we propose a new text-to-image generation model using pre-trained BERT, which is widely used in the field of natural language processing. The pre-trained BERT is used as a text encoder by performing fine-tuning with a large amount of text, so that rich information about the text is obtained and thus suitable for the image generation task. Through experiments using a multimodal benchmark dataset, we show that the proposed method improves the performance over the baseline model both quantitatively and qualitatively.

Keywords: text to image generation; multimodal data; BERT; GAN

Citation: Na, S.; Do, M.; Yu, K.; Kim, J. Realistic Image Generation from Text by Using BERT-Based Embedding. *Electronics* **2022**, *11*, 764. <https://doi.org/10.3390/electronics11050764>

Academic Editor: George A. Papakostas

Received: 23 November 2021

Accepted: 11 February 2022

Published: 2 March 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Although many deep learning methods have been developed for single modality, the real world we experience is multimodal, so research on multimodal deep learning is essential for AI to make meaningful progress [1]. Text-to-image generation is a representative example of multimodal learning [2,3]. Many images exist with tags or descriptions of them, and the text serves to clarify the meaning of the image [4]. Text-to-image generation is a mixture of two modalities, text and image, which are the most challenging modalities in deep learning, but it is difficult to learn because text as input and image as output have completely different characteristics. There are three problems that need to be addressed in order to generate images from text. First, you need to learn a text representation that captures what is visually important. Second, it is necessary to generate high-quality images similar to real images through text feature representation. Third, it is necessary to extract high-quality feature expressions for texts that have not been seen during learning. Therefore, the text-to-image generation model consists of a text encoder part for text embedding and a GAN part that generates an image using it. It is important to effectively learn the task of generating various images corresponding to the semantic information of the text.

In this paper, we propose a text-to-image model using BERT-based embedding and high-quality image generation using StackGAN. Existing text-to-image generation studies have a problem of creating empty spaces between data in the text manifold by using a pre-trained text encoder for a zero-shot visual recognition task. In this paper, we try to solve this problem by fine-tuning the pre-trained BERT to be suitable for the text-to-image

generation task. When the fine-tuned BERT is used as a text encoder, there is little space between data in the text manifold, so text representation can be effectively extracted, and it is shown that it is possible to generate a more realistic image compared with existing studies using efficient embedding. Through experiments, the proposed method shows qualitative and quantitative performance improvement compared with the existing methodologies on the CUB multimodal benchmark.

2. Related Work

A generative adversarial network (GAN) is a deep neural network model consisting of a generator network and a discriminator network [5]. The generator neural network takes a random vector or a vector extracted from the latent space as input, and generates data such as images, audio, and text. The discriminator neural network discriminates between real data and fake data generated from the generator neural network. The generator and discriminator neural networks are trained simultaneously in competition with each other. As a result, we can get a generator neural network that generates data similar to real data.

For the generation of image data, convolutional neural network [6] has been introduced to the GAN model. Radford et al. [7] proposed a family of network architectures called deep convolutional GAN (DCGAN), which allows training a pair of deep convolutional generator and discriminator networks. DCGANs make use of convolutions, which allow the spatial downsampling and upsampling operators to be learned during training. Conditional GAN (CGAN) is a generative adversarial network that inputs a vector extracted from latent space and conditional information [8]. By using the conditional information, CGANs can generate data of a desired class. CGANs also have the advantage of being able to provide better representations for multimodal data generation. Image-to-image translation, image style transfer, and photo-realistic image generation using CNN and GAN model are also related to this research [9–12].

In text-to-image(T2I) tasks, not only should the image capture all the content of the given text description, but also the quality of the generated image should be good. In order to satisfy this condition, various T2I models have been proposed. The first GAN-based T2I model is GAN-INT-CLS [13] where a class label of an image is simply replaced by a text embedding in CGAN. GAN-INT-CLS has shown that it can generate images but cannot guarantee the quality of images synthesized using text encoders that have learned the relationship between labeled images and text in advance. Since it is difficult to generate high-resolution images with end-to-end direct learning, StackGAN, a two-stage approach of sketch and refinement, has been proposed [14]. StackGAN consists of the first stage of generating a low-resolution image and the second stage of generating a high-resolution image by integrating the low-resolution image and text. The conditioning augmentation method is applied to compensate for discontinuities in high-dimensional embeddings that hinder the learning of the image-generating part. Experimental results showed that it produces high-resolution images of 256×256 , later this method was developed to generate 512×512 images using hierarchical discriminators HDGAN [15].

