Deep Generative Models

Anirudh Seth
aniset@kth.se

September 2020

1 Introduction

Generative modeling has emerged as a major branch of unsupervised machine learning. Such models are immensely powerful in learning useful data distributions which also allows them to generate synthetic data. Recent advances in neural networks architectures combined with the progress in computational power and optimization techniques have enabled such networks to model high-dimension data including text [18] and speech [9] and videos and images [12],[24],[14] possible.In this essay , I review three different neural network architectures of generative modeling and there application to images-:

1. Variational autoencoders (VAEs) [15] - NVAE [24]
2. Flow-based generative models [6],[5] - GLOW [14]
3. Generative adversarial networks [8] - StyleGANv2 [12]

The applications of above models range from text to image translation [17] reconstruction of 3d models using images [19],drug discovery [1], music generation [7] etc.

2 Methods

2.1 Glow: Generative Flow with Invertible 1x1 Convolutions [14]

Glow is a flow based model which builds on the previous works of NICE [5] and RealNVP [6]. Given latent variables z , input x such that $x = g_\theta(z)$ and $z \sim p_\theta(z)$, flow based models assume g function to be invertible or a composition of other invertible functions (a.k.a normalizing flow [5]) such that ,inference on latent variable is $z = g_\theta^{-1}(x)$ The change of variable formula when applied to the above setup gives a relationship between the input density,p(x) and latent space density p(z).

$$\log p_\theta(\mathbf{x}) = \log p_\theta(\mathbf{z}) + \log |det(d\mathbf{z}/d\mathbf{x})| = \log p_\theta(\mathbf{z}) + \sum_{i=1}^{K} \log |det\,(d\mathbf{h}_i/d\mathbf{h}_{i-1})| \tag{1}$$

where $\mathbf{h}_0 \triangleq \mathbf{x}$ and $\mathbf{h}_K \triangleq \mathbf{z}$. The log determinant of the jacobian can be computed efficiently as shown in previous work [3] and [22]. Each flow step g_i in GLOW consists of three operations
1.ActNorm - this activation normalisation layer is used to scale and shift the activations replacing the use of batch normalization as in RealNVP [6]. This allows the use of minibatch of size 1 per processing unit for larger scale images without performance degradation. The scale and the bias are trainable parameters.
2.1x1 convolution: In order to ensure that each dimension can affect other dimensions , a permutation of the variables is implemented in the flow steps. This can be done by reversing the order of the input channels by using a fixed random permutation as implemented in RealNVP. In Glow ,the authors realize this as a special case of a linear transformation with a square matrix i.e 1x1 convolution with equal number of input and output dimensions where the weights are initialized as random rotation matrices. During training these weights are learnt as opposed to fixed values as in RealNVP.
3.Affine Coupling Layer Following the flow architecture in RealNVP [5], the authors use a sequence (which can be stacked) of bijective transformations known as affine coupling layer.The log-determinant are computationally efficient to compute as shown in previous work [5].

The above flow operations are then implemented on a multi-step architecture as in RealNVP [6] with L levels each operating on a varying scales of images ,each level having K flow steps.

The intuition is to transform the dependent distribution over pixels of an image to an independent distribution over latent variables, apply MLE estimate basis our assumption. Given the above mathematical set up, learn the data distribution (Equation: 1)and in turn produce high quality synthetic data.

2.2 Analyzing and Improving the Image Quality of StyleGAN [12]

GAN, Generative Adversarial Networks is a generative model that consists of two networks- a Generator- that generates synthetic images from the latent variables, and a Discriminator which distinguishes between the images synthesized by the generator and the real images.The two networks are then trained together in an adversarial fashion until the images from the generator are realistic enough.StyleGAN [11] is an implementation of the above architecture, instead of feeding the latent variables $z \in \mathcal{Z}$ to the generator directly, it uses a mapping network that first map points from latent space to an intermediate latent space $w \in \mathcal{W}$. This new latent space is then used to produce styles that control the different layers of the generator.StyleGANv2 [12] proposes some architectural modifications to the StyleGAN [11] that further improves its results as mentioned below.

- The original model produces images that have blob like artifacts which become quite prominent with the increasing resolution. The authors attribute the problem to the AdaIN normalisation , which normalizes the mean and the variance of every feature map separately losing information about co-relation. This operation can be broken into a normalisation and modulation step as shown in Figure 1-b where the noise (b and B in figure) and bias respectively are applied within a style block (shown in grey).The authors empirically observe more predictable results by moving these operations outside the style block (as seen in Figure 1-c) and only operating on the standard deviation.The idea of instance initialisation is to remove the effect of scaling factor (on the weights) from the output of convolution feature maps.The authors make a statistical assumption -variables in the input activation's are i.i.d with a unit standard deviation. With the above assumption the authors show that architecture in figure:1-c) can be simplified into a single mod->demod->convolution step as shown in figure :1-d). The proposed architecture although weaker since it relies on statistical approximations removes the blob artifacts while still maintaining good performance in terms of FID [11].

