A Detailed Study on Generative Adversarial Networks

Shailender Kumar
Dept. of Computer Science and Engineering
Delhi Technological University
Delhi, India
shailenderkumar@dce.ac.in

Sumit Dhawan
Dept. of Computer Science and Engineering
Delhi Technological University
Delhi, India
sumitdhawan6@gmail.com

Abstract—**From past decades, along with increase in computing power, different generative models have also been developed in the area of machine learning. Among all such generative models, one very popular generative model, called as Generative Adversarial Networks, has been introduced and researched upon, in past few years only. It is based on the concept of two adversaries constantly trying to outperform each other. Objective of this review is to extensively study the GAN related literature and provide a summarized form of the studied literature available on GAN, including the concept behind it, its objective function, proposed modifications in base model, and recent trends in this field. This paper will help in giving a thorough understating of GAN. This paper will give an overview of GAN and discuss the popular variants of this model, its common applications, different evaluation metrics proposed for it, and finally its drawbacks, conclusion of the paper and future course of action.**

Keywords—deep learning; generative adversarial networks; neural network; generative model; image processing

I. INTRODUCTION

There has been many advancements in the field of machine learning, especially deep learning as more and more computing or processing power is becoming available. Deep Learning helps us in extracting high level, useful and abstract features from the input data and using those features in classification and generation tasks. This approach is commonly known as representation learning and is based on how the human mind learns anything. Concept of generative models (based on deep learning) forms the basis of Generative Adversarial Networks (or GANs). Traditional generative models like Restricted Boltzmann machine [15] and Variational Auto Encoder [14] were based on concepts like Markov chains and maximum likelihood estimation. Based on the distribution of input data, they estimate the distribution of generated data, but as a result of not being good in generalization, their performance and outcome is affected. A new concept in the field of generative models, called as Generative Adversarial Networks (GANs) was introduced in the year of 2014 by Goodfellow et al. [1]. It consists of one generator and one discriminator, which are adversaries of each other and thus, constantly trying to outperform each other and thus, improving themselves in the process of doing so. GANs are based on learning the joint probability distribution.

Image Synthesis is a problem which has been researched a lot upon. It means generating an image based on the hidden and visible features of an already existing image. GANs are being used largely in the area of image synthesis (or image processing, in general) as they have been proven to work really well with the images. GANs consist of two models which are trained simultaneously against each other. Generator is tasked with generating new data points based on how the data points are distributed in input samples and thus, deceiving the Discriminator with the generated/fake data points as real data points. Discriminator is tasked with catching the bluff of Generator by classifying the received data points as generated or a real (taken directly from the input sample space). It is similar to a zero-sum game between two opponents. Models are trained using back-propagation [17] and dropouts (to prevent over-fitting).

GANs can be classified based on their architecture and loss function used to train the generator [16]. It has been shown that GANs can produce pretty good and high quality images which seem convincing enough to be considered as real images. GANs have grown exponentially since they were first proposed in 2014. As of now, there are many variants of generative adversarial networks, each created specifically for a specific problem in multiple research areas like image synthesis, style transfer, image enhancement, image to text conversion, text to image conversion, detecting an object, etc., making GANs a hot topic for research purpose as well as in real world applications. Despite all the growth and research going on, GANs suffer from some problems and have shortcomings [9] like vanishing gradient, mode collapse, non-convergence, absence of universal performance evaluation metrics. Also, various solutions have been proposed for above mentioned drawbacks.

This survey analyses and presents theory, applications, shortcomings, state of the art variants of GANs and recent trends in this field. This survey is organized in parts as follows. Section II defines the background, structure and loss function of GANs. A comparison of variants of GANs is presented in Section III. Section IV discusses different evaluation metrics. The applications of GANs are given in Section V. Section VI

978-1-7281-5371-1/20/$31.00 ©2020 IEEE

discusses the shortcomings of GANs. Finally, Section VII gives the conclusions and future scope.

