applied sciences

MDPI

Article

A Study on Generating Webtoons Using Multilingual Text-to-Image Models

Kyungho Yu, Hyoungju Kim, Jeongin Kim, Chanjun Chun and Pankoo Kim *

Department of Computer Engineering, Chosun University, 309 Pilmun-Daero, Dong-Gu, Gwangju 61452, Republic of Korea; infinitegh@chosun.ac.kr (K.Y.); snowlisakim@gmail.com (H.K.); jungingim@gmail.com (J.K.); cjchun@chosun.ac.kr (C.C.)
* Correspondence: pkkim@chosun.ac.kr

Abstract: Text-to-image technology enables computers to create images from text by simulating the human process of forming mental images. GAN-based text-to-image technology involves extracting features from input text; subsequently, they are combined with noise and used as input to a GAN, which generates images similar to the original images via competition between the generator and discriminator. Although images have been extensively generated from English text, text-to-image technology based on multilingualism, such as Korean, is in its developmental stage. Webtoons are digital comic formats for viewing comics online. The webtoon creation process involves story planning, content/sketching, coloring, and background drawing, all of which require human intervention, thus being time-consuming and expensive. Therefore, this study proposes a multilingual text-to-image model capable of generating webtoon images when presented with multilingual input text. The proposed model employs multilingual BERT to extract feature vectors for multiple languages and trains a DCGAN in conjunction with the images. The experimental results demonstrate that the model can generate images similar to the original images when presented with multilingual input text after training. The evaluation metrics further support these findings, as the generated images achieved an Inception score of 4.99 and an FID score of 22.21.

Keywords: multilingual BERT; text-to-image; DCGAN; webtoon; GAN

Citation: Yu, K.; Kim, H.; Kim, J.; Chun, C.; Kim, P. A Study on Generating Webtoons Using Multilingual Text-to-Image Models. *Appl. Sci.* **2023**, *13*, 7278. https://doi.org/10.3390/app13127278

Academic Editor: Oscar Reinoso García

Received: 20 April 2023
Revised: 15 June 2023
Accepted: 15 June 2023
Published: 19 June 2023

1. Introduction

With the advancement of deep learning techniques, they have been widely utilized in various fields, demonstrating high performance [1–3]. Image generation technology has progressed rapidly, starting with generative adversarial networks (GANs), which generate fake images similar to the original ones through an adversarial learning process between the generator and discriminator networks. Currently, they can generate photorealistic images that are difficult to distinguish from those drawn by humans.

Text-to-image technology refers to the process of enabling computers to generate images based on input text [1,4]. Deep learning-based text-to-image technology creates images that reflect the contextual features of the text used to condition an image generator [5]. Traditional text-to-image approaches extract keywords from sentences, synthesize images corresponding to those keywords, and require human intervention. However, deep-learning-based text-to-image models extract features from the input text and generate images based on these features. Text-to-image models trained on GANs create images similar to the input text by conditioning the image generator to the feature vector of the sentence. However, this method fails to reflect the contextual meaning of individual words in a sentence, thus generating low-quality images. To overcome this limitation, attention mechanisms have been introduced in AttnGAN [6] to improve the quality of the generated images. Since then, text-to-image models such as StackGAN [7], MirrorGAN [8], R-GAN [9], Stacking VAE and GAN [10], and Dcfgan [11] have been developed. Despite

the development of GAN-based text-to-image models, they suffer from unstable image generation due to the training imbalance between the generator and discriminator. To overcome these limitations, text-to-image technology has recently focused on multimodal learning and diffusion models. Multimodal learning involves representing information in various forms, such as images, sounds, and text, and clustering similar data using contrastive learning [12]. Dalle-2 [13] developed by OpenAI, and DreamBooth [14], developed by Google, utilize multimodal space as an encoder and a diffusion model as a decoder to generate images based on the predicted image embedding from the input text. Currently, most deep learning-based text-to-image methods couple English-language text with image data, which limits their accessibility to countries where English is not spoken as the primary language. Furthermore, translating the native text into English as a preprocessing step incurs additional complexity and may not achieve the same efficiency. Therefore, a multilingual text-to-image technology that can generate images of similar quality, regardless of the input language, must be developed.

A deep learning-based text-to-image model that generates images similar to those drawn by humans can be used in the entertainment industry, for example, to create virtual characters and animations. Webtoons, a portmanteau of "web" and "cartoon," are comics that are published on the internet. The webtoon creation process involves story planning, storyboarding/sketching, coloring, and background drawing. As each stage of webtoon production requires human involvement, it is time-consuming and costly. To minimize human intervention, artificial intelligence-based methods, such as automatic coloring, automatic line drawing, and style transfer techniques, are typically utilized in different stages of webtoon creation.