In the natural language processing (NLP) field, many studies using large amounts of text are being conducted [16,17]. Especially BERT is a language model using only encoders from the Transformer architecture, and is used to obtain embeddings, which are vector representations of text for natural language processing and machine translation [18]. BERT achieves the best performance in most tasks of GLUE [19], a natural language understanding benchmark by generating context-sensitive embeddings, and is widely used in the field of natural language processing. Because BERT can be trained with a relatively small amount of domain data, BERT is used not only in natural language processing fields such as machine translation, but also in various fields requiring text embedding. In [20], BERT was used to draw a face from a text description. The advantage of BERT-based embedding is that it can work for relatively long text descriptions and that it is possible to

learn with a small amount of face data by using pre-trained BERT. In [21], BERT was also used for supervised image generation work. In this paper, pre-trained BERT is used for text embedding without fine tuning.

3. Realistic Image Generation from Text by Using BERT-Based Embedding

The realistic image generation model from text using BERT-based embedding proposed in this paper utilizes the structure of the stack generative adversarial network (StackGAN). Therefore, the proposed model consists of (1) BERT-based text embedding, (2) low-resolution image generation from text using BERT-based embedding, and (3) realistic high-resolution image generation with text using BERT-based embedding as shown in Figure 1.

Figure 1. Generating realistic images from text using BERT-based embeddings.

In the text embedding process, pre-trained BERT is used, and fine-tuning is performed on the target dataset. A low-resolution image is generated by using the condition-augmented embedding and a random vector as inputs to the generator, and the discriminator performs learning by discriminating the low-resolution image and the actual image generated along with the text embedding. The process of generating realistic high-resolution images from text using BERT-based embedding corresponds to stage 2 in Figure 1. In the process, condition augmentation is also performed using BERT-based embedding in Figure 1. A high-resolution image is generated by using the condition-augmented text embedding and the low-resolution image generated from the learned generator of stage 1 as an input to the generator, and the discriminator performs learning by discriminating the high-resolution image and the actual image generated with text embedding.

The 3 steps of our proposed model can be summarized as followings:

- (1) Fine-tune the pre-trained BERT using the target dataset to obtain text embedding.

- (2) In stage 1, generate a low-resolution image as the input for the text embedding from fine-tuned BERT and random noise.
- (3) In stage 2, generate a high-resolution image as the input for the text embedding from fine-tuned BERT and the low-resolution image from stage 1.

3.1. BERT-Based Text Embedding

In this paper, pre-trained BERT is used to obtain embedding vectors for text descriptions. The pre-trained BERT was fine-tuned by creating blanks in the sentences for the target dataset as much as the number of repetitions.

Algorithm 1 describes the procedure of fine-tuning the BERT. The pre-trained BERT θ_{BERT} is initialized in the first line of the algorithm. In the second line S_f is the number of iterations for fine-tuning and $T = \{t_1, t_2, \dots, t_n\}$ is the text description of the dataset. In the third and fourth lines, the BERT is fine-tuned for the description of the data set using Adam repeatedly S_f times. During fine-tuning, the MLM (masked language modeling) task, which is a semi-supervised learning method, is learned for the given text description. The MLM task is a blank insertion problem, and it is a task to cover a part of the input token and match it. Through this, we learn the probability of token appearance considering the bidirectional context. The result of Algorithm 1 is a text embedding to be used in performing image operations with text.

Algorithm 1 Fine-tuning of BERT

- 1: Initialize parameters pre-trained BERT θ_{BERT}
 - 2: Set number of fine-tuning step S_f , descriptions of datasets $T = \{t_1, t_2, \dots, t_n\}$
 - 3: **For** $f = 1, \dots, S_f$: **do**
 - 4: Perform step of adam update on θ_{BERT} with T
 - 5: **End for**
-

3.2. Generating Low-Resolution Images from Text

The stage1 generator produces a low-resolution image that matches only the rough shape and color by using the text embedding vector and random vector extracted from the text encoder. The discriminator uses an image and a text embedding vector as inputs and determines whether the text matches the image or not.