II. THEORY AND STRUCTURE OF GAN

A. *Basic Theory*

Behind the motivation for GANs, is Nash equilibrium of Game Theory [1], which is a solution for a non-cooperative game between two adversaries, in which each player already knows all the strategies of other player, therefore no player gains anything by modifying their own strategy.

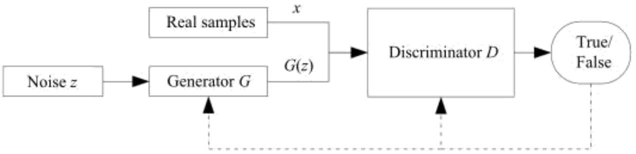

Fig. 1. Structure of Generative Adversarial Networks

The main aim of GAN is to arrive to this Nash equilibrium. Fig. 1 depicts the basic structural model of a standard GAN. Any function which can be differentiated can be used as the function for equations of generator as well as discriminator. Here, G and D are two differentiable functions that represent the generator and the discriminator respectively. Inputs given to D is x (real data), and z (random data or simply, noise). Output of G is fake data produced as per the probability distribution of actual data (or pdata), which is termed as G(z). If actual data is given as input to Discriminator, it should classify the input data as real data, labeling it 1. If fake or generated data is given as input to Discriminator, it should classify the input data as fake data labeling it 0. Discriminator strives to classify the input data correctly as per the source of data, On the other hand, Generator strives to deceive the Discriminator by making generated data G(z) similar and in line with the real data x. This game like adversarial process helps in improving the performance of both Discriminator and Generator slowly and gradually throughout the process. Therefore, slowly, Generator is able to generate better images that look more real, because it has to fool the improved and more efficient Discriminator (when compared to the previous iteration of training).

B. *Loss Function*

In this part of the section, loss function and learning methodology of GANs will be discussed. Following is the mathematical representation of loss function (or objective function) of GANs, as discussed in [1].

$$min_G max_D V(D, G) = E_{x\sim p(x)}[log\, D(x)] + E_{x\sim p(z)}[log(1\text{-}D(G(z)))] \quad (1)$$

where,

x - Obtained by sampling the distribution of real data or p(x).

z - Obtained by sampling the distribution of prior data or p(z) (which can be Gaussian or Uniform distribution).

E[x] - Expected value of any random variable x.

D(x) - The probability that x is sampled from actual data and not from the generated data.

Main aim of Discriminator is to maximize the objective function by minimizing D(G(z)) along with maximizing D(x), which signifies Discriminator is classifying real data as real and fake data as fake. Main aim of Generator is to minimize the objective function by maximizing D(G(z)) which signifies that Discriminator is classifying generated data (which is fake) as real. The training data consists of real data as well as generated data. Discriminator is initially trained on real as well as fake data. After that, its weights are made non trainable. Then the generator is trained. This process is repeated until required level of performance is achieved. In initial phase of learning, due to poor performance of Generator, Discriminator can catch the lie and reject generated samples easily as they are easily distinguishable from the real samples. Therefore, saturation of log(1 − D(G(z))) happens and derivative tends to 0, preventing gradient descent from occurring. So, to correct this, Generator is now trained to maximize the value of log(D(G(z))) instead of minimizing the value of (1 − D(G(z))). This modified objective function results in the derivative to be high and provides much better gradients in the initial phases of learning

III. GAN MODELS

Since the idea of GAN was proposed, many variations have been done in the original model, resulting in different models of GANs. These variations include the improvement of structure for efficiency, changes done for any specific application or problem statement like, style transfer from one image into another [18], image enhancement [13], completing an incomplete image [20-21], and generating an image from text [19].

A. *Fully Connected or Vanilla GAN*

It was first introduced as a baseline model for GANs in 2014 by Goodfellow et al. [1]. As the overlap of probability distributions of actual data and fake data is very little, the measure of similarity between two distributions, also known as the Jensen-Shannon divergence, may become constant, which results in the problem of vanishing gradient. It doesn’t perform that well in the case of more complex images.

