May-June 2020
ISSN: 0193-4120 Page No. 4694-4697

Web Server Based StackGAN-v2 Implementation

[1]Sowmya Sundari LK, [2]Adarsh R Nair, [3]Gokul Dev, [4]Jalay Rupera, [5]Shehan Silva

[1,2,3,4,5]School of Computing Science and Technology, REVA University, Karnataka, India
[1]somyasundarilk@reva.edu.in, [2]R15CS497@cit.reva.edu.in, [3]R17CS135@cit.reva.edu.in,
[4]R17CS176@cit.reva.edu.in, [5]R17CS157@cit.reva.edu.in

Article Info
Volume 83
Page Number: 4694-4697
Publication Issue:
May-June 2020

Abstract

Implementation of StackGAN on a website with the help of a web server to generate unique pictures based on a description given by a user. Computer vision generating good quality images from the user's text description is a troublesome problem and has many useful applications. Samples from current Text to Image methods can essentially demonstrate comprehension of the explanations given, but they fail to provide the information needed and the vivid sections of the objects. With this paper, we use Stacked Generative Adversarial Networks (StackGAN) to get realistic image using the text descriptions given by the user. Via a method of sketch-refinement we break the hard issue down into more manageable subproblems. We have used StackGAN because StackGAN-v2 shows more robust training behavior than StackGAN-v1 by approximating several distributions together. In StackGAN images are generated from different branches of the tree at multiple scales corresponding to the same scene. Comprehensive tests and comparisons with benchmark data sets indicate that significant changes are made by the suggested approach in generating photorealistic images based on text descriptions.

Keywords: *Text to Image; StackGAN-v2; GANs*

Article History
Article Received:*19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *12 May 2020*

1. Introduction

With the assistance of Ian Goodfellow, Generative Adversarial Networks (GANs) had been cautioned. In a particular context, GANs are a generator collection and a discriminator capable of contesting objectives. The generator is programmed to provide you with samples that are similar to the distribution of truth information to trick the discriminator, Although the discriminator is intended to distinguish the actual distributed samples of truth statistics to fool the discriminator, while truth facts are designed to separate real facts and fake samples.

We have tried to implement the Stacked generative adversarial network into the website to come up with the photographs from the given description by the user. StackGAN will be useful for users who need uniquecontent.

2. Related Work

The project has been made based on the paper for text to image generation using STACKGAN. The methodology was used to obtain images after undergoing a 2-Stage process that involves converting the generated image from a rough figure to a higher resolution image.

The second paper involves a more stable version of STACKGAN called STACKGAN-V2(STACKGAN++) that is more stable than the first and produces better quality images than before. Our project is based in this particular version as it gives us better and more reliable results for image generation. we have chosen to make images of birds due to the various attributes and characteristics available to describe various types of birds.

Variational Autoencoders (VAEs) are formulating the problem of probabilistic graphical models with the aim of maximizing the scope of information. Generative Adversarial Networks (GANs) have recently demonstrated encouraging success for producing sharper images. But the training uncertainty makes it hard for GANs to get high resolution images. plenty of works are proposed to stabilize the training and make the image qualities better. Most methods use simple conditioning variables like attributes or class labels. There are also works that are conditioned on photos to get photos, including the transition of domain picture modifications and super-resolution. Super-resolution techniques, however, may only add small specifics to low-resolution images, and

May-June 2020
ISSN: 0193-4120 Page No. 4694-4697

may not fix major flaws. At the other hand, the latter stages in our suggested StackGANs will not only add details to the low-resolution images generated from earlier stages but also correct issues within them. Several methods are being developed recently for extracting images from unstructured text.

[4] Constructed an AlignDRAW model by studying how to approximate the correlation between the text and the generating canvas.[5] used PixelCNN conditional to acquire images using text descriptions and object position constraints. [6] Approximate Langevin sampling method was used to get text-conditioned images. Their sampling approach however involves an inefficient iterative process of optimization. With conditional GANs, we successfully generated 64 ubiquitous images based on input text information for birds.

Though the implementation has been effective, there is a need to make a convenient UI webpage in order to use this technology for various practical applications. We use a webpage that runs the code created and puts the output on a different webpage. The STACK GANs we have used create multiple images though we only need one to show the result based on the description given by the user. Due to security reasons its complicated to create a website that can access local programs, hence there is a need to use either cloud, using an application (HTA) that provides the same UI and performs the required objectives. The model has been trained using various sample pictures of birds hence they are limited to giving just bird-based outputs. However, it is possible to train it with pictures of other animals provided that we give the attributes that can be used to describe these animals as different animals have different sets of attributes. Besides living things , non living objects can also be generated. The complexity increases proportionally to the increase in attributes that can be used in the description. It improves accuracy of the generated images but it increases execution time of the code, thus causing minor inconveniences. So as of now we limit the usage and applications to animals with simple easy to describe attributes and features.

