Divya Hitesh Chhipani
200369536

[ECS795P] Deep Learning and Computer Vision
Critical Analysis Report on Unsupervised Learning by GANs

Unsupervised Learning

In the field of Deep Learning, models tend to perform better when there is an availability of labelled data in huge amounts. However, in many real-world applications, it is difficult to collect or generate more labelled data and if we try, it's an expensive and time-consuming task. For these reasons, Xie et al.[2] were among the first to investigate unsupervised learning algorithms which are designed to derive insights from the data without any supervision which is usually provided in the form of labelled data. Some of these algorithms are Clustering, ResNets[3], Autoencoders[4], and GANs[1].

Generative Adversarial Networks

Generative modelling is an unsupervised learning task that involves automatic discovery of patterns from random input data for the model to generate new samples similar to those from original dataset without accessing the original dataset. The model consists of two neural networks that are pitted against each other. The generator tries to generate the samples as close to the dataset samples and the discriminator network is assigned with the task to distinguish between the samples created by the generator and the corresponding sample from the dataset. Finally, if the training for generator goes well, the discriminator starts to classify fake data as real resulting in a decrease in its accuracy. Both the generator and discriminator are neural networks arranged in a way that the output from the generator is connected directly to the discriminator's input which also takes in dataset samples for classification. Through backpropagation, the discriminator's classification provides a feedback that the generator uses to update its weights. GANs are used in various tasks such as image reconstruction, generating images from a given text, super-resolution of images, natural language applications such as text generation, text-to-speech applications to name a few.

Main trends and Key ideas

One of the basic objectives of GANs is to **generate high quality realistic images**. Variants have been suggested to overcome limited capacity of the vanilla GAN architecture. DCGAN[5] introduced the deconvolutional process while LAPGAN[6] uses the up-sampling process enabling the model to have larger capacity and also producing better quality images. GAN variants like BEGAN[7], PROGAN[8], SAGAN[9] and BigGAN[10] have some modifications on the loss function to enhance the image quality e.g., use of Wasserstein distance as loss function. Other architectures use different models for the discriminator such as use of autoencoder in BEGAN which compares generated images and real images in pixel level, helping generator produce easy-to-reconstruct data. PROGAN utilizes a deeper architecture and the model is growing with the training progressing which improves the learning stability for discriminator and generator making it easier for the model to learn how to produce high resolution images. SAGAN employs spectral normalization for better image quality. BigGAN, an extension to SAGAN, employs a deeper model with larger batch size to generate high resolution images.

Producing diverse images is the most challenging problem for GANs. In terms of architecture-variant GANs, only SAGAN and BigGAN address such kind of issue. Benefiting from self-attention mechanism, approaches such as those in SAGAN and BigGAN integrate self-mechanisms to both discriminator and generator helping to process large receptive field, which helps GANs a lot in terms of learning multi-class images.

Discussing about **improving discriminative ability of GANs**, the architecture-variants focus on adding more encoding information to GANs compared to the original GAN. CGAN[11] is the one of the initial models to introduce usage of encoded labels together with images as input to the discriminator and noise input to the generator, in which the input noise and the input image are now encoded with labels. InfoGAN[12] is based on CGAN, which has one classifier acting as a discriminator for distinguishing real and fake samples, and an additional classifier to classify the input images (including generated images). A further extension to InfoGan is AC-GAN[13] which additionally considers real images conditioned by labels and performs classification for both generated images and real images. BiGAN[14] introduces learning the inverse mapping, which also shows the improvement on the quality of generated images.

Issues of GANs

Vanilla variant of GANs had several issues mentioned in the original paper the most important being that it was not State of the Art at that time and did not perform better than most models available at that time. The original GAN was only applied to MNIST, Toronto face dataset and CIFAR-10 because of **its limited capacity** of the architecture. Deconvolution and up-sampling process have proven useful for this issue.

Another issue highlighted for GANs is related to **convergence** during training. As the generator gets better with training, the discriminator performance gets worse as the discriminator cannot easily differentiate between real and fake samples. Eventually, if the generator succeeds, then the discriminator has a 50% accuracy. This progression poses a problem as the discriminator feedback gets less meaningful over time and the generator uses this random feedback resulting in poor quality output. Two solutions to this are – adding random noise to the discriminator input and to use regularization for discriminator’s network weights.