B. *Deep Convolutional GAN (DCGAN)*

Another variant of GANs was proposed by Radford et al. (2015) [4], known as Deep Convolutional Generative Adversarial Networks (DCGANs), which is based on Convolutional Neural Networks with following changes:

- Removing fully connected hidden layers.
- In the generator, a fractional-strided convolutions are used instead of pooling layers. In the discriminator, strided convolutions are used instead of pooling layers.
- Using the concept of batch normalization in generator as well as discriminator.
- Applying ReLU activation function in all but the last layer of generative model. LeakyReLU activation function is applied in each and every layer of the discriminator.

C. Wasserstein GAN (WGAN)

Wasserstein GAN (WGAN) was proposed by Arjovsky et al. [5] to resolve the issue of vanishing gradient. Instead of the Jensen-Shannon divergence, it uses the Earth-Mover distance, for calculating the similarity between probability distribution of real or actual data and fake or generated data. Discriminator is represented using f, which is a critic function, and is built on Lipschitz constraint. WGAN, though useful in a comparatively stable training of GANs, still produces data samples which are low in quality and even sometimes fail in the process of converging.

D. Wasserstein GAN with Gradient Penalty (WGAN-GP)

This model can be considered as an improvement to the basic WGAN, proposed by Gulrajani et al. [12]. They found that the weight clipping in GAN, which is done in order to apply a Lipschitz constraint on the critic, leads to failure in training. This causes abnormal behavior. So an alternative was proposed to apply the Lipschitz constraint on critic, that is, it works by penalizing the value of norm of the gradient of the f (or critic function) with respect to the input, instead of clipping of weights. This method tends to converge quickly and generates samples of relatively high quality when compared to WGAN version with weight clipping.

E. Least Square GAN (LS-GAN)

In conventional GANs, during the process of back-propagation, sigmoid cross entropy loss function is used. However, sometimes, during the learning process, the issue of vanishing gradient occurs due to this particular loss function. As a solution, Mao et al. [22] came up with LS-GAN. It uses the least square loss function. LS-GANs generate higher quality images and also are more stable during the whole learning process compared to normal GANs.

F. Semi-GAN (SGAN)

In basic GAN, during training, only one class label is required which is used to specify data source (real or fake). Odena [10] proposed Semi GAN (SGAN) by further adding class labels (say, N) of actual data while discriminator D is trained. After training, D will classify the input to one of the N+1 classes, where the extra class is used to specify the data source as per G. It is shown that this method can create a better classifier which generates samples of high quality when compared to standard GAN, as it is trained to learn the features and associates them with correct class labels.

G. Conditional GAN (C-GAN)

In conventional GAN, only latent space is provided to generator. The conditional GAN, as proposed by Mirza [11] changes that by providing an additional parameter (label y) along with latent space, to the generator and train it to produce corresponding images. Discriminator is provided with real images and label as its input, to distinguish real images better. It is shown that this model can produce digits similar to those of MNIST dataset, conditioned on class labels (0, 1, 2, 3…9). Hence, the name is Conditional GAN.

H. Bidirectional GAN (BiGAN)

In normal GANs, the generator learns by mapping latent feature vector to real data probability distribution. But it lacks an efficient mechanism which maps the real data to data in latent space. Bidirectional GANs (BiGANs) was proposed by Donahue et al. [23]. It maps the actual data probability distribution on to latent space, hence helping to learn to how to extract relevant features.

IV. EVALUATION METRICS

Apart from the manual inspection and evaluation, of samples produced by Generator module of GANs, there are some quantitative measures like Average Log Likelihood [24], Inception Score (IS) [9], Wasserstein metric [5], Frechet Inception Distance (FID) [25], etc. IS should be high and FID should be low for high quality generated images. Based on these evaluation metrics, GAN model's performance can be better judged and thus can be improved by introducing the required modifications in model.

