January-February 2020
ISSN: 0193-4120 Page No. 16070 - 16074

Generative Adversarial Networks (GAN): A Survey

Anchal Kamra, Computer Science and Engineering, Department, Chandigarh University, Gharuan, Punjab, India

Kishan Kumar, Computer Science and Engineering, Department, Chandigarh University, Gharuan, Punjab, India

Article Info
Volume 82
Page Number: 16070 - 16074
Publication Issue:
January-February 2020

Article History
Article Received: *18 May 2019*
Revised: *14 July 2019*
Accepted: *22 December 2019*
Publication: *28 February 2020*

Abstract:
Recently, Generative Adversarial Networks (GANs) has emerged as a re- search interest in the field of deep learning. GAN works on the analysis of distribution of ground truth data samples and creating new samples by using that analysis. It workson two models, Generator and Discriminator, trained using backpropagation through a competitive procedure. GAN has gained enormous popularity in the fields of image processing, Vision computation, image synthesis, natural language processing, signal processing. This paper highlights basic architecture of GAN, and different variants of GAN being used byresearchers

Keywords: *Generator, Discriminator, Stochastic gradientdescent, Backpropagation, Lossfunction*

I. INTRODUCTION

In the past few decades machine learning has gained a huge popularity .The use of machine learning in every field has provided the boost to almost every industry it may be ranging from banking sector to autonomous self driven cars, from health to mar- keting sector. Deep learning is the branch of machine learning where we use the con- cept of the artificial neural network to construct a deep neural network having three or more hidden layers. When the network is fed with a large dataset and is provided by computational power it could outperform every machine learning techniques. Ma- chine learning and deep learning has also affected some of the major research areas in the field of image classification, computer vision etc. Some of the work related to deep learning deals with the use and advancement of Generative Adversarial Net- works(GANs) which was introduced by Goodfellowet al. (2014) [1]. The Generative Adversarial Network is the structure for estimating the generative model via the ad- versarial process. Here main thought of concern is to create a model which can gener- ate similar datasets after being trained on real datasets. Generative Adversarial Net- works [2] belongs to architecture that is generative in nature. After training the model this network architecture is able to generate new data points whose properties will look very much similar to the data points on which the network has been trained. To accomplish the goal it needs to create two models containing multilayer perceptrons [3]. First one out of these is called Generator (G) that capture a data distribution and other one is called discriminator model that gives the probability of the data created by G is similar to real training data. Both G and D models are multilayer perceptrons, and backpropagation can be used to train this entire system[4].

The Generative Adversarial Networks (GAN) framework carries two models, the generative model which perpetually generates the fake data like images and discrimi- native model which distinguish between the generated data and the ground truth data. It trains and edifies itself using the authentic dataset and also the fake engendered data. The Generative model has huge potential and it can mimic the set of data on which it is trained with tremendous accuracy. Following section of this paper will be focusing on Motivation for GAN followed by its Architecture and working models. This paper put emphasis on different types of GAN and their relatedworks

January-February 2020
ISSN: 0193-4120 Page No. 16070 - 16074

II. Motivation:

This astonishing concept can be used for generating new data points from the exist- ing ones. NVIDIA has generated a set of celebrity images using this concept [26]. The generated images look very similar to a celebrity. It is really hard to distinguish that people in the images are not a real celebrity and doesn't exist. Hence, we can use this concept to generate new data points and use them to make our machine learning or deep learning model more robust.

GANs can be used for creating anime characters. GAN can replace the artists that were used for creating anime characters as basic cost for game development and ani- mation production is very high. We can use Generative Adversarial Networks to cre- ate anime characters. Figure 1 shows the image to image translation by changing from zebra to horse and vice versa on two unordered images collections. It can be used for increasing resolution. Given a low-resolution image, it can generate the same image with a higher resolution [26]. If we have a pose and image of a person then we can generate the image of a person with the given pose [7].

Cycle Generative Adversarial Network [5] is a cross-domain transfer GAN also called as Unpaired Image to Image translation. This type of Generative Adversarial Network transforms images from one domain to another.

It can be used to create an image using the text description of the image known as Stack Generative Adversarial Network [6]. It is similar to a sketch artist of crime branch who listens to the person explaining the face of a person and just by listening he/she creates the image of that person.