For webtoons to attract readers, their overall stories are important. To keep readers interested and immersed as webtoons progress, tension and resolution must be provided. To ensure that the readers' interest and immersion are not lost, the author plans the story before creating the webtoon. In the planning stage, the author sets the genre, characters, and worldviews. Before drawing the webtoon, the author describes the scene in detail, which is called a treatment. Treatment is a term used not only for webtoons but also for content production, such as movies and dramas. When producing movies, dramas, or webtoons, treatments are used as references to describe the key elements of a scene, such as the time, place, and characters involved. Thus, treatments contain rich information regarding the scenes of the final webtoon product, thus making them suitable input data for deep-learning-based text-to-image models.

To overcome these limitations, we proposed a multilingual text-to-image model that can generate webtoon images from not only English but also other languages as inputs. To develop a multilingual text-to-image model, we constructed and trained a webtoon dataset consisting of English and Korean text and image data. The training process involves extracting features from multilingual text inputs using multilingual BERT and training a GAN-based text-to-image model with image data. Once the training is complete, the multilingual text-to-image model can generate webtoon images when given a multilingual text input. The main contributions of this study are as follows: (1) Unlike existing text-to-image technology that generates images from text written in English, the proposed multilingual text-to-image model uses multilingual BERT as a text encoder to generate images from multilingual input. (2) As the webtoon dataset is trained on a text-to-image model, it is expected to contribute to the creation of webtoon content by allowing the model to generate webtoon images from multilingual text inputs.

The remainder of this paper is organized as follows: Section 2 describes the self-supervised pre-training models and deep learning-based text-to-image technology used in this study. Section 3 proposes a construction method for the multilingual treatment-webtoon dataset and the GAN-based text-to-image model. Section 4 trains the constructed multilingual text-to-image dataset using the text-to-image model and verifies the results. Finally, Section 5 concludes this study and discusses future research directions.

2. Related Work

2.1. Self-Supervised Pre-Trained Models

Natural language processing (NLP) is a field of artificial intelligence that focuses on teaching machines to understand human language. Therefore, the human language must be translated into a suitable format, which is called word embedding [15]. Word embedding refers to the mapping of words that constitute a text into a real-valued vector. When words are input into this type of word embedding, they are vectorized in a multi-dimensional space, which allows the degree of similarity between words to be measured. Embedding models, such as word2vec and GloVe, predict intermediate or surrounding words within a predefined window size in a sentence, resulting in a word embedding. However, this method cannot fully capture the contexts of words in sentences, resulting in the same embedding being assigned to words with different contextual meanings. To address this issue, embeddings from the language model (ELMo) were developed, which is a pre-trained language model that uses bidirectional LSTMs to reflect the contextual meaning in the sentence-embedding models [16]. Later, BERT was developed as a large-scale pre-trained language model based on transformer architecture [17]. BERT is a pre-trained language model that uses attention mechanisms to prevent the information loss that can occur with RNN or LSTM models as the length of the input sentence increases [18]. The BERT model comprises a transformer encoder composed of multi-head attention and fully connected layers. The input sentence was tokenized, segmented, and position-embedded before being partially masked and input into the transformer. The training was performed by predicting the masked tokens, thereby enabling the model to understand the context. Thus, the trained model can extract feature vectors from the input sentences, which can then be used to perform several NLP tasks, such as sentence classification and question answering [19]. Numerous transformer-based derivative models have been developed after BERT and have demonstrated excellent performance in SOTA.