In another scenario, if the discriminator is too good, then generator training can fail due to **vanishing gradients** as the discriminator is unable to provide enough information for the generator to make progress. The introductory paper on GANs[1] proposed a modification to minimax loss to deal with vanishing gradients. Also, Wasserstein loss is proposed[15], but these loss functions are changed according to the architecture thus it is architecture-specific loss which cannot generalize to other architectures.

As mentioned previously, diversity in generated images is one of the goals while training GANs. When the generator finds one or a few samples regardless of the input, and the discriminator does not distinguish between a real input and generator’s output; the generator may lazily learn to produce only that output. This looks like an ideal progression in training but it may lead to a failure which is described as **Mode collapse**. This situation is somewhat mitigated by using more diverse set of samples, or employing Wasserstein loss helping the discriminator to not get stuck in a local optima.

Evaluation for GANs is mainly divided into two main types - qualitative referring to the visual quality of generated images from a human perspective; and quantitative measures to evaluate model based on criterions like overfitting, low diversity of generated samples, and mode dropping for GANs. Human annotation is the most commonly used qualitative measure which is time-consuming and expensive. Inception Score (IS)[16], and Frechet Inception Distance (FID)[17] are proposed as alternative measures to evaluate visual quality of images. Examples for Quantitative metrics are sliced Wasserstein distance (SWD)[8], kernel maximum mean discrepancy (MMD)[18], classifier two-sample tests (C2ST)[19]. Even with existence of an array of metrics, it is still difficult to compare the performance of the models and evaluate quality of images generated by GAN at par with the human judgement.

The performance of GANs which are conditioned on labels such as CGAN and its successors depends on the dataset being well-labelled, which may pose challenges on some real-world applications.

Variants of GANs which cater to specific task being handled and hence, there is no general version of GAN which can be applied to a variety of tasks. We have majorly seen applications of GANs in the field of Computer Vision. It will be interesting to work towards using GANs for multi-modal data catering to a broader set of Natural Language applications for Virtual agents and Machine Translation.

References –

1. Goodfellow Ian, et al. “Generative Adversarial Nets”. Advances in neural information processing systems. 2016.
2. J. Xie, R. B. Girshick, and A. Farhadi. Unsupervised deep embedding for clustering analysis. In ICML, 2015.
3. K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385,2015.
4. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. chap. Learning Internal Representations by Error Propagation, pp. 318–362. MIT Press, Cambridge, MA, USA (1986).
5. A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv 1511.06434, 2015.
6. Denton, E. L., Chintala, S., Fergus, R., et al. (2015). Deep generative image models using a laplacian pyramid of adversarial networks. In Advances in neural information processing systems (pp. 1486– 1494).
7. Berthelot, D., Schumm, T., and Metz, L. BEGAN: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717, 2017.
8. Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196, 2017.
9. H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, “Self-attention generative adversarial networks,” arXiv preprint arXiv:1805.08318, 2018.
10. A. Brock, J. Donahue, and K. Simonyan, “Large scale GAN training for high fidelity natural image synthesis,” arXiv preprint arXiv:1809.11096, 2018.
11. Mehdi Mirza and Simon Osindero. “Conditional Generative Adversarial Nets”. arXiv preprint arXiv:1411.1784 (2014)
12. X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, “InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets,” in Advances in neural information processing systems, 2016, pp. 2172–2180.
13. A. Odena, C. Olah, and J. Shlens, “Conditional image synthesis with auxiliary classifier gans,” in Proceedings of the 34th International Conference on Machine Learning, vol. 70. JMLR, 2017, pp. 2642–2651.
14. J. Donahue, P. Krahenb ¨ uhl, and T. Darrell, “Adversarial feature learning,” ¨ arXiv preprint arXiv:1605.09782, 2016
15. M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein GAN. ArXiv e-prints, Jan. 2017.
16. T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training GANs,” in Advances in Neural Information Processing Systems, 2016, pp. 2234–2242.
17. M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and 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. 6626–6637.
18. A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Scholkopf, and A. Smola, “A kernel two-sample test,” ¨ Journal of Machine Learning Research, vol. 13, no. Mar, pp. 723–773, 2012.
19. A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Scholkopf, and A. Smola, “A kernel two-sample test,” ¨ Journal of Machine Learning Research, vol. 13, no. Mar, pp. 723–773, 2012.