Log-likelihood (also called as Kullback-Leibler divergence or KL divergence) is one of the widely used standard metrics for evaluating and measuring the performance of generative models [24]. It helps in measuring the likelihood of generated distribution being consistent with actual data distribution. Maximum likelihood or 0 KL divergence will produce perfect samples which will seem real.

Inception Score is probably one of the most used evaluation metric for GAN evaluation, and it was proposed by Salimans [9]. A pre-trained neural network (for example, Inception v3 model) is used to capture the relevant and important properties of generated samples, i.e., image quality and image diversity. High image quality is represented by narrow distribution of label and high image diversity is represented by uniform distribution of labels of all samples combined. As both distributions are very much dissimilar (one is narrow, another is uniform), KL divergence should be high, further giving good Inception score.

$$KL\ divergence = KL(C||M) = p(y|x) * (log(p(y|x)) - log(p(y))) \quad (2)$$

where,
C is conditional probability distribution, p(y|x) and,
M is marginal probability distribution, p(y)

Inception score doesn't measure the variations in fake images if contrasted with actual images, therefore Heusel et al. [25] introduced Frechet Inception Distance. It works on the features and transforms generated data and maps it to feature space. This feature space is obtained by the last pooling layer, just before output classification. It is summarized by calculating the covariance as well as mean for fake and actual data. By calculating this Frechet Inception Distance (also known as Wasserstein-2 distance) between these Gaussian distributions, quality and variation of produced data is measured.

As of yet, there is no unanimous decision regarding which one is a better evaluation measure. Different scores focus on different aspects of the image generation process. But some metrics seem more plausible than others, like FID has the advantage of being more robust to noise. And as claimed in [25], FID can tell us how similar are real and fake images, and this measure is considered more efficient than IS.

V. APPLICATIONS OF GANS

GANs are being used widely in fields like Image processing and Natural Language Processing. Following are few of the main applications of GANs apart from classical application of generating data similar to the input data.

1) Inter-domain conversion of content of one image to another using CGAN was proposed by Isola et al. [18]. It is named pix2pix which is effective at generating photos from label maps and colorizing images.
2) Zhu et al. [18] propose CycleGAN, which can also be used for translating image from one domain to another, in the absence of a training example of paired images. It can perform style transfer, photo enhancement, object transfiguration, attribute transfer, etc.
3) Solution to the problem of Image super-resolution was provided in [13], where Ledig et al. present SRGAN. VGG network [26] is used as the discriminator. It is shown in [13] that it is possible to generate photorealistic image with high resolution. But the texture information which SRGAN generates, doesn't seem that much real, and is noisy. Therefore, Wang et al. [29] propose an Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN) which aims to improve the network's architecture.
4) Zhang et al. [28] propose that GAN can generate such sentences which seem too real by utilizing the concept of long short term memory (LSTM). Li et al. [30] use GANs to perform speech to text conversion by capturing the relevance of the dialogue. SeqGAN [31] uses reinforcement learning to not only generates speech but even poems and music.
5) Antipov et al. [27] introduced a model on the concept of GAN that can be used for automatic face aging. It produces synthetic images. Facial attributes are modified in similar kinds of work but in this, emphasis is given on preserving basic facial attributes and original person's identity in aged version of the face.