2.2. Deep Learning-Based Text-to-Image Generation

Deep learning-based text-to-image generates images using the semantic information of the input sentence. GAN-based text-to-image generation can be divided into two stages: extraction of semantic information from the text and text-to-image generation. First, the sentence is embedded using word embedding, and features are extracted by inputting them into an RNN or transformer structure. In the image-generation stage, the GAN inputs the semantic information of the text, along with noise of the same size as that of the output-generated image, to the generator to create an image. By contrast, the discriminator compares the images generated by the generator with real images. Through the adversarial learning of two neural networks that constantly create fake images and judge whether the generated image is real, they generate fake images that are similar to the real ones. The initial deep-learning-based text-to-image model uses a deep convolutional GAN (DCGAN) [5]. The DCGAN-based text-to-image model generates an image using the conditional vector of a sentence as the condition. Although the generated image is similar in meaning to the input sentence, it does not reflect the contextual meaning of each word in the sentence, thus generating low-quality images. To overcome these disadvantages, the AttnGAN, which introduces an attention mechanism for image generation, was developed. AttnGAN first uses the feature vector of the sentence to create an image, and when generating the next-stage image, it combines the attention map of the word with the image vector to create an improved image step-by-step. AttnGAN can express each word in a sentence in a detailed manner compared with DCGAN; subsequently, StackGAN [7], MirrorGAN [8], DM-GAN [9], R-GAN [10], Stacking VAE and GAN [10], and Dcfgan [11] were developed, which can express the meaning of the input sentence more precisely and generate high-resolution images.

Recently, diffusion-model-based text-to-image methods have shown better performance compared with GAN-based methods. GANs suffer from the problem of imbalanced training between the generator and discriminator, which leads to collapse. When

the generator creates images that can easily fool the discriminator, it stops learning. By contrast, diffusion models generate images by iteratively adding noise to the training data in the forward process and recovering data from noise in the reverse process [20]. After training, the diffusion models generate images similar to the training data using a reverse process. The DALL-E 2 model released by OpenAI uses contrastive language image pre-training (CLIP) as an encoder and the diffusion model as a decoder to generate images from text [13]. CLIP extracts features from both text and image data and determines the similarity between them using contrastive learning. After training, text and image features with similar characteristics are densely packed in a multimodal space [21]. CLIP extracts an image vector similar to the input text from the pre-trained model and generates an image by inputting it into the diffusion model. Diffusion model-based text-to-image methods have shown superior performance compared with GAN-based methods on benchmark datasets in generative modeling [22,23]. Examples of diffusion model-based text-to-image generation methods include Dalle-2 [13], DreamBooth [14] and imagegen.

GAN and diffusion models have different strengths and weaknesses [24,25]. The GAN has the advantage of a shorter training time compared with diffusion models; however, it has a risk of model collapse owing to the learning imbalance between the generator and discriminator. By contrast, diffusion models have the advantage of generating a wider variety of high-quality images than GANs, despite requiring longer training times. In this study, we trained a DCGAN-based text-to-image model on a multilingual text-to-image webtoon dataset specialized in the webtoon domain to enable the application of deep learning-based text-to-image technology in the webtoon industry with a shorter training time.

3. Text-to-Image Generation Using Multilingual BERT

In this section, we propose a multilingual text-to-image model that generates webtoon images when multilingual text input is provided. The proposed text-to-image model for webtoon generation aims to generate webtoons similar to the given text input in English and Korean and proceeds in the following two steps: First, we constructed a multilingual text-to-webtoon dataset using the MSCOCO dataset, a benchmark dataset, and a cartoon-based GAN. Next, we used Multilingual BERT to extract sentence vectors from the multilingual text-to-webtoon dataset and used them as conditions for DCGAN training.

3.1. Webtoon Dataset

To generate webtoon images using the treatment as input for text-to-image generation models, the model must be trained on both the treatment and webtoon datasets. As constructing a large-scale dataset consisting of actual treatments and webtoons is challenging, the benchmark dataset commonly used in generative models, the MSCOCO dataset, was transformed into cartoons using a pre-trained CartoonGAN [26]. The MSCOCO dataset is a description photo dataset released by Microsoft (Washington, DC, USA) that consists of 123,287 training and validation images, each accompanied by five descriptions [27]. The officially provided dataset includes only English; therefore, the Korean MSCOCO dataset, which was translated from the MSCOCO dataset by the AI hub in Korea, was additionally used to construct a multilingual treatment dataset [28]. The Korean MSCOCO dataset also consisted of five Korean descriptions per image, resulting in ten sentences per image: five in English and five in Korean. The pre-trained CartoonGAN was trained in four styles, as shown in Figure 1. Thus, 1,232,870 treatment webtoon data pairs were constructed for each style, thereby resulting in 4,931,480 data pairs for all four styles.

Figure 1. Examples of datasets transformed using CartoonGAN.