3. Methodology

The StackGAN is implemented on a website using a webserver. The caption of a user will be sent to a server and from a server the description will be StackGAN code where the pictures will be generated. Once the picture is generated the picture will be sent back to the user using a web server. This is how StackGAN is implemented on a website using a web server and cloud services of any cloud service provide

A. StackGAN

StackGAN-v1 has two different networks, StackGAN-1 and Stack-2 GAN, which can model images with low to high resolution. Having the system wider. [8] is proposing to model a set of multi-scale image distributions to the current total Network. StackGAN-v2 is made up of several generators and discriminators. Small resolution pictures with high resolution pictures are generated from Various branches of the tree using [8]. The generator takes the image distribution to that scale for each dimension, and the discriminator calculates the likelihood of a sample originating from training images of that scale other than the generator. Additionally, generators and discriminators are trained and generators trained together to estimate the multiple distributions.

The working of StackGAN-v2 can be understood properly from the fig (1).

StageGAN-v2: it increases the low-resolution image from 4x4 to 64x64 to 128x128 and 256x256 and so on and complete the detailed image of the object description by reading the text description again and again producing a high resolution photo-realistic image by upsampling and down sampling the image as the software deconstructs the images through both upsampling and downsampling and by having blocks of residual and joining mechanisms with FC with reshape of the object .The jcu discriminator as we can observe in the diagram given below where the real image and false images are merged through downsampling we can take the necessary data as the software produces both the unconditional loss and conditional loss .This whole process is on a platform which has generators in a tree like structure finally producing the image that the user requires.

B. Implementation Blueprint

The fig (2) shows the blueprint of implementation of StackGAN into a website. We see in simple process where we open the webpage where the software lies for the user to enter the description of the object with the detailed description of the object which is now the second step. after the submission of the detailed are implemented the software runs the data and analyzes the data that the user has put in the input. The webpage begins the execution of the image that is being generated by the data given by the python program. The next step after the execution process is complete is when the data moves to another web page where the output of the python code will be sent in this particular new web page. The final step is where the generated output image will be shown based on the description given by the users input of the description of the object.

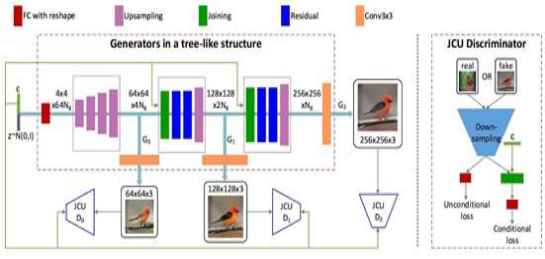

Figure1: Framework of StackGAN-v2

May-June 2020
ISSN: 0193-4120 Page No. 4694-4697

Figure 2: Flow chart of implementation

4. Dataset

The image data needed for the training the detection model was obtained from the Caltech-UCSD Birds dataset. The caltech-UCSD dataset comes with images of birds with around200 species. The CUB dataset consists of a list of different annotations of birds like given in Table 1.

Table 1. Bird dataset annotations

Annotation	Bird 1	Bird 2	Bird N
Forehead color	Black	Black	Black
Breast Color	White	White	White
Bread Pattern	Solid	Solid	Solid
Head Pattern	Plain	Capped	Plain
Back Color	White	White	Black
Wing Color	Grey/White	Grey	White
Leg Color	Orange	Orange	Orange
Size	Medium	Large	Medium
Bill Shape	Needle	Dagger	Dagger
Wing Shape	Pointed	Tapered	Long
….	….	….	….
Primary Color	White	White	White

5. Results

We get a set of images based on the descriptions given by the user, as shown below.

1. The description given was " A black bird with big belly and small beak" and the Result 1 is the generated image from the given description.

Result 1

2. The description given was "A white bird with black wings and black under tai" and the generated Image is Result 2 generated from the given description.

Result 2

3. The description given was "A brown bird with white spots and white patches" and The Result 3 is the generated image of the given description.

Result 3

6. Conclusion and Future Scope

The Web Server based Website is created to implement the StackGAN-v2. Using this implementation method,

May-June 2020
ISSN: 0193-4120 Page No. 4694-4697

any developer can create new things from this or Their can be improvements to this website from the open source community. The purpose of creating this website-based implementation is to convert this model into business model in future so that the work of artist or the content generator to generate the content easily using the UI created by us.

By use this as an API for developers, this framework can be additionally applied to the cloud services. This can also be implemented for specific purposes in the various mobile applications. It can be used as a business model to create photographs of humans or something for various applications. It can be used to create a picture of offenders in less time for national security agencies. In future it can support more variety of attributes and features for description given by the user.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In CVPR,2016

[2] I. P. Durugkar, I. Gemp, and S. Mahadevan. Generative multi-adversarial networks. In ICLR,2017M

[3] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein. Unrolled generative adversarial networks. In ICLR, 2017.

[4] E. Mansimov, E. Parisotto, L. J. Ba, and R. Salakhutdinov. Generating images from captions with attention. In ICLR,2016.

[5] S. Reed, A. van den Oord, N. Kalchbrenner, V. Bapst, M. Botvinick, and N. de Freitas. Generating interpretable images with controllable structure. Technical report,2016

[6] A. Nguyen, J. Yosinski, Y. Bengio, A. Dosovitskiy, and J. Clune. Plug & play generative networks: Conditional iterative generation of images in latent space. In CVPR, 2017.

[7] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In ICLR,2018.

[8] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, XiaoleiHuan, Dimitris N. Metaxas “StackGAN++: Realistic Image Synthesis with Stacked Generative Adversarial Networks.