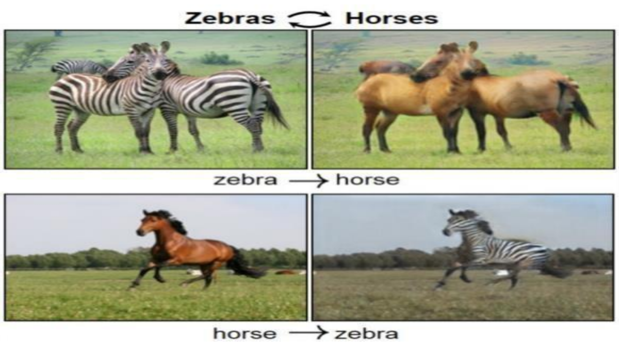

Fig.1. Image-to-Image translation using the given input image to create a desired image [5]

ARCHITECTURE

The whole architecture of Generative Adversarial Network has only two compo- nents a Generator (G) and a Discriminator (D) [1]. The Generator is used to create data points which will look similar to the real dataset, where as the Discriminator is used to distinguish between the real dataset and the fake data created by the Genera- tor. Both of them play a game the Generator will try to trick the Discriminator and on the other hand Discriminator will try to detect the forged data and give the Generator a very low probability. So, in other words both of them are an adversary of each other. We can create Generative and Discriminative model using supervision or without the supervision of desired output called as supervised and unsupervised learning respec- tively [14]. In supervised learning we have a label or real dataset and in unsupervised learning, we don't have a real dataset. Figure 2 shows block diagram ofGAN.

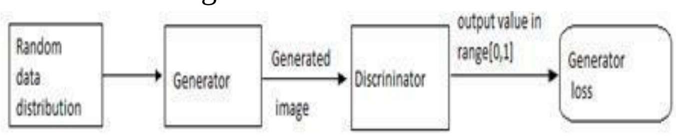

Fig.2. Block diagram ofGAN

1.1 **Loss function [1]** plays a crucial role in guiding deep learning to achieve a state of art. It is used to compute the difference between the estimated value produced by the deep layered neural network and the original dataset.

1.2 **Gradient Descent [3**] is the technique used to optimize the deep neural n et by using the loss. It uses differentiation method of Mathematics for optimization. In **simple** words, it gives the direction in which the deep neural network should optimize its parameters.

1.3 **Backpropagation [4]** is the algorithm which is used to optimize the parameters of the deep neural net by back-**propagating** the loss and calculates the value for the parameters using

January-February 2020
ISSN: 0193-4120 Page No. 16070 - 16074

the gradient descentmethod.

1.4 **Generative adversarial network [1]** uses a multilayer neural network to cre- ate a dataset which is similar to the real dataset given. To understand its intuition, suppose we have a dataset of images. The image could be of anything. Now, if we train our network on the dataset for a minimum of 300 epochs. After training, if we feed the network with a random distribution of numbers and reshape it into the net- work requirement then the output of the architecture would be an image that would look like a real image and it would be really hard to classify that the image is notreal.

1.5 Basic Working of the Generator and Discriminator **Model** Generatormodel

The generator model then outputs a generated data whose dimension is the same as the real data. Then, this output is given to the discriminator and it results the probabil- ity of difference. This probability value is used to compute the generator loss to help the generator learn. So, the generator is again optimized so that it can increase the probability of created data to be rated highly by the discriminator. Then, the generator is optimized by applying stochastic gradient descent.

$$\nabla_{\theta_g} \frac{1}{m} \sum_{i=1}^{m} \log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right)$$

This term is used to optimize the probability that the created data G (z) is highly similar to the original data, having m as number of samples. When we apply a gradi- ent to generator discriminator parameters are kept constant and vice versa.

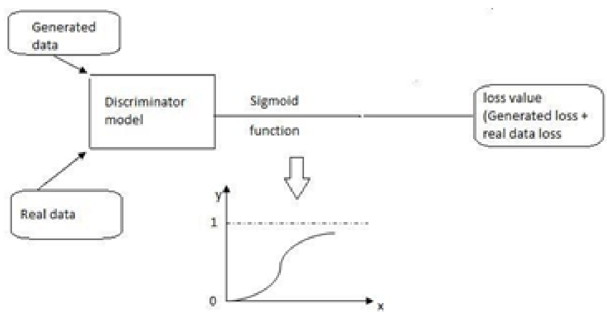

Fig 3: Discriminator model in the form of sigmoid function

A. Discriminator Model

At the first phase of training, the weights of discriminator are initialized randomly and it is then fed with original data and the data being created by generator model. Firstly, the discriminator computes the real data loss and generated data loss which is then back-propagates the total loss to adjust its weight using stochastic gradient de- scent. The loss is calculated using the output from the sigmoid function. Sigmoid function is used to get an output which is in the range between 0 and 1[1]. The dis- criminator is updated by increasing its stochastic gradient.

$$\nabla_{\theta_d} \frac{1}{m} \sum_{i=1}^{m} \left[\log D\left(\boldsymbol{x}^{(i)}\right) + \log\left(1 - D\left(G\left(\boldsymbol{z}^{(i)}\right)\right)\right)\right]$$

The term log D(x) correlates to optimizing the probability that the original data(x) rated highly. Training of GAN is done by traversing dataset n times forming different epochs.