3.2. Text-to-Image Generation with Multilingual BERT

A deep learning-based text-to-image model generates images from input text in two stages. The first stage is feature extraction of the input text, and the second stage is the training of a GAN model by combining the extracted text features with noise. Extracting text features that reflect the contextual characteristics embedded in the text is crucial because the quality of the generated images is heavily influenced by the quality of the learned features. Essentially, if the model fails to extract features that capture the essence of the keywords between the text and image, the generated image may deviate from the intended meaning of the text. Therefore, in this study, as shown in Figure 2, we utilized a pre-trained multilingual BERT model [29], which has demonstrated high performance in NLP, to extract the feature vectors of sentences. The sentence feature vectors are obtained using the "cls" token in the BERT model as the sentence vector.

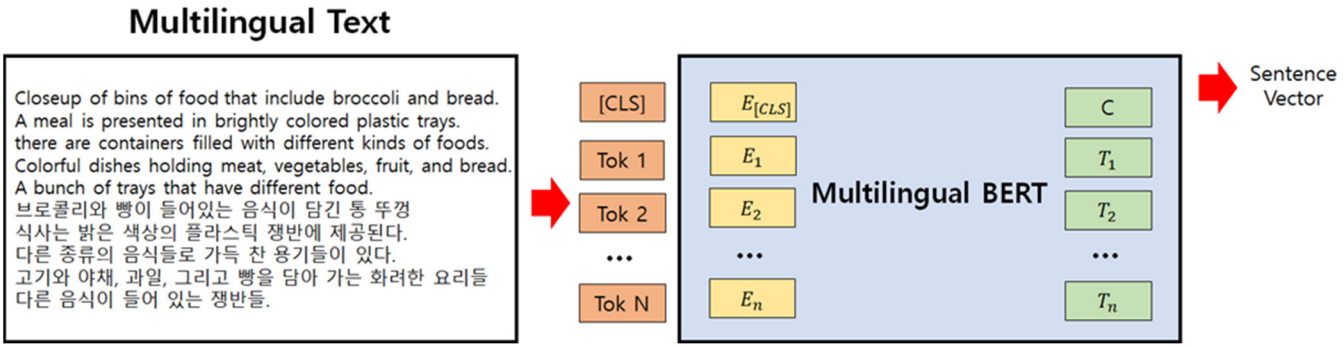


Figure 2. Sentence vector extraction using Multilingual BERT. The sentence on the left side of the figure is a sentence written in Korean with the same meaning as the English.

To generate webtoons based on the extracted features of the multilingual treatments, a GAN-based DCGAN model was used. Figure 3 shows the structure of the text-to-image model using DCGAN. The GAN generates images similar to the original data through competition between two neural networks called the generator and discriminator. The generator learns by reflecting the distribution of the input data, generates fake data from random vectors, and compares the data generated by the generator with the original data to determine whether the data are real or fake. To generate webtoons from multilingual treatments, the sentence vectors extracted through the Multilingual BERT are combined with noise and input to the DCGAN for training. The proposed generator model consists of six blocks comprising a convolutional layer, dropout layer, batch normalization layer, ReLU layer, and upsamples as they pass through each block. After passing through the final layer, it generates an image size of $3 \times 64 \times 64$. The discriminator consists of six blocks, comprising a convolutional layer, batch normalization layer, leaky ReLU layer, and downsamples as it passes through each block. Finally, it combines text features and uses a sigmoid to determine whether the text is real or fake.

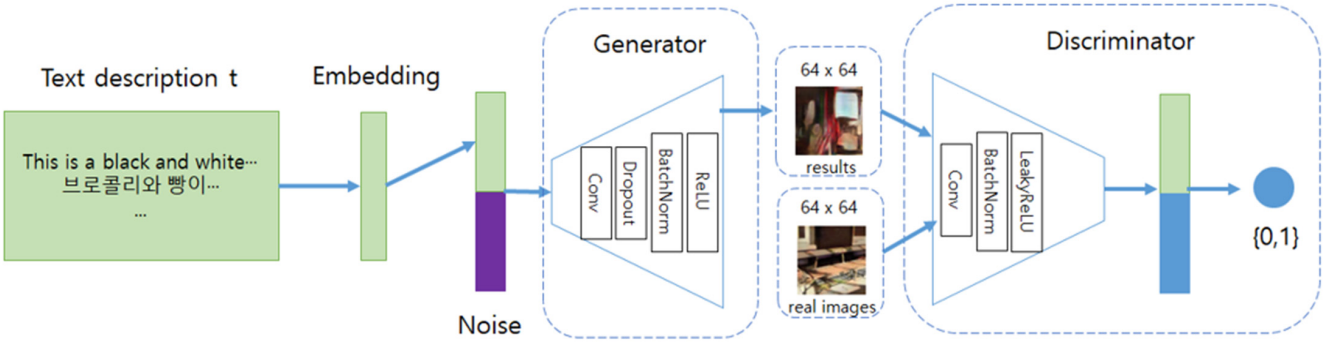


Figure 3. Architecture of text-to-image model using DCGAN. The text on the left side of the figure is an example of a sentence in both English and Korean with the same meaning.

