Learned Data Augmentation using VQ-Vae

Arnav Dantuluri

Abstract:

In this paper, I propose a simple and easily reproducible method to enhance and extend datasets from as few as 1000 images to as much as 10000 or in essence as many as the user requires. My approach combines a proper latent space modeling of the VAE which is then modified using a process called vector quantization. With these techniques along with enhancing model parameterization and training a simple convolutional neural network can achieve accuracies of up to 93% on synthetic data which proves extremely helpful especially when handling datasets with very few images.

Introduction:

Machine learning is one of the largest and fastest-growing fields in computer science and is also one of the most data-intensive fields with industry-acceptable models requiring close to an excess of millions of images. Models like GPT-3 and BERT which are the current flagships for natural language processing were trained on hundreds of billions of words with BERT-Base taking over 4 days to train on 4 TPUs[1], keep in mind that TPUs are specifically built to train machine learning models, in comparison, it can take over 7 days to train BERT-Base on 8 gpus[2]. The main takeaway is that machine learning models require extremely large amounts of data and a long time to train. The time factor of machine learning cannot be easily overcome without improvements in hardware or cloud computing or additional computational libraries such as JAX which can significantly speed up matrix calculations and the data factor can be tackled using linear transformations such as crops, resizes, or random noise generation. While these methods are effective in their own right they do not tackle the inherent problem of a lack of data. Linear transformations are the same image just modified to a certain extent providing no real advantage to the lack of data; they just provide a solution to the problem of overfitting. To tackle the problem of data, researchers have been trying methods of learned data augmentation for years[3][4] however most of these methods utilize generative adversarial neural networks which requires a ton of computational resources or utilize a variational autoencoder which is not a effective as vq-vaes at making meaningful representations from the latent space. Visuals differences will be discussed towards the end of the paper.

Variational Autoencoders:

Variational autoencoders were first introduced in 2013[5] by Diederik P. Kingma and Max Welling. Variational Autoencoders attempt to reproduce the original X data from a Z or a latent space that exists in a lower-dimensional space than that of the original data. It is compressed by the Encoder the equation which can be written by,

$$p_\theta(x) = \int\limits_{Z} p_\theta(x|z)q(z)dz$$

$p_\theta(x|z)$ where represents the sample distribution often represented as either a Gaussian noise or a Bernoulli distribution. The decoder is another neural net. Its input is the representation z, it outputs the parameters to the probability distribution of the data, and has weights and biases ϕ. Taking for example the mnist dataset, let's say the handwritten number images are black and white and represent each pixel as 0 or 1. The probability distribution of a single-pixel can be then represented using a Bernoulli distribution. The decoder gets as input the latent representation of a digit *z* and outputs 784 (in the case of 28x28 images) Bernoulli parameters, one for each of the 784 pixels in the image. The decoder then takes in the latent space and tries to return the latent space to the original data that was passed into the encoder but because all the data cannot be contained within the reduced/compressed data the decoder often produces certain variations in the reconstructed image which in this case is extremely favorable due to the need for increased data.

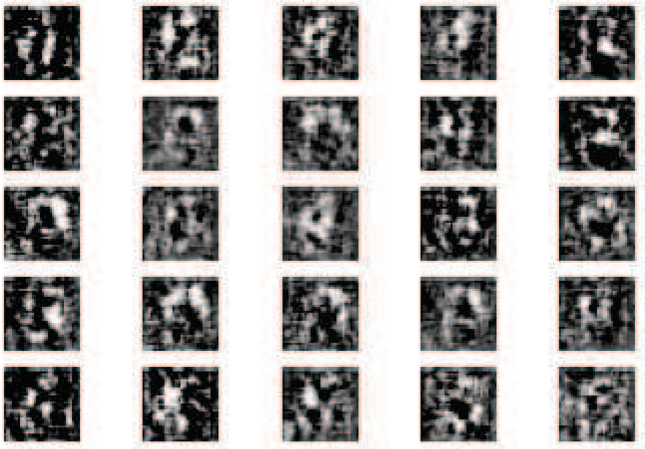
Vector Quantised Variational Autoencoders:

Vector Quantised variational autoencoders or VQ-VAE’s for short were first introduced in 2013[6] in a paper titled “Neural Discrete Representation Learning”. VQ-Vae provides multiple advantages over methods such as basic Variational Autoencoders or even GANs in the sense that the use of a VQ-Vae would prevent “posterior collapse” a phenomenon that occurs due to an extremely powerful decoder in the Vae causing certain latents to be ignored. This poses a major problem especially when dealing with data generations because the ignored latents cause data to be lost forming an incorrect image from the latent space. To explain the magnitude of this issue let’s take an example. In the case of medical imaging (CT scans, pneumonia identification, etc.) if the decoder suffers from posterior collapse it becomes useless as a means to produce meaningful representations of the data. This could pose a major issue, especially for medical data when the accuracy of the model could save a patient’s life meaning the data the model reproduces must be accurate enough to allow an identifier model, such as a convolutional neural network, to be able to replicate that accuracy. However, VQ-Vaes’ utilize a discrete latent space which leads to the issue of running backpropagation with the model not being able to run back propagations through the latent space. This issue is resolved by using a method called straight to gradient which copies the gradients over from the decoder to the encoder thereby bypassing the “codebook” vector that the VQ-Vae produces. As aforementioned, this is what produces the error with backpropagation as the latent space is non-differentiable and gradients cannot be pushed through the bottleneck which is resolved with the previously mentioned process by copying gradients over.

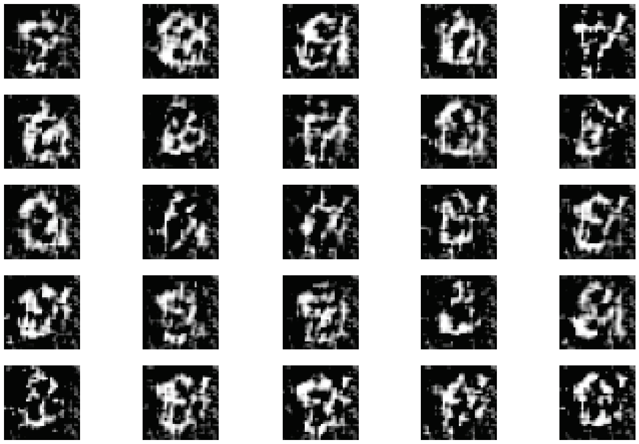
Experiments Run:

To explore how much of an advantage a VQ-VAE really has over other models such as a GAN or a VAE I trained each variation of the most popular generator models to serve as a baseline to compare to the VQ-VAE. Let's delve into the results

The results produced by a deep convolutional generative adversarial neural network (DCGAN). Utilizes Convolutional neural networks for better image classification by the discriminator.

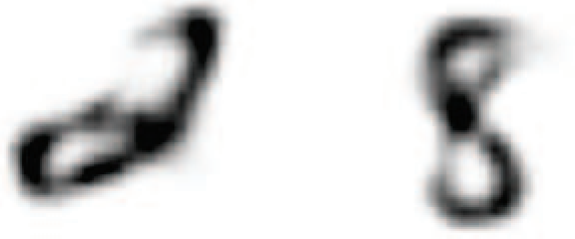


Results after 25 epochs (baseline no of epochs for each model)



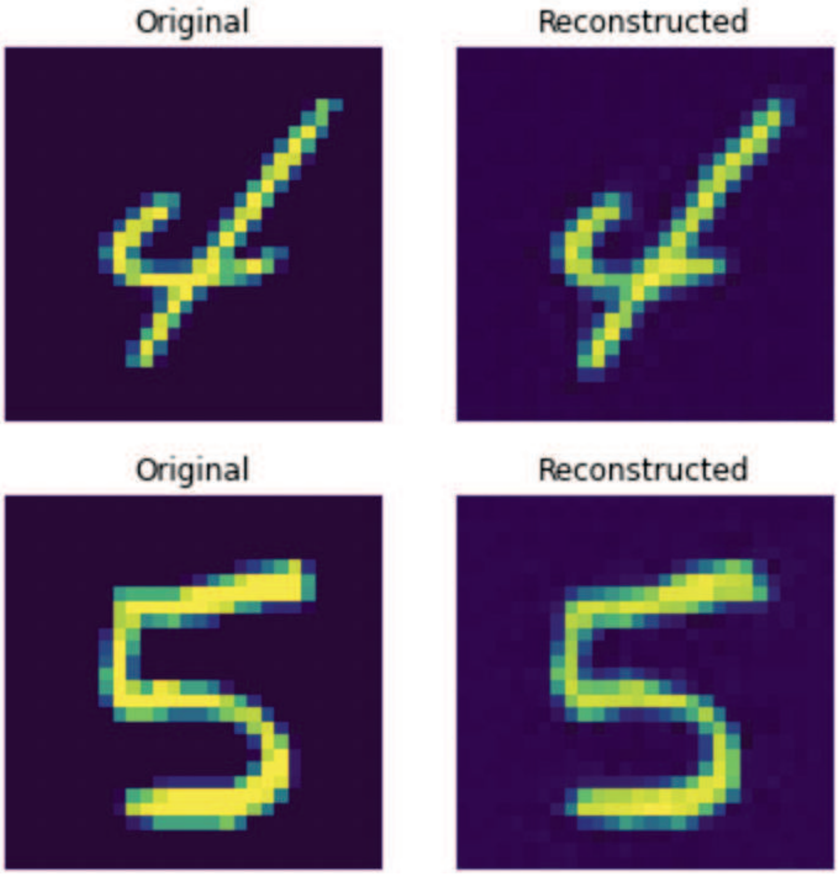
Results after 550 epochs of training.