III. Literature Survey

Generative Adversarial Network has attained a magnificent result in representation learning [7], image editing [8] and image generation [9]. In recent the techniques accepted the idea for conditional generation of images and its applications, such as 3D data [10], future prediction [11], panting [12], text2image [13]. GANs achieved great success because of its indistinguishable image from the original ones. GAN has vari- ous variants which had gained popularity in all these years. Some of these are:

1.6 Deep ConvolutionalGAN

Unsupervised representation learning was the main focus of DCGAN. At that pre- sent era supervised learning with CNN's gained a large attention but unsupervised learning with CNN's hardly got and attention. The work of Alec Radford & Luke Metz[7] came with the work that reduces the difference between supervised learning using CNN's and unsupervised learning using CNN's.

January-February 2020
ISSN: 0193-4120 Page No. 16070 - 16074

Figure 4 shows the generation of fake images using DCGAN.

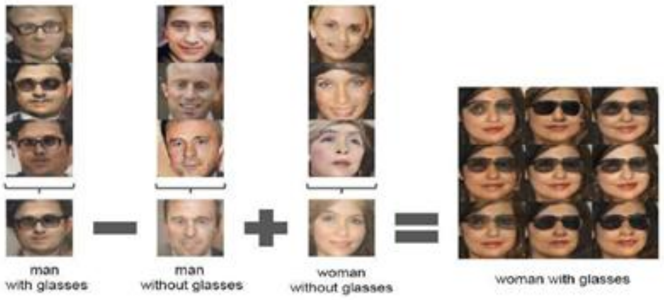

Fig.4. Generation of fake image using DCGAN [7]

1.7 InfoGAN

An InfoGAN is an unsupervised learning in which it splits the generator input into two parts:-the tradition noise vector and the latent vector. It can be used for SVHN dataset and can be used to describe the visual concept of hairstyle; emotions etc with celebrities face [18]. Information maximization Generative Adversarial Network adds only small computational amount to GAN and this model is easy to train. InfoGAN's mutual information can be applied with method like Auto-encoding variationalbayes [19].

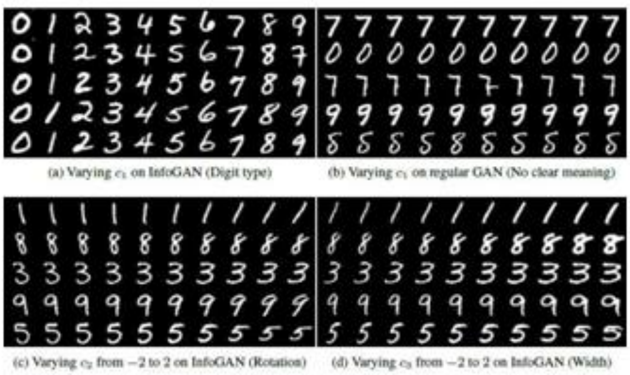

Fig.5. InfoGAN image clarification on digits [18].

1.8 Laplacian Pyramid ofGANs

The Laplacian Pyramid Generative Adversarial Network was first proposed by Denton et al [22]. The author tested the model in the following three data sets: a) LSUN data set, (b) CIFAR10 and (c) STL10. The methodology used for LAPGAN is generation of images in coarse-to-fine fashion. The Laplacian pyramid is an extrac- tion of Gaussian pyramid using downsampling and upsampling function [16, 17].

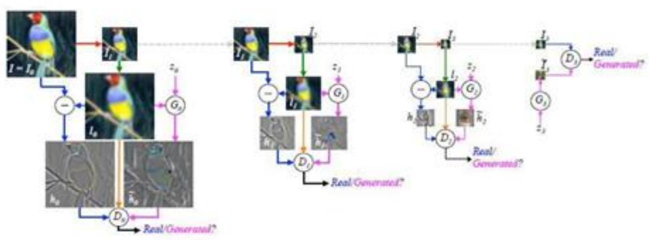

Fig.6. the structure of LAPGAN [16]

IV. WassersteinGAN

WGAN is a new algorithm is a new algorithm which improved the learning stability and avoided the problems of mode collapse and provided the useful curve for debugging and hyper parameter search. WGAN was proposed by Martin Arjovsky in his paper [20]. Gulrajaniet al[21] elaborated various problems of weight clipping of WGANs and also came up with an alternative idea that demonstrate an algorithm with more stability across a verity of architecture.

The following table shows the comparison between different GAN's used in super- vised and unsupervised learning. SaifuddinHitawala made a comparative study on different types of GAN's and there enlisted their different learning types, architectural network ,gradient updates etc[16].