The text-to-image loss function using the DCGAN is given by Equation (1), where G represents the generator function, D represents the discriminator function, x is the input vector, z is the random vector, and φ(t) represents the feature vector of the input text. The discriminator is trained to have a larger value for Equation (1) to distinguish between real and fake data, whereas the generator is trained to have a smaller value for Equation (1) to generate fake data similar to the real data. The discriminator of the DCGAN used in this study receives three types of data as inputs and is trained to classify real image-right text as real, real image-wrong text as fake, and fake image-real text as fake. To prevent overfitting, label smoothing and feature matching were used in the experiments. The loss function is as follows:

$$\min_{D}\max_{G} \mathrm{E}(\mathrm{D},\mathrm{G}) = \mathrm{E}_{(\mathrm{x},\mathrm{t})\sim \mathrm{p}_{\mathrm{data}}(\mathrm{x},\mathrm{t})}[\log \mathrm{D}(\mathrm{x},\varphi(\mathrm{t}))] + \mathrm{E}_{\mathrm{z}\sim \mathrm{p}_{\mathrm{z}}(\mathrm{z}),t\sim \mathrm{p}_{t}(t)}[\log 1 - \mathrm{D}(\mathrm{G}(\mathrm{z},\varphi(\mathrm{t})))] + \mathrm{E}_{\mathrm{x}\sim \mathrm{p}_{x}(x),\hat{t}\sim \mathrm{p}t(t)}[\log 1 - \mathrm{D}(\mathrm{x},(\mathrm{z},\varphi(\hat{t})))]\Big] \quad (1)$$

4. Experiment

4.1. Datasets

In this section, a webtoon dataset for training a multilingual text-to-image model is constructed using a benchmark dataset. To create a multilingual webtoon dataset, the MSCOCO dataset released by Microsoft and the Korean MSCOCO dataset translated and released by the AI Hub were used, and CartoonGAN was used to transform them into cartoons, as mentioned in Section 3.1. The original MSCOCO dataset consists of 82,783 training images and 40,504 validation images, each containing five English descriptions. The Korean MSCOCO dataset was added to create 1,232,870 pairs of multilingual webtoon data, with each image containing ten descriptions (five in English and five in Korean). CartoonGAN can transform images into four different styles; therefore, a total of 1,232,870 pairs of webtoon data were constructed by pairing one image with one treatment. Table 1 shows examples of multilingual treatment webtoon datasets.

Table 1. An example of a multilingual treatment-webtoon dataset composed of Korean text with the same meaning as English.

Train Image	Caption
	A tennis player in action on the court.
	a male in a white shirt is playing tennis
	a man that is playing tennis on a court
	A male tennis player hits the ball on a grass court.
	A man in motion hitting a tennis ball with a tennis racket on a tennis court.
	테니스 선수가 코트에서 뛰고 있다.
	흰 셔츠를 입은 남자가 테니스를 치고 있다.
	코트에서 테니스 치는 남자
	한 남자 테니스 선수가 잔디 코트에서 공을 친다.
	테니스 코트에서 테니스 라켓을 가지고 테니스 공을 치며 움직이는 남자
	Close-up of bins of food that include broccoli and bread.
	A meal is presented in brightly colored plastic trays.
	there are containers filled with different kinds of foods
	Colorful dishes holding meat, vegetables, fruit, and bread.
	A bunch of trays that have different food.
	브로콜리와 빵이 들어 있는 음식이 담긴 통 뚜껑
	식사는 밝은 색상의 플라스틱 쟁반에 제공된다.
	다른 종류의 음식들로 가득 찬 용기들이 있다.
	고기와 야채, 과일, 그리고 빵을 담아 가는 화려한 요리들
	다른 음식이 들어 있는 쟁반들.