Let D_1 be the discriminator of stage1, G_1 the generator, h the text embedding of given text description through fine-tuned BERT, h_{ca} the text embedding extracted from the Gaussian conditioning variable, x_{row} the low-resolution image, z the random vector, and λ the regularization parameter. The stage 1 discriminator is trained to maximize Equation (1), and the stage1 generator is trained to minimize the loss function of Equation (2). $D_{KL}(\mathcal{N}(\mu_0(h), \Sigma_0(h)) \parallel (0, X_{row}))$ from Equation (2) is the regularization term which is the KL divergence, which extracts latent text representation from an independent Gaussian distribution. The Gaussian conditioning variable h_{ca} is sampled from $\mathcal{N}(\mu_0(h), \Sigma_0(h))$ to provide randomness.

$$\mathcal{L}_{D_1} = \mathbb{E}_{(x_{row}, h) \sim p_{data}} [\log D_1(x_{row}, h)] + \mathbb{E}_{z \sim p_z, h \sim p_{data}} [\log(1 - D_1(G_1(z, h_{ca}), h))] \quad (1)$$

$$\mathcal{L}_{G_1} = \mathbb{E}_{z \sim p_z, h \sim p_{data}} [\log(1 - D_1(G_1(z, h_{ca}), h))] + \lambda D_{KL}(\mathcal{N}(\mu_0(h), \Sigma_0(h)) \parallel \mathcal{N}(0, X_{row})) \quad (2)$$

Algorithm 2 presents the process of generating a low-resolution image from text using BERT-based embedding. This corresponds to stage 1 of Figure 1. In line 2, $\theta_{BERT}^{fine-tuned}$, which is the fine-tuned BERT model and the other parameters such as the number of iterations S_1 , and the learning rates α and β of the generator and discriminator are set. $T = \{t_1, t_2, \dots, t_n\}$ is the text descriptions $X = \{x_1, x_2, \dots, x_n\}$ is the images of dataset. In the for loop, the mini-batch image x_i and the text t_i are extracted and learning is performed

for S_1 times. A random vector is extracted in line 5, and the text embedding is obtained by taking the mini-batch text as the input of the BERT in line 6. In line 8, h_{ca} is obtained through conditional augmentation, and in line 10, a low-resolution image is generated by input with a random vector. Thereafter, the discriminator determines whether the image and the text match, and proceeds with the process of updating the parameters of the generator and the discriminator in line 14 and 15, respectively.

Algorithm 2 Low-Resolution Image Generation from Text by Using BERT-Based Embedding

```

1: Initialize parameters of stage1 generator and discriminator  $G_1, D_1$ 
2: Set fine-tuned BERT  $\theta_{BERT}^{fine-tuned}$ , number of stage1 step  $S_1$ , and  $G_1, D_1$  learning rates  $\alpha, \beta$ 
   text descriptions and images of dataset  $T = \{t_1, t_2, \dots, t_n\}$ ,  $X = \{x_1, x_2, \dots, x_n\}$ 
3: For  $step = 1, \dots, S_1$  do
4:   For  $i = 1, \dots, n$  do
5:      $z \sim \mathcal{N}(0, 1)$  # sample of random noise
6:      $h \leftarrow \theta_{BERT}^{fine-tuned}(t_i)$  # sample of matching text description
7:      $h' \leftarrow \theta_{BERT}^{fine-tuned}(t'_i)$  # sample of miss – matching text description
8:      $h_{ca} \leftarrow ca(h)$  # conditioning augmentation of matching text description
9:      $h'_{ca} \leftarrow ca(h')$  # conditioning augmentation of miss – matching text description
10:     $\hat{x}_{row} \leftarrow G_1(z, h_{ca})$  # generated image from text description
11:     $s_{row}^r \leftarrow D_1(x_{row}, h_{ca})$  # real image, right text description
12:     $s_{row}^w \leftarrow D_1(x_{row}, h'_{ca})$  # real image, wrong text description
13:     $s_{row}^w \leftarrow D_1(\hat{x}_{row}, h_{ca})$  # fake image, right text description
14:     $D_1 \leftarrow D_1 - \alpha \partial \mathcal{L}_{D_1} / \partial D_1$  # update stage1 discriminator  $\mathcal{L}_{D_1}$  = equation (1).
15:     $G_1 \leftarrow G_1 - \beta \partial \mathcal{L}_{G_1} / \partial G_1$  # update stage1 Generator  $\mathcal{L}_{G_1}$  = equation (2).
16:   End for
17: End for