Table 1:-the comparative study of GAN's [16]

Criteria	Vanilla GAN	CGAN	LAPGAN	DCGAN	AAE	GRAN	InfoGAN	BiGAN
Learning	Supervised	Supervised	Unsupervised	Unsupervised	Supervised, semi-supervised and unsupervised	Supervised	Unsupervised	Supervised and unsupervised
Network Architecture	Multilayer perceptrons	Multilayer perceptrons	Laplacian pyramid of convolutional networks	Convolutional networks with constraints	Autoencoders	Recurrent convolutional networks with constraints	Multilayer perceptrons	Deep multilayer neural networks
Gradient Updates	SGD with k steps for D and 1 step for G	SGD with k steps for D and 1 step for G	No updates	SGD with Adam optimizer for both G and D	SGD with reconstruction and regularization steps	SGD updates to both G and D	SGD updates to both G and D	No updates
Methodology / Objective	Minimize value function for G and maximize for D	Minimize value function for G and maximize for D conditioned on extra information	Generation of images in coarse-to-fine fashion	Learn hierarchy of representations from object parts to scenes in both G and D	Inference by matching posterior of hidden code vector of autoencoder with prior distribution	Generation of images by incremental updates to a "canvas"	Learn disentangled representations by maximizing mutual information	Learn features for related semantic tasks and use in unsupervised settings
Performance Metrics	Log-likelihood	Log-likelihood	Log-likelihood and human evaluation	Accuracy and error rate	Log-likelihood and error-rate	Generative Adversarial Metric (proposed)	Information metric and representation learning	Accuracy

V. Conclusion and FutureScope:

Till now generative adversarial networks have been mostly used on image data. And have performed very well on image datasets. Pix2Pix is an application of Gener- ative Adversarial Network where an image which doesn't have colors can be fed to the network and the network returns an image

January-February 2020
ISSN: 0193-4120 Page No. 16070 - 16074

with colors imposed on the real image. But the Generative Adversarial Network is not limited to an image only they have also worked very well with different concepts like Autoencoder called Adversarial Autoencoder [23]. We also have an extension to Generative Adversarial Network called Information Maximizing Generative Adversarial Network (InfoGAN) [18] which is used for facial and objectrecognition.

Generative Adversarial Networks has not been used with Reinforcement Learning [24] as it can generate data similar to real data. It could be used with Reinforcement Learning [24] to create a simulation environment similar to the real environment and we could train our agent on both environment. In the end, the sum total of the weights will be as good that the agent will be in the generalizedstate.

REFERENCES

1. Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, SherjilOzair, Aaron Courville, and YoshuaBengio. "Generative adversarial nets." In Advances in neural information processing systems, pp. 2672-2680.2014.
2. Zhao, Junbo, Michael Mathieu, and YannLeCun. "Energy-based generative adversarial network." arXiv preprint arXiv:1609.03126(2016).
3. Ghosh, Soumyadip, Mark Squillante, and EbisaWollega. "Efficient Stochastic Gradient Descent for Distributionally Robust Learning." arXiv preprint arXiv:1805.08728(2018)..
4. Golkar, Siavash, and Kyle Cranmer. "Backdrop: Stochastic Backpropagation." arXiv pre- print arXiv:1806.01337(2018).
5. Zhu, Jun-Yan, Taesung Park, Phillip Isola, and Alexei A. Efros. "Unpaired image-to-image translation using cycle-consistent adversarial networks." In Proceedings of the IEEE inter- national conference on computer vision, pp. 2223-2232. 2017.
6. Zhang, Han, Tao Xu, HongshengLi, Shaoting Zhang, Xiaogang Wang, XiaoleiHuang, and Dimitris N. Metaxas. "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks." In Proceedings of the IEEE International Conference on Computer Vision, pp. 5907-5915.2017.
7. Radford, Alec, Luke Metz, and SoumithChintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." arXiv preprint arXiv:1511.06434(2015).
8. Zhu, Jun-Yan, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. "Generative visu- al manipulation on the natural image manifold." In European Conference on Computer Vi- sion, pp. 597-613. Springer, Cham,2016.
9. Denton, Emily L., SoumithChintala, and Rob Fergus. "Deep generative image models us- ing a laplacian pyramid of adversarial networks." In Advances in neural information processing systems, pp. 1486-1494.2015.
10. Singh, H.. (2017). Overcoming challenges in 3D NAND volume manufacturing. Solid State Technology. 60. 18-21.
11. Chen, Bill & Bottoms, Bill & Armstrong, Dave & Isobayashi, Atsunobu. (2015). ITRS 2.0: Heterogeneous Integration. Solid State Technology. 58. 13-17.
12. Choe, Jeongdong. (2015). Comparison of 1Ynm NAND architectures and beyond. Solid State Technology. 58. 24-25.
13. Vogler, Debra. (2013). A Preview of Semicon West 2013. Solid State Technology. 56. 30-32.
14. Monahan, Kevin. (2006). Addressing 32nm half-pitch challenges with double-patterning lithography. Solid State Technology. 49. 28-30.