4.2. Webtoon Generation Using DCGAN

To generate webtoons by inputting multilingual text into a deep learning-based text-to-image model, the multilingual webtoon dataset was trained using the DCGAN model. As training a DCGAN with a real-time pre-trained multilingual BERT to extract feature vectors for multilingual text takes a longer time, the feature vectors were extracted and stored in advance and then loaded and used during training. The experiment was conducted using 820,752 multilingual text webtoon data pairs for each style, excluding incomplete data. The validation data consisted of 405,040 data points, which were divided into validation and test datasets at an 8:2 ratio. The size of the images used in the experiment was $3 \times 64 \times 64$, the noise dimension was 100, the batch size was 32, the learning rate was set to 0.0002 for both the generator and discriminator, and the optimizer used was Adam. The experiment was conducted using four A5000 GPUs, and the deep learning framework used for the experiment was PyTorch.

Figure 4 shows the images generated by the generator when the validation data text was input as the number of training epochs increased. As shown in Figure 4a, when the number of training epochs was one, the generated images were vague. As the training progressed, the generator began drawing the shape of the objects, as shown in Figure 4e, for up to 100 epochs. However, the generator could not generate further images after a certain point. Therefore, our experiments suggest that the model trained for 75 epochs is suitable for the webtoon dataset constructed in this study, based on the loss of the generator and discriminator as well as the image output using the validation data.

Figure 4. Image generation at different epochs on validation data.

Figures 5–7 show the images generated by inputting the test data into the generator model trained for 75 epochs. Figures 5 and 6 show the webtoons generated when Korean and English treatments were input, respectively, and Figure 7 shows the webtoons generated when treatments with the same meaning in Korean and English were input. Evidently, although the similarity with the original images was not high, the generator was still able to produce some shapes that were expressed in the text. When analyzing objects, the experimental results suggest that the generator does not express objects such as humans and animals well but does express objects such as chairs relatively well. Figure 7 shows the images generated by inputting the same meaning of Korean, English, Japanese, and Chinese into the DCGAN trained in this research. Although the generated webtoons may be different from each other, the semantic shapes of the words expressed in the treatments are similarly represented. Similar to the previous experiments, it can be observed that the generator is drawing shapes related to the keywords in the sentences.

Caption	Real image	Fake image
(a) 의자와 테이블과 여자가 있는 방		

Figure 5. *Cont.*

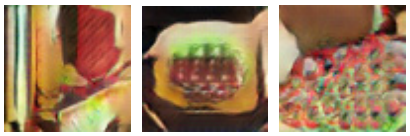
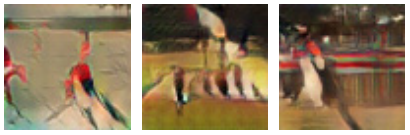
(b) 텔레비전과 테이블이 있는 생활 공간		
(c) 치즈 브로콜리와 닭고기가 들어 있는 흰색 접시		
(d) 거울 아래에 있는 욕실 싱크대		
(e) 한 남자가 경기에서 테니스 공을 서브하기 위해 준비한다		

Figure 5. Webtoon generated from Korean treatment input. The Korean sentences (a–e) have the following English equivalents with the same meaning: (a) A room with chairs, tables, and a woman. (b) A living space with a television and a table. (c) A white plate with cheese, broccoli, and chicken. (d) A bathroom sink below the mirror. (e) A man prepares to serve a tennis ball in a game.

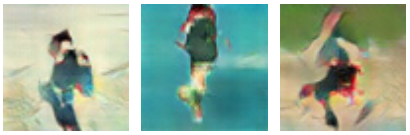
Caption	Real image	Fake image
youth_surfing_on_a_body_of_water_outside		
This_is_a_black_and_white_picture_of_a_chester_bench		
A smiling person holding a snowboard standing on a snow-covered hill		
A turkey dinner shows corn, peas, mashed potatoes, and biscuits all on one plate		
A business called Ray's Tavern with motorcycles sitting outside it		

Figure 6. Webtoon generated from English treatment input.