```

3.3. Generating High-Resolution Images from Text

In stage 2, a realistic high-resolution image is produced by supplementing omitted details to the image generated in stage 1. It takes a low-resolution image and the text embedding fine-tuned on BERT. Let D_2 be the stage 2 discriminator and G_2 the stage 2 generator. The stage 2 discriminator is trained to maximize Equation (3), and the stage 2 generator is trained to minimize Equation (4). The random noise is used only in stage 1, not in this stage. Instead, a low-resolution image generated from the generator in stage 1, x_{row} is used. A high-resolution image, x_{high} , is generated by inputting the text embedding vector extracted from the text encoder and the low-resolution image generated from the stage1 generator. The discriminator determines whether the image and text match by inputting an image and text embedding vector.

$$\mathcal{L}_{D_2} = \mathbb{E}_{(x_{row}, h) \sim p_{data}} [\log D_2(x, h)] + \mathbb{E}_{x_{row} \sim p_{G_1}, h \sim p_{data}} [\log(1 - D_2(G_2(x_{row}, h_{ca}), h))] \quad (3)$$

$$\mathcal{L}_{G_2} = \mathbb{E}_{x_{row} \sim p_{G_1}, h \sim p_{data}} [\log(1 - D_2(G_2(x_{row}, h_{ca}), h))] + \lambda D_{KL}(\mathcal{N}(\mu_0(h), \Sigma(h)) \| \mathcal{N}(0, X)) \quad (4)$$

Algorithm 3 presents the procedure of generating realistic high-resolution images from text using BERT-based embeddings. In line 2, $\theta_{BERT}^{fine-tuned}$, the number of iterations S_2 , and the learning rates γ and ω of the generator and discriminator are set. In the for loop, the mini-batch image x_i and the text t_i are extracted and learning is performed for S_2 times. A random vector is extracted in line 5, and the text embedding is obtained by taking the mini-batch text as the input of the BERT in line 6. In line 10, text embedding h_{ca} and a random vector z are input together, and a low-resolution image is generated using stage1 generator. In line 11, the stage 2 discriminator determines whether the image and the text match, and the process of updating the discriminator and the generator is carried out in lines 15 and 16, respectively.

Algorithm 3 Realistic High-Resolution Image Generation from Text by Using BERT-Based Embedding

```

1: Initialize parameters of stage2 generator and discriminator  $G_2, D_2$ 
2: Set fine-tuned BERT  $\theta_{BERT}^{fine-tuned}$ , number of stage2 step  $S_2$ , and  $G_2, D_2$  learning rates  $\gamma, \omega$ 
   text descriptions and images of dataset  $T = \{t_1, t_2, \dots, t_n\}$ ,  $X = \{x_1, x_2, \dots, x_n\}$ 
3: For  $step = 1, \dots, S_2$  do
4:   For  $i = 1, \dots, n$  do
5:      $z \sim \mathcal{N}(0, 1)$  # sample of random noise
6:      $h \leftarrow \theta_{BERT}^{fine-tuned}(t_i)$  # sample of matching text description
7:      $h' \leftarrow \theta_{BERT}^{fine-tuned}(t'_i)$  # sample of miss – matching text description
8:      $h_{ca} \leftarrow ca(h)$  # conditioning augmentation of matching text description
9:      $h'_{ca} \leftarrow ca(h')$  # conditioning augmentation of miss – matching text description
10:     $\hat{x}_{row} \leftarrow G_1(z, h_{ca})$  generated row resolution images from stage1 generator
11:     $\hat{x}_{high} \leftarrow G_2(\hat{x}_{row}, h_{ca})$  # generated image from text description
12:     $s_{high}^r \leftarrow D_2(x_{high}, h_{ca})$  # real image, right text description
13:     $s_{high}^w \leftarrow D_2(x_{high}, h'_{ca})$  # real image, wrong text description
14:     $s_{high}^f \leftarrow D_2(\hat{x}_{high}, h_{ca})$  # fake image, right text description
15:     $D_2 \leftarrow D_2 - \gamma \partial \mathcal{L}_{D_2} / \partial D_2$  # update stage1 discriminator  $\mathcal{L}_{D_2} = \text{equation (3)}$ .
16:     $G_2 \leftarrow G_2 - \omega \partial \mathcal{L}_{G_2} / \partial G_2$  # update stage1 Generator  $\mathcal{L}_{G_2} = \text{equation (4)}$ .
17:   End for
18: End for

```

4. Experiments

We validate the proposed method both quantitatively and qualitatively. The experiment was conducted using CUB [22], which is one of the popular benchmark datasets for image generation from text. In our experiments, we used RTX 5000(16GB) $\times$ 4, and Intel(R) Xeon(R) Silver 4210R CPU @ 2.40GH $\times$ 2, with Pytorch 1.4.0 on Ubuntu 18.04.2.