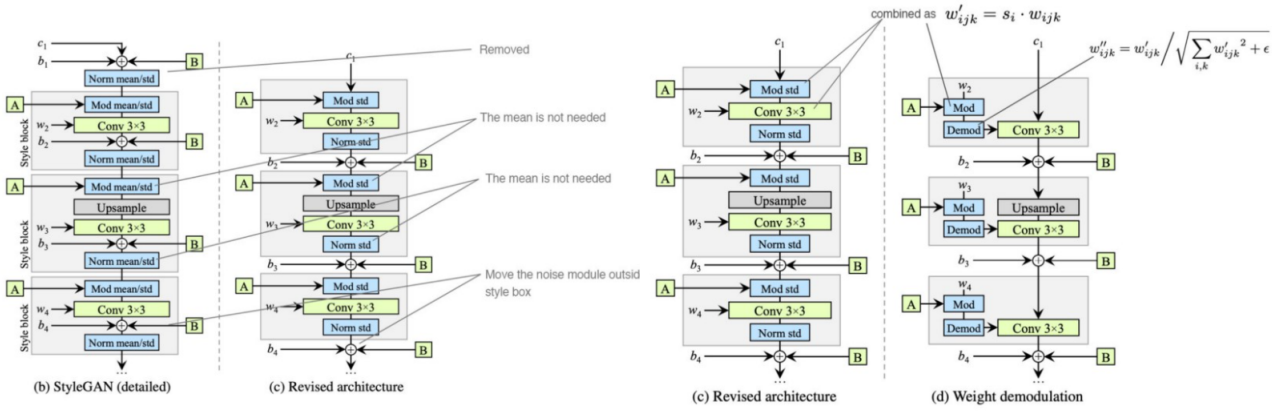

Figure 1: StyleGAN architecture and the modifications applied in StyleGAN2. Image Source [10]

- The authors also observe a relationship between image quality and perceptual path length (PPL) 1) which they consider is a additional metric to measure performance of GAN.Motivated by the results, they implement a path length regularization scheme that produces a smoother latent space , is more reliable and is also easier to invert. This (inversion) essentially means that it makes it easier to identify which network generated the image which can have unique applications as discussed in further sections.

- Another artifact of the original model is when features like teeth or eyes remain stuck and don't move smoothly over a changing image.The key issue identified is the progressive growing in the

model which has a strong location preference for details. StyleGANv2 [12] instead uses skip connections that improve the PPL score and a residual discriminator that improves FID performance. This revised architecture without progressive growing removes such artifiacts.

2.3 NVAE: A Deep Hierarchical Variational Autoencoder [24]

NVAE [24] is a deep hierarchical VAE that uses a bidirectional encoder , generator to produce high quality images.The principal of a hierarchical VAE is to capture the importance of low level simpler features using a shallow network and subsequent high level complex features using deeper networks similar to a hierarchy. The architecture is designed to tackle two major problems associated with VAE's 1.) scaling the network while maintaining stability and 2.) Increase the receptive field / expressivity of NN.

The multi step hierarchical set up in NVAE allows it to capture the long range correlations of the data.The generative model first starts from small spatially arranged latent variables and then samples from the hierarchy with increasing spatial dimension enabling the model to capture long range as well as fine grained dependencies. The receptive field of the encoder and decoder which are implemented using deep residual networks can be improved by increasing the kernel size of the convolution layers but this comes at the cost of parameters and complexity. This can be overcome by using depth-wise convolutions but they operate independently on each channel and therefore have reduced expressivity.The authors borrow the approach from MobileNetV2[23] of first expanding the number of channels by convolution and then remapping back them to the original size. The authors use Batch Normalization with a Swish activation function as opposed to weight normalization and ELU like other VAE's. The memory requirements of the model are minimized by using an efficient mixed precision library and gradient check-pointing [2].

In order to improve the KL optimization, the authors use the below approaches

- They propose a residual distribution for which the KL term can be optimized using a single encoder network. For stabilizing the VAE , i.e no sudden changes in encoder output as input changes , they use Spectral Regularization.
- The second proposal is to use normalizing flows to generate more expressive approximate posterior.

3 Comparison

NVAE and GLOW are examined on CIFAR-10 [16], ImageNet 32x32 [4] and CelebA HQ [13].Both models follow the same test and train splits [24] for a fair comparison. NVAE clearly outperforms GLOW in performance measured in terms of bits/dimension (the lower the better) (Table 1 in [24]) on all the datasets.This metric reports the negative log likelihood (usually measured in logit space) in the image space as derived in [21]. The performance of NVAE is further improved by implementing flows especially for CelebAHQ where we see a reduction 0.33 bpd in comparison to GLOW. The results from Table 2 , Figure 4,5 in [14] show that flow based model like GLOW can not only generate high-resolution images (up to 256 x 256)but can also do an exact inference on latent variables unlike NVAE that approximates it using optimizations.