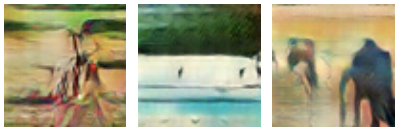
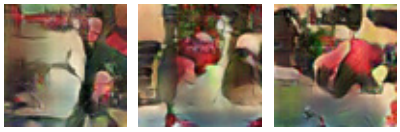
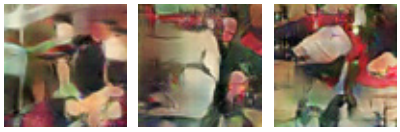
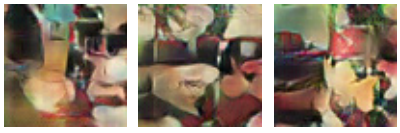
Caption	Real image	Fake image
(a) A man surfs and plays in the foamy ocean		
(b) 한 남자가 거품이 이는 바다에서 서핑을 하고 놀았다		
(c) 一个男人在泡沫飞溅的海上冲浪和玩耍		
(d) ある男性が泡立つ海でサーフィンをして遊びました		
(e) A plate of food with bread, grape tomatoes, cheese, cucumbers, and sauce on it		
(f) 두툼한 빵 위에 야채와 치즈를 얹은 샌드위치		
(g) 在厚实的面包上铺上蔬菜和奶酪的三明治		
(h) 野菜とチーズをのせた分厚いパンのサンドイッチ		
(i) Two beds in a room with a light and green and white curtain		
(j) 밝은 색과 초록색 그리고 흰색 커튼이 있는 방에 있는 두개의 침대		
(k) 两张床位于有明亮色彩、绿色和白色窗帘的房间中		
(l) 明るい色と緑と白のカーテンがある部屋には 2 つのベッドがあります		

Figure 7. Webtoon generated from multilingual treatment input with the same meaning. (a) is an English sentence with the same meaning as (b–d), and similarly, (e) is an English sentence with the same meaning as (f–h), and (i) is an English sentence with the same meaning as (j–l).

Table 2 summarizes the proposed multilingual text-to-image model on the test dataset in terms of the inception and Frechet inception distance (FID) scores. For the evaluation measurements, a total of 1,620,160 images were used, with four images per text. The inception score evaluates the quality and diversity of the generated images, whereas the FID score compares the mean and covariance values of the feature values of the real and generated images. The inception and FID scores for the multilingual text-to-image model were 4.992 and 22.212, respectively. The inception score of the DCGAN model trained on the MSCOCO dataset was 7.88 [4]. A lower inception score of the DCGAN model trained on the multilingual webtoon dataset was expected because of the distortion of the represented shapes in the images during the cartoonization process.

Table 2. Performance evaluation of the proposed multilingual text-to-image model.

Dataset	Inception Score	FID Score
Multilingual Webtoon 1,620,160 image	4.992	22.212

5. Conclusions

In this study, we proposed a multilingual text-to-image model that can generate webtoons when multilingual inputs are provided. To construct a multilingual webtoon dataset, we used the Korean MSCOCO dataset from the AI Hub and transformed it into webtoon images using CartoonGAN. The resulting webtoon dataset consisted of Korean and English treatments in four art styles, and we constructed 1,232,879 pairs of multilingual text webtoon data. We used a GAN-based DCGAN model as the text-to-image model and trained it on a dataset of 820,752 pairs from one style of a multilingual webtoon dataset. The DCGAN used for training consists of a generator and discriminator. The generator uses the feature vector of the treatment and noise as input to generate a fake image, which is then evaluated by the discriminator to update the network weights and ultimately generate an image similar to the input webtoon image.

We validated the model by generating images using validation data and found that for up to 100 epochs of training, the model could accurately represent the shapes present in the treatments; however, the model stopped learning after this point. Therefore, we considered training the generator for 75 rounds, which produces high-quality images suitable for our multilingual webtoon dataset. When the generated images were evaluated using the test data, an inception score of 4.992 and a FID score of 22.212 were obtained. Although the generated images generally did not accurately represent the shapes of the original data, we confirmed that the shapes present in the input treatments could be expressed in the generated webtoons. We observed that the DCGAN trained on the webtoon dataset performed worse than that trained on the MSCOCO dataset. We speculate that this result was due to the distortion of the shapes in the images when the MSCOCO data were cartoonized, which affected the training. However, when we input the same multilingual sentences, the generated images represented the semantic shapes of the words expressed in the text.

In future studies, we plan on training a multilingual webtoon dataset using the latest text-to-image models to generate high-quality webtoons. This will help webtoons preview the scenes they envision in advance, which can aid the overall storytelling process. In addition, generating webtoons from treatments allows authors to modify and use them, thus leading to time and cost savings in webtoon production.