These results truly illustrate how costly it is to implement a generative adversarial network that can produce meaningful data representations. It would take approximately 5000 epochs to produce images that even closely resemble that of the original data which is not taking into account the amount of data that we have access to.

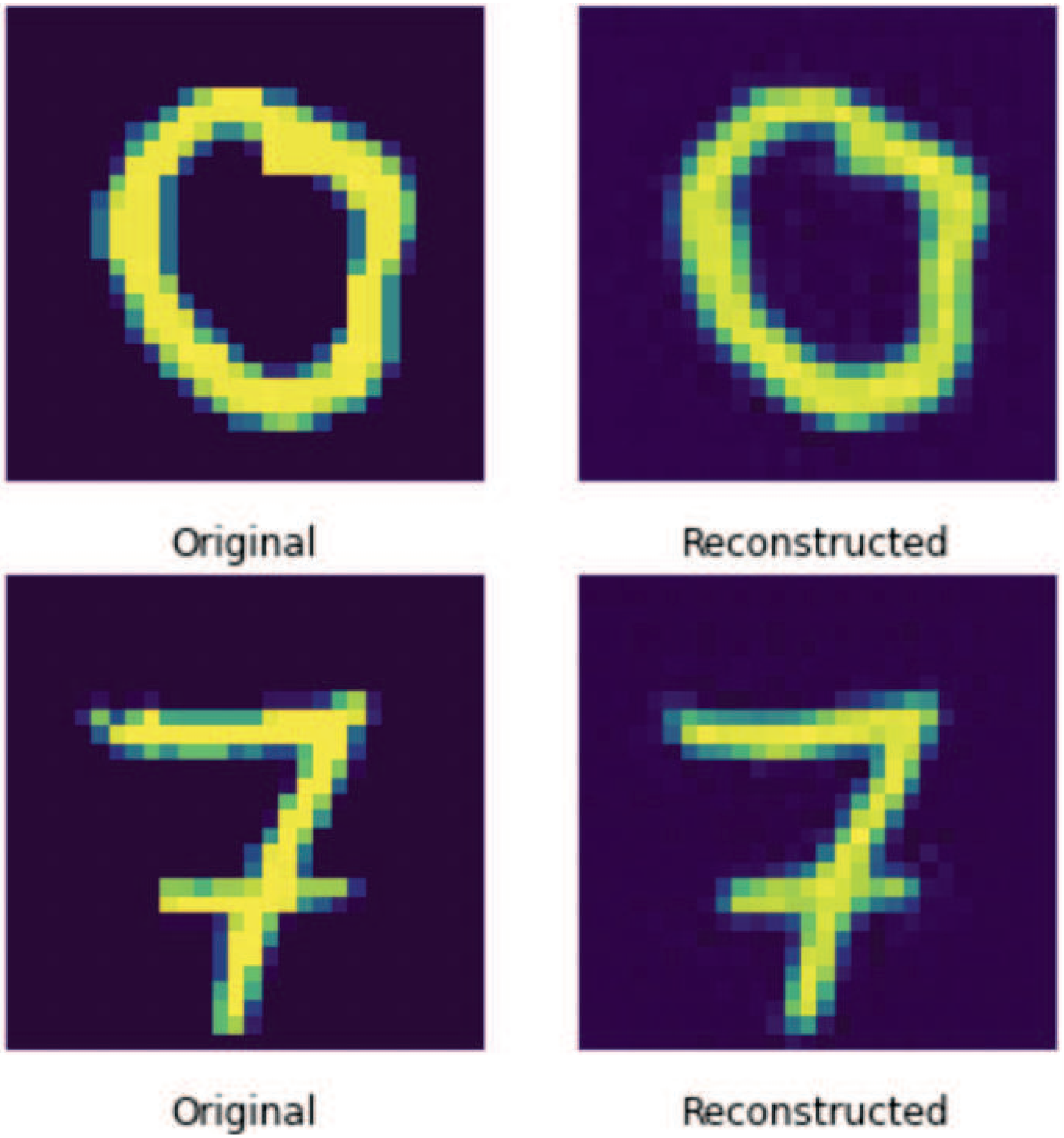


Results produced after 25 epochs of training for a variational autoencoder. Not bad but not nearly a adequate representation to deal with the data deficit.

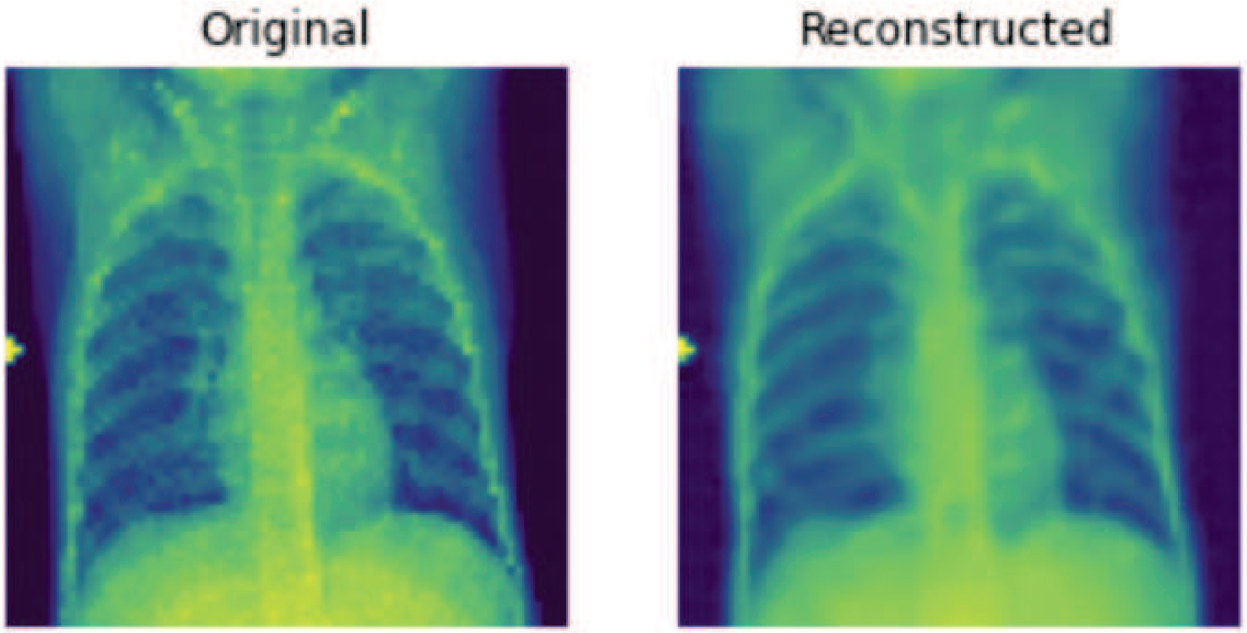
Now for the vector quantized variational autoencoder:



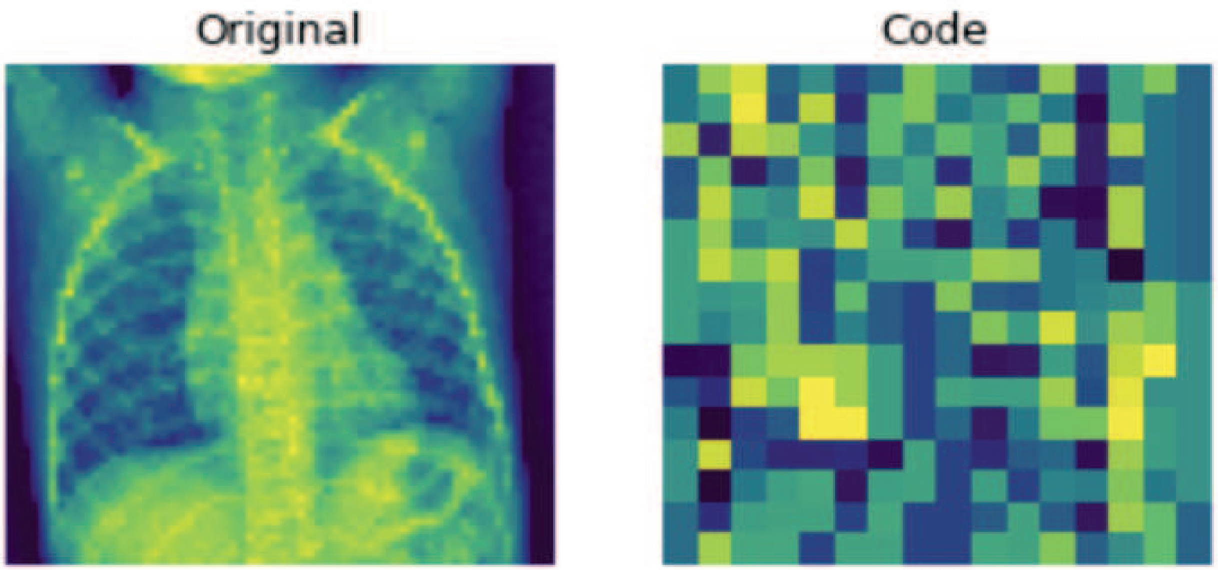
The results after 30 epochs of training. Absolutely stunning. It was able to produce a representation that almost replicated the model directly except for a little noise in the reconstructed image. Let’s take a look at another example.



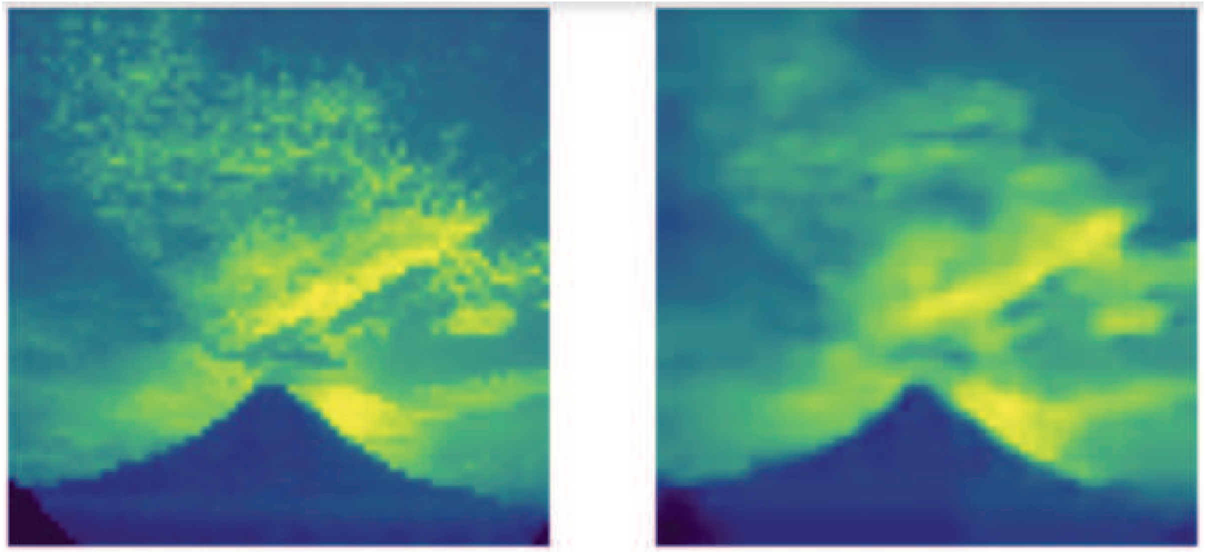
While this is truly amazing we do need to test the method further. Creating a meaningful representation from numeric values with little to no complexity cannot be compared to the complexity of an image such as a brain or a CT scan of human lungs . So let’s test the method further with the pneumonia dataset from kaggle[7].