NVAE trains slower (ranging from 34imgs/sec to 64imgs/sec) when compared to StyleGANv2 which trains at 31imgs/sec with optimisations .The network also significantly suffers from instability during training. This is also highlighted by the authors in the Appendix A of [24]. The choice of hyperparameters during annealing often resulted in issues for CelebA HQ and FFHQ dataset making the model highly sensitive to parameter initialisation.

StyleGANv2 makes it easier to accredit a synthetic image to its source as a result of training with path length regularization. This improvement in performance can be seen in Figure 10 of the original paper [12] which is measured in terms of LPIPS distance between the original image and the re synthesized projection of the generated image for the LSUN car and FFHQ dataset.This encourages the application of this model on deep fake detection. The network was also shown to achieve a speedup of up to 40% with weight demodulation, lazy regularization, and code optimizations without significant reduction in performance proving the robustness of the model.

In terms of parameters efficiency,GLOW suffers the most, to generate high quality images at a higher resolutions, it uses over 200 million parameters and up to 600 convolution layers [20] which makes the training time quite large and replication of results quite infeasible.

Each model has some advantage over the other as addressed above but all of them are capable of learning the true data distribution and thus capable of generating realistic images.

References

[1] Yuemin Bian and Xiang-Qun Xie. *Generative chemistry: drug discovery with deep learning generative models*. 2020. arXiv: 2008.09000 [q-bio.BM].

[2] Tianqi Chen et al. *Training Deep Nets with Sublinear Memory Cost*. 2016. arXiv: 1604.06174 [cs.LG].

[3] Gustavo Deco and Wilfried Brauer. "Higher Order Statistical Decorrelation without Information Loss". In: *Advances in Neural Information Processing Systems 7*. Ed. by G. Tesauro, D. S. Touretzky, and T. K. Leen. MIT Press, 1995, pp. 247–254. URL: http://papers.nips.cc/paper/901-higher-order-statistical-decorrelation-without-information-loss.pdf.

[4] J. Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255.

[5] Laurent Dinh, David Krueger, and Yoshua Bengio. *NICE: Non-linear Independent Components Estimation*. 2014. arXiv: 1410.8516 [cs.LG].

[6] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. *Density estimation using Real NVP*. 2016. arXiv: 1605.08803 [cs.LG].

[7] Hao-Wen Dong et al. *MuseGAN: Multi-track Sequential Generative Adversarial Networks for Symbolic Music Generation and Accompaniment*. 2017. arXiv: 1709.06298 [eess.AS].

[8] Ian J. Goodfellow et al. "Generative Adversarial Nets". In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'14. Montreal, Canada: MIT Press, 2014, pp. 2672–2680.

[9] Raza Habib et al. *Semi-Supervised Generative Modeling for Controllable Speech Synthesis*. 2019. arXiv: 1910.01709 [cs.CL].

[10] Jonathan Hui. *GAN — StyleGAN StyleGAN2*. https://medium.com/@jonathan_hui/gan-stylegan-stylegan2-479bdf256299. 2020 (accessed September 20, 2020).

[11] Tero Karras, Samuli Laine, and Timo Aila. *A Style-Based Generator Architecture for Generative Adversarial Networks*. 2018. arXiv: 1812.04948 [cs.NE].

[12] Tero Karras et al. *Analyzing and Improving the Image Quality of StyleGAN*. 2019. arXiv: 1912.04958 [cs.CV].

[13] Tero Karras et al. *Progressive Growing of GANs for Improved Quality, Stability, and Variation*. 2017. arXiv: 1710.10196 [cs.NE].

[14] Diederik P. Kingma and Prafulla Dhariwal. *Glow: Generative Flow with Invertible 1x1 Convolutions*. 2018. arXiv: 1807.03039 [stat.ML].

[15] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2013. arXiv: 1312.6114 [stat.ML].

[16] Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images". In: *University of Toronto* (May 2012).

[17] Bowen Li et al. *Controllable Text-to-Image Generation*. 2019. arXiv: 1909.07083 [cs.CV].

[18] Yang Li et al. "A Generative Model for Category Text Generation". In: *Information Sciences* 450 (Mar. 2018). DOI: 10.1016/j.ins.2018.03.050.

[19] Thu Nguyen-Phuoc et al. *HoloGAN: Unsupervised learning of 3D representations from natural images*. 2019. arXiv: 1904.01326 [cs.CV].

[20] OpenAI. *Glow: Better Reversible Generative Models*. https://openai.com/blog/glow/. 2018 (accessed September 20, 2020).

[21] George Papamakarios, Theo Pavlakou, and Iain Murray. *Masked Autoregressive Flow for Density Estimation*. 2018. arXiv: 1705.07057 [stat.ML].

[22] Danilo Jimenez Rezende and Shakir Mohamed. *Variational Inference with Normalizing Flows*. 2016. arXiv: 1505.05770 [stat.ML].

[23] Mark Sandler et al. *MobileNetV2: Inverted Residuals and Linear Bottlenecks*. 2019. arXiv: 1801.04381 [cs.CV].

[24] Arash Vahdat and Jan Kautz. *NVAE: A Deep Hierarchical Variational Autoencoder*. 2020. arXiv: 2007.03898 [stat.ML].