Author Contributions: K.Y. conducted the deep learning model experiments and manuscript writing for this paper. H.K. and J.K. handled the data set collection and transformation tasks. C.C. contributed to the experimental design and writing of the manuscript. P.K. supervised the development of the idea as well as overseeing the funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported as a 'Technology Commercialization Collaboration Platform Construction' project of the INNOPOLIS FOUNDATION (Project Number: 1711177250).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Agnese, J.; Herrera, J.; Tao, H.; Zhu, X. A survey and taxonomy of adversarial neural networks for text-to-image synthesis. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, e1345. [CrossRef]
2. Prakash, A.J.; Patro, K.K.; Samantray, S.; Pławiak, P.; Hammad, M. A Deep Learning Technique for Biometric Authentication Using ECG Beat Template Matching. *Information* **2023**, *14*, 65. [CrossRef]
3. Patro, K.K.; Prakash, A.J.; Samantray, S.; Pławiak, J.; Tadeusiewicz, R.; Pławiak, P. A Hybrid Approach of a Deep Learning Technique for Real–Time ECG Beat Detection. *Int. J. Appl. Math. Comput. Sci.* **2022**, *32*, 455–465.
4. Tewari, A.; Fried, O.; Thies, J.; Sitzmann, V.; Lombardi, S.; Sunkavalli, K.; Martin-Brualla, R.; Simon, T.; Saragih, J.; Nießner, M.; et al. State of the art on neural rendering. *Comput. Graph. Forum* **2020**, *39*, 701–727. [CrossRef]
5. 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, New York, NY, USA, 19–24 June 2016; pp. 1060–1069.
6. 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, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1316–1324.
7. 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, Venice, Italy, 22–29 October 2017; pp. 5907–5915.
8. Qiao, T.; Zhang, J.; Xu, D.; Tao, D. Mirrorgan: Learning text-to-image generation by redescription. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1505–1514.
9. Qiao, Y.; Chen, Q.; Deng, C.; Ding, N.; Qi, Y.; Tan, M.; Ren, X.; Wu, Q. R-GAN: Exploring human-like way for reasonable text-to-image synthesis via generative adversarial networks. In Proceedings of the 29th ACM International Conference on Multimedia, Virtual, 21–25 October 2021; pp. 2085–2093.
10. Zhang, C.; Peng, Y. Stacking VAE and GAN for context-aware text-to-image generation. In Proceedings of the 2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM), IEEE, Xi'an, China, 13–16 September 2018.
11. Tao, M.; Wu, S.; Zhang, X.; Wang, C. Dcfgan: Dynamic convolutional fusion generative adversarial network for text-to-image synthesis. In Proceedings of the 2020 IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), Chongqing, China, 6–8 November 2020; Volume 1, pp. 1250–1254.
12. Shi, Y.; Zhao, Y.; Liu, X.; Zheng, F.; Ou, W.; You, X.; Peng, Q. Deep adaptively-enhanced hashing with discriminative similarity guidance for unsupervised cross-modal retrieval. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 7255–7268. [CrossRef]
13. Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv* **2022**, arXiv:2204.06125.
14. Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; Aberman, K. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv* **2022**, arXiv:2208.12242.
15. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781.
16. Sarzynska-Wawer, J.; Wawer, A.; Pawlak, A.; Szymanowska, J.; Stefaniak, I.; Jarkiewicz, M.; Okruszek, L. Detecting formal thought disorder by deep contextualized word representations. *Psychiatry Res.* **2021**, *304*, 114135. [CrossRef] [PubMed]
17. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
18. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2017; Volume 30.
19. Tenney, I.; Das, D.; Pavlick, E. BERT rediscovers the classical NLP pipeline. *arXiv* **2019**, arXiv:1905.05950.
20. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 6840–6851.
21. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, Virtual, 18–24 July 2021; pp. 8748–8763.
22. Kim, G.; Kwon, T.; Ye, J.C. Diffusionclip: Text-guided diffusion models for robust image manipulation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2426–2435.

23. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 10684–10695.
24. Dhariwal, P.; Nichol, A. Diffusion models beat GANs on image synthesis. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 8780–8794.
25. Saharia, C.; Chan, W.; Chang, H.; Lee, C.; Ho, J.; Salimans, T.; Fleet, D.; Norouzi, M. Palette: Image-to-image diffusion models. In Proceedings of the ACM SIGGRAPH 2022 Conference Proceedings, Vancouver, BC, Canada, 7–11 August 2022; pp. 1–10.
26. Chen, Y.; Lai, Y.K.; Liu, Y.J. Cartoongan: Generative adversarial networks for photo cartoonization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9465–9474.
27. MSCOCO. Available online: https://cocodataset.org/ (accessed on 1 January 2023).
28. AI Hub. Available online: https://aihub.or.kr/ (accessed on 1 January 2023).
29. Pires, T.; Schlinger, E.; Garrette, D. How multilingual is multilingual BERT? *arXiv* **2019**, arXiv:1906.01502.