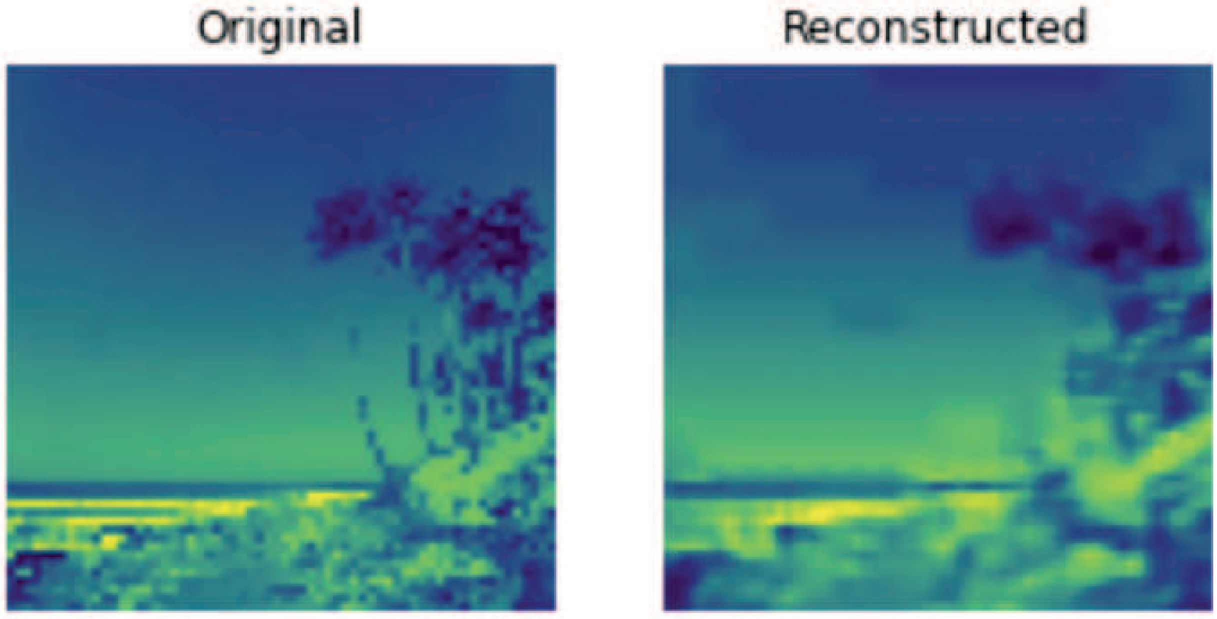
Pretty good, especially for only 500 epochs. Also important to consider the fact that the original representation also was quite noisy meaning the reconstructed image might be less accurate but the representation does contain the important features as the original image. It would be much better if we ran the model for 5000 epochs or 10000 but considering the wide array of use cases it might be better to let the model train for 430,000 updates as Dall-E was trained for including of course higher computationally effective devices such as GPUs and TPUs. The deconstructed image and the latent space showing how powerful the encoder network is after only 500 epochs.



As an experiment I wanted to utilize a dataset with a wide variety of scenes and lighting effects[8] to see if maybe the model might reconstruct an image with major differences caused because of the wide variety of images between the original images in the dataset.



The best representation after 300 epochs. Almost no differences is cloud positioning and mountain shape. Even the lighting seems to match! Slight difference in height of mountain however.



The worst representation. Image has the same general shape but is extremely noisy.

Ways to advance the model:

As always there are always multiple methods to improve the model. In this case it is possible to use PixelCNN as a encoder. PixelCNN is an extremely powerful convolutional neural network and it is possible to use the feature extraction functionality of the CNN to make a meaningful representation of the data hence making it much easier for the decoder to remap the original image. It is also possible to use some form of manifold sampling in a low sample setting size as proposed here[9].

Conclusion:

In conclusion I propose a method to extend a dataset with relatively few images 4000 or more to an extremely large amount of data that can, in theory, be extended infinitely but that introduces the problem of dilution of the image leading to excessive loss of features but at least a minimum of two times the amount of data. I also proposed ways that the reader can utilize to effectively improve upon the given baseline of VQ-VAE.

Resources:

[1] Training BERT at a University - Towards Data Sciencehttps://towardsdatascience.com › training-bert-at-a-univ…

[2] Training BERT at a University - Towards Data Sciencehttps://towardsdatascience.com › training-bert-at-a-univ…

[3] https://arxiv.org/pdf/2105.00026.pdf

[4] https://arxiv.org/pdf/2012.00848.pdf

[5] https://arxiv.org/pdf/1312.6114.pdf

[6] https://arxiv.org/pdf/1711.00937.pdf

[7] https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia

[8] https://www.kaggle.com/datasets/arnaud58/landscape-pictures

[9] https://arxiv.org/pdf/2103.13751.pdf