4.1. Experimental Setup

4.1.1. Datasets

Table 1 provides information on the benchmark dataset used in the experiment: the training set, test set, and the number of sentences for each image of the CUB benchmark dataset. The CUB dataset consists of 200 classes. It consists of 11,788 images of birds, each with 10 textual descriptions. In about 80% of the CUB dataset, the proportion of the object is less than half the image size. In the experiment of this paper, preprocessing was performed so that the ratio of the object was greater than 0.75 by using a bounding box for the object.

Table 1. Statistics of dataset.

Dataset	Train	Test	#Captions per Image
CUB	8855	2,933	10

4.1.2. Evaluation Metric

As quantitative evaluation indicators, we use the inception score (IS) [23] and frechet inception distance (FID) [24].

The IS, which is the first metric for evaluating GAN, is computed by the inception-v3 network. Inception-v3 is an image classification model composed of a convolutional neural network [25]. In order to calculate the inception score, we pre-trained using the ImageNet dataset consisting of 1000 classes and about 1.2 million images. The image generated from the text is input to the pre-trained inception model, and the generative model is evaluated based on the output result. Equation (5) is the formula for calculating the inception score. By calculating the difference between two probability distributions

by the Kullback–Leibler divergence, the difference in information entropy that can occur when sampling is performed using another distribution similar to a specific distribution is calculated. For example, if the distribution of class A is different from the distribution of other classes, the inception score has a large value. This means that images of class A have different characteristics from images of other classes.

$$IS(G) = \exp\left(\mathbb{E}_{x \sim p_g} D_{KL}(p(y|x) \| p(y))\right) \quad (5)$$

The FID has been proposed to compensate for the disadvantage that the inception score does not use the distribution of actual data. Equation (6) is the formula for calculating the FID, which measures the distance between the synthetic data distribution, p_f , and the real data distribution, p_r . Therefore, the smaller the FID value, the better the visual quality of images.

$$FID = \|m - m_r\|_2^2 + Tr(C + C_r - 2(CC_R)^{1/2}) \quad (6)$$

4.1.3. The Compared Models

The experimental results are compared with GAN-INT-CLS, GAWWN, and AttnGAN, which are direct image generation methods, and the StackGAN which is a stacked method.

- **GAN-INT-CLS**
GAN-INT-CLS is an early generative adversarial neural network-based text-to-image generation model. This model creates a 64×64 image with a text description [13].
- **GAWWN**
GAWWN is also a generative adversarial neural network-based text-to-image generation model. This model creates a 128×128 image with a text description [26].
- **AttnGAN**
AttnGAN is a generative adversarial neural network based on attention mechanism. This model creates a 256×256 image with a text description [27].
- **StackGAN**
StackGAN is a text-to-image image creation model based on a stacked architecture. This is divided into the process of generating a 64×64 low-resolution image in the first stage and a 256×256 image in the second stage [14].
- **StackGAN(ours)**
StackGAN(ours) is a model implemented using the PyTorch library for a stack generative adversarial neural network model for comparison experiments. Due to the differences in the library used and initial weights, we get different performance measures from the original paper.
- **StackGAN+BERT(ours)**
StackGAN+BERT(ours) is the text-to-image generation model proposed in this paper. We use BERT as a text encoder to generate a 256×256 image from text.

4.2. Experimental Result

4.2.1. Quantitative Results

Table 2 shows the comparison of quantitative experimental results for generating realistic images from texts using BERT-based embedding proposed in this paper. The model proposed in this paper is denoted as StackGAN+BERT. As a quantitative experimental result, it was shown that the IS of the proposed model was about 0.74 higher than that of the existing stack generative adversarial neural network, and the FID was 14.1 lower.