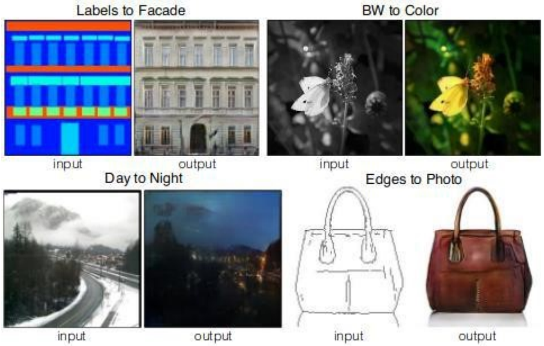

Fig. 2. Few examples of applications of GANs [8]

VI. DRAWBACKS

The main shortcoming in the training process of GANs is that there is always a risk of mode collapse. It occurs when data produced by GANs is mostly focused on very less modes or sometimes even on one single mode, which results in comparatively less diverse samples. A solution to the above problem is batch regularization or mode regularization. In this, decent size of batches of data samples or various data points are included so that the diversity is improved. Another solution is proposed in [33], in which the samples generated by different models are combined together. Moreover, optimizing the objective function (like done in WGAN [5]) can resolve the problem as well.

The lack of stability in the process of training is a big issue in itself. The GAN model's parameters oscillate, destabilize and never converges to the Nash equilibrium, that is each opponent always countermeasures other opponent actions, making the models harder to converge. In contrast to the other generative models, GANs evaluation problem is much more challenging as there is no consensus on what evaluation metrics should be used for GANs [34]. Another shortcoming of GANs is the problem of diminished gradient. It occurs if the discriminator gets very powerful while training, causing the gradient of generator's function to reduce and slowly vanish, and in turn, the generator learns nothing. This imbalance between models of generator and discriminator, results in overfitting. Also, GANs are very much sensitive to the selection of hyper parameters.

VII. CONCLUSION AND FUTURE SCOPE

This paper analyses and summarizes the background of GANs, its basic theory, structure, variants, evaluation metrics, applications, shortcomings and future scope. It gives an overall and broad understanding of the literature present on GAN. A wide variety of applications of GANs is shown in this paper. New as well as better solutions to new and existing challenges in GANs are yet to be explored with an aim to increase the efficiency of GANs. The field of GAN is a promising research area, but even with all the development in its research area, it has its own problems like unstable training, non-convergence, lack of consensus on an evaluation metric, requirement of more computing power and the model's complexity. In conclusion, GAN is an interesting and useful research field with many applications but also, a lot of work needs to be done to overcome existing challenges, as it is still comparatively a new field.

New work is being continuously done to overcome the limitations of GANs. For example, WGAN can resolve the mode collapse problem as well as the training instability problem, but only partially. Therefore, preventing the problem of mode collapse in GANs remains an open research problem. Also, other research areas include the presence of Nash equilibrium and the theory of the convergence of a GAN model. GANs are being widely utilized in the area of Computer Vision but comparatively not that much in other fields like Natural Language Processing. This limitation is because of the different properties related to image and non-image data. GANs can be used for interesting applications in various fields, therefore research is going on for that and also on how to increase the efficiency and improve the performance of GAN.

REFERENCES

[1] I. Goodfellowet al., ''Generative adversarial nets,'' in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf.

[2] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. New York, USA: MIT Press, 2016.

[3] I. Goodfellow, "NIPS2016 tutorial: generative adversarial networks," arXiv: 1701.00160, 2016.

[4] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," arXiv: 1511.06434, 2015.

[5] Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein gan." arXiv preprint arXiv:1701.07875 ,2017.

[6] L. J. Ratliff, S. A. Burden, and S. S. Sastry, "Characterization and computation of local Nash equilibria in continuous games," in Proc. 51st Annu. Allerton Conf. Communication, Control, and Computing (Allerton), Monticello, IL, USA, 2013, pp. 917−924.

[7] Hitawala, Saifuddin. "Comparative Study on Generative Adversarial Networks." arXiv preprint arXiv:1801.04271, 2018.

[8] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In CVPR, 2017

[9] Salimans, Tim, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. "Improved techniques for training gans." In Advances in Neural Information Processing Systems, pp. 2234-2242. 2016.

[10] A. Odena, "Semi-supervised learning with generative adversarial networks," arXiv: 1606.01583, 2016.