Table 2. IS and FID of GAN-INT-CLS, GAWWN, AttnGAN, StackGAN, StackGAN(ours), and our proposed model on the CUB dataset.

Dataset	Model	IS $\uparrow$	FID $\downarrow$
CUB	GAN-INT-CLS	2.88 ± 0.04	68.79
	GAWWN	3.62 ± 0.07	67.22
	StackGAN	3.70 ± 0.04	51.89
	AttnGAN	4.36 ± 0.03	-
	StackGAN(ours)	4.16 ± 0.02	38.73
	StackGAN+BERT(ours)	4.44 ± 0.04	37.79

The text-to-image generation model proposed in this paper is different from the embedding method used in previous studies. The embedding of the proposed method was fine-tuned to fit the pre-trained BERT with a large amount of data for the text-to-image task. Due to this, the space between data in the text manifold is small, so that features can be extracted from texts that are not seen during training in the fine-tuning process. However, existing studies use limited datasets to pre-train the zero-shot visual recognition task. Therefore, the empty space between data in the text manifold is relatively large, and bad features are extracted for the text expression that are not seen during training.

In general, deep learning can achieve good output when good feature values are input. In text-to-image tasks, it is also easy to create an image that matches the text description when a high-quality text representation is input. In other words, the T2I model using the BERT-based embedding proposed in this study improved the performance of the quantitative evaluation methods IS and FID scores by generating images using high-quality text embedding.

4.2.2. Qualitative Results

In the image generation field, it is difficult to measure the performance of the model only with quantitative evaluation, and it is necessary to perform qualitative evaluation on the generated image.

Figure 2 is a comparative figure that performed qualitative evaluation on the CUB dataset. For the text used for evaluation, the text description that was not used for learning was used. Qualitative evaluation results show that the proposed model generates realistic high-resolution images when compared with existing studies. For example, the existing stack generative adversarial neural network generated images without webbed feet from the text description “A bird with a medium orange bill white body gray wings and webbed feet”, but the proposed model generated images including them. Moreover, in the text description, “This small bird has a white breast, light gray head, and black wings and tail”, the existing stack generative adversarial neural network failed to generate a tail, but the proposed model showed that it did. It has been shown that high-quality images are generated from high-quality text representations.

Text Description	“The bird is short and stubby with yellow on its body”	“A bird with a medium orange bill white body gray wings and webbed feet”	“A small bird with varying shades of brown with white under the eyes”	“This small bird has a white breast, light grey head, and black wings and tail”
64 × 64 GAN-INT-CLS
128 × 128 GAWWN
256 × 256 StackGAN
256 × 256 StackGAN +BERT

Figure 2. Qualitative Results on CUB.

5. Discussion and Conclusions

In this paper, we proposed a T2I model capable of generating realistic images from text using BERT-based embedding. The proposed model was fine-tuned to fit the pre-trained BERT, which exhibits high performance in the natural language processing field, to the task of image creation in text. Due to this, there is less free space between data in the text manifold. Therefore, it was possible to extract text embeddings of relatively high quality compared with the existing embedding methods for texts that were not seen during learning in the fine-tuning process. As a result of the experiment, the quantitative evaluation of IS was about 0.74 high and FID was 14.1 low, showing that the proposed method is effective. Compared with the existing text-to-image generation model, our method generates high-resolution images with diversity for unseen textual descriptions. In the future, we will verify the effect of BERT-based embedding on text-to-image creation tasks using various data, and try applying various keyword extraction algorithms for effective analysis of input text. In addition, we plan to conduct research on designing sophisticated loss functions and generating images with higher resolution from text using a small amount of data.