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

[12] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. "Improved Training of Wasserstein GANs". arXiv preprint arXiv:1704.00028, 2017

[13] Ledig, Christian, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P. Aitken et al. "Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network." In CVPR, vol. 2, no. 3, p. 4. 2017.

[14] Kingma, Diederik P., and Max Welling. "Auto-Encoding Variational Bayes." ArXiv:1312.6114 [Cs, Stat], May 2014. arXiv.org, http://arxiv.org/abs/1312.6114.

[15] Fischer, Asja, and Christian Igel. "An Introduction to Restricted Boltzmann Machines." Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, edited by Luis Alvarez et al., Springer, 2012, pp. 14–36. Springer Link.

[16] Wang, Zhengwei, et al. "Generative Adversarial Networks: A Survey and Taxonomy." ArXiv:1906.01529 [Cs], June 2019. arXiv.org, http://arxiv.org/abs/1906.01529.

[17] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, ''Learning representations by back-propagating errors,'' Nature, vol. 323, pp. 533-536, Oct. 1986.

[18] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," arXiv preprint arXiv:1703.10593v6, 2017.

[19] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, and H. Lee, "Generative adversarial text to image synthesis," arXiv preprint arXiv:1605.05396, 2016.

[20] Z. Chen, S. Nie, T. Wu, and C. G. Healey, "High resolution face completion with multiple controllable attributes via fully end-to-end progressive generative adversarial networks," arXiv preprint arXiv:1801.07632, 2018.

[21] Y. Li, S. Liu, J. Yang, and M.-H. Yang, "Generative face completion," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 3911–3919.

[22] Mao, Xudong, et al. "Least Squares Generative Adversarial Networks." ArXiv:1611.04076 [Cs], Apr. 2017. arXiv.org, http://arxiv.org/abs/1611.04076.

[23] Donahue, Jeff, et al. "Adversarial Feature Learning." ArXiv:1605.09782 [Cs, Stat], Apr. 2017. arXiv.org, http://arxiv.org/abs/1605.09782.

[24] Hamid Eghbal-zadeh, Gerhard Widmer, "Likelihood estimation for generative adversarial networks", 2017.

[25] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, in: Advances in Neural Information Processing Systems, 2017, pp. 6629–6640.

[26] Simonyan, Karen, and Andrew Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." ArXiv:1409.1556 [Cs], Apr. 2015. arXiv.org, http://arxiv.org/abs/1409.1556.

[27] Antipov et al., "Face aging with conditional generative adversarial networks", 2017.

[28] Y. Z. Zhang, Z. Gan, and L. Carin, "Generating text via adversarial training," Proc. Workshop on Adversarial Training, Barcelona, Spain, 2016

[29] Wang, Xintao, et al. "ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks." ArXiv:1809.00219 [Cs], Sept. 2018. arXiv.org, http://arxiv.org/abs/1809.00219.

[30] J. W. Li, W. Monroe, T. L. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," arXiv: 1701.06547, 2017.

[31] L. T. Yu, W. N. Zhang, J. Wang, and Y. Yu, "SeqGAN: sequence generative adversarial nets with policy gradient," arXiv: 1609.05473, 2016.

[32] Killoran, Nathan, et al. "Generating and Designing DNA with Deep Generative Models." ArXiv:1712.06148 [Cs, q-Bio, Stat], Dec. 2017. arXiv.org, http://arxiv.org/abs/1712.06148.

[33] A. Ghosh, V. Kulharia, V. P. Namboodiri, P. H. S. Torr, and P. K. Dokania, ''Multi-agent diverse generative adversarial networks,'' in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 8513–8521

[34] Borji, Ali. "Pros and Cons of GAN Evaluation Measures." ArXiv:1802.03446 [Cs], Oct. 2018. arXiv.org, http://arxiv.org/abs/1802.03446.

[35] Vijayakumar, T. (2019). Comparative study of capsule neural network in various applications. Journal of Artificial Intelligence, 1(01), 19-27