Author Contributions: Conceptualization, S.N. and M.D.; methodology, S.N.; formal analysis, S.N.; writing—original draft preparation, S.N.; writing—review and editing, K.Y. and J.K.; supervision, J.K.; All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (2021R1A2C2008414), the Next-Generation Information Computing Development Program through the National Research foundation of Korea (NRF) funded by the Ministry of Science, ICT (NRF-2017M3C4A7083279), and the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2021-2020-0-01789) supervised by the IITP (Institute for Information and Communications Technology Planning and Evaluation).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Baltrušaitis, T.; Ahuja, C.; Morency, L.P. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 423–443. [CrossRef] [PubMed]
- Frolov, S.; Hinz, T.; Raue, F.; Hees, J.; Dengel, A. Adversarial text-to-image synthesis: A review. *arXiv* **2021**, arXiv:2101.09983. [CrossRef] [PubMed]
- Uppal, S.; Bhagat, S.; Hazarika, D.; Majumder, N.; Poria, S.; Zimmermann, R.; Zadeh, A. Multimodal research in vision and language: A review of current and emerging trends. *Inf. Fusion* **2021**, *77*, 149–171. [CrossRef]
- Qi, D.; Su, L.; Song, J.; Cui, E.; Bharti, T.; Sacheti, A. Imagebert: Cross-modal pre-training with large-scale weak-supervised image-text data. *arXiv* **2020**, arXiv:2001.07966.
- Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the Advances Neural Information Processing Systems Conference, Palais des Congrès de Montréal, MTL, Canada, 8–13 December 2014; pp. 2672–2680.
- Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
- Radford, A.; Metz, L.; Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In Proceedings of the 5th International Conference Learning Representations Workshop Track, San Juan, Puerto Rico, 2–4 May 2016.
- Mirza, M.; Osindero, S. Conditional generative adversarial nets. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 1335–1344.
- Zhu, J.-Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the International Conference Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232. Available online: <https://arxiv.org/abs/1703.10593> (accessed on 20 November 2021).
- Gatys, L.A.; Ecker, A.S.; Bethge, M. Image style transfer using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 2414–2423.
- Dong, C.; Loy, C.C.; He, K.; Tang, X. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 295–307. [CrossRef] [PubMed]
- Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; Shi, W. Photo-realistic single image super-resolution using a generative adversarial network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
- Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; Lee, H. Generative adversarial text to image synthesis. In Proceedings of the International Conference on Machine Learning (PMLR), New York, NY, USA, 20–22 June 2016.
- Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; Metaxas, D.N. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
- Zhang, Z.; Xie, Y.; Yang, L. Photographic text-to-image synthesis with a hierarchically-nested adversarial network. In Proceedings of the IEEE Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6199–6208.
- Sun, L.; Wang, J.; Zhang, K.; Su, Y.; Weng, F. RpBERT: A Text-image Relation Propagation-based BERT Model for Multimodal NER. In Proceedings of the thirty-fifth AAAI Conference on Artificial Intelligence (AAAI), Palo Alto, CA, USA, 2–9 February 2021.
- Zhang, Q.W.; Zhang, X.; Yan, Z.; Liu, R.; Cao, Y.; Zhang, M.L. Correlation-Guided Representation for Multi-Label Text Classification. In Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI), Montreal, QC, Canada, 19–27 August 2021.
- Mirza, M.; Osindero, S. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, Minnesota, 3–5 June 2019; Volume 1, pp. 4171–4186.
- Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S.R. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Brussels, Belgium, 2–4 November 2018.

20. Wang, T.; Zhang, T.; Lovell, B. Faces a la Carte: Text-to-face generation via attribute disentanglement. In Proceedings of the 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikola, HI, USA, 5–9 January 2021. arXiv:2006.07606.
21. Pavlo, D.; Lucchi, A.; Hofmann, T. Controlling style and semantics in weakly-supervised image generation. In Proceedings of the European Conference on Computer Vision, Glasgow, UK, 23–28 August 2020; pp. 482–499.
22. Wah, C.; Branson, S.; Welinder, P.; Perona, P.; Belongie, S. *The Caltech-UCSD Birds-200-2011 Dataset, Technical Report CNS-TR-2011-001*; California Institute of Technology: Pasadena, CA, USA, 2011.
23. Barratt, S.; Sharma, R. A note on the inception score. *arXiv* **2018**, arXiv:1801.01973.
24. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In Proceedings of the International Conference on Neural Information Processing Systems (NIPS), Hong Kong, China, 4 January 2017; pp. 6626–6637.
25. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26–30 June 2016.
26. Reed, S.E.; Akata, Z.; Mohan, S.; Tenka, S.; Schiele, B.; Lee, H. Learning what and where to draw. In Proceedings of the International Conference on Neural Information Processing Systems (NIPS), Barcelona, Spain, 5–10 December 2016.
27. Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; He, X. AttnGAN: Fine-grained text to image generation with attentional generative adversarial networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (ICCV), Salt Lake City, UT, USA, 18–22 June 2018.