Analysis and Improvement of Adversarial AutoEncoder

Hyeonsik Yang
Seoul National University
Seoul, Korea
hsyang1222@snu.ac.kr

Sangyeop Kim
Seoul National University
Seoul, Korea
pdg01117@snu.ac.kr

Nuri Han
Seoul National University
Seoul, Korea
nrhan0623@snu.ac.kr

Abstract

AAE (Adversarial Auto-Encoder) is a generative model that combines VAE and GAN. GAN is difficult to learn, and GAN can fall into a mode collapse phenomenon. The images produced by VAE lack reality and have a blurry feature. AAE tries to improve its shortcomings by combining the VAE and GAN models, but there are four limitations. First, AAE assumes that the latent vector of AAE is i.i.d gaussian, and accordingly, prior is entangled. In addition, there are limitations in using pixel-wise errors and using batch norms that are reported to have an adverse effect on image generation. In this paper, we propose an IAAE model that improves these four limitations.

Our proposed technique did not improve much compared to the existing technique. We studied the reasons, and found out that there was a big problem with AAE.

1. Introduction

A generative model aims to learn the distribution of training data and generate new data following that distribution. Generative model is being studied mainly on the VAE[10]-based model and GAN[4]-based model, and AAE model combining the advantages of the previous two models is also actively studied. VAE consists of a network that encodes a data sample into a latent vector and another network that decodes a vector into a data space.

The VAE normalizes the output z of the encoder to follow the prior with the standard normal distribution, and defines the loss to the reconstruction error between the input and output. The GAN consists of a generator network that maps the latent vector into a data space and a discriminator network that distinguishes real and generated data. Generators are trained to generate more realistic data, and discriminators are competitively trained to distinguish between real and generated data.

The image generated by VAE has poor realism and blurry characteristics, and GAN is difficult to learn and has the disadvantage of mode collapse. AAE attempted to improve the shortcomings of the two models by combining VAE and GAN, but there were still various limitations. First, the latent vector was assumed to be an i.i.d. Gaussian Distribution for convenience in implementation, which led to the late vector being entangled. Second, pixel-wise error which is sensitive to translation in data is used Finally, batch norm which has been reported to adversely affect image generation in recent studies is used.

There has been research to solve some of the problems of AAE directly or indirectly, but there is no research that has solved all of them. In this study, the improved Adversarial AutoEncoder (IAAE) model is proposed by improving all of them.

2. Motivation

Deep learning models that understand and regenerate high-complexity distributions such as sounds, images, and videos have recently performed well. AAE is a model that combines VAE and GAN, which can be easily extended to semi-supervised learning scenarios and has the advantage of having competitive classification performance on various datasets. However, the AAE does not generate an image that is qualitatively equivalent to GAN. In this study, we will analyze four reasons for cause and propose an improved AAE (IAAE) that solves these problems.

First, AAE assumed that the distribution of the latent vector follows a Gaussian distribution. However, recent studies such as StyleGAN [8] have reported that there is a clear performance improvement when the latent vector is mapped to a non-Gaussian space.

Second, according to the study of Bengio *et al*. [2], there is a disentanglement as one of the characteristics that good representation should have. Gaussian distribution is a good distribution to have a disentanglement characteristic, but it is difficult to learn a latent vector that follows Gaussian distribution as a learning method of general AAE. This limitation of the AAE can be interpreted as a decrease in the quality of the image.

Third, AAE uses pixel-by-pixel loss. The pixel-by-pixel loss makes the image more plausible and allows AAE to avoid mode collapse, but causes the problem that the quality of the output image cannot be improved. This is because the error increases rapidly even with a little image translation.

Finally, AAE uses a strong normalization method such as batch normalization, and the latest research such as StyleGAN2 [9] reported that strong normalization adversely affects image generation.

In this study, we propose IAAE that solves these problems and we try to verify the validity of the IAAE.

3. Background

3.1. Adversarial Auto-Encoder(AAE)

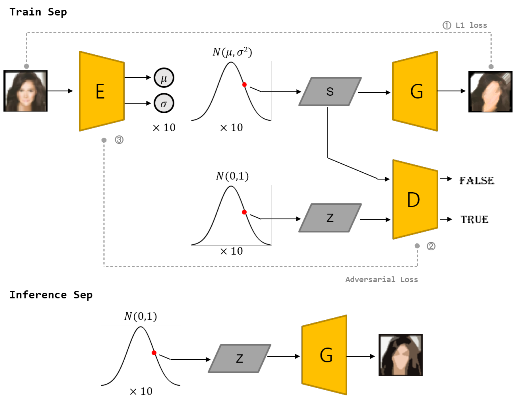

Figure 1. Train and Inference of AAE

The overall structure of the AAE is shown in Figure 1. AAE is a combined VAE and GAN model that creates z by putting the original image into Encoder as shown in the VAE method, and then reconstruct the original image in Decoder(①). In the meantime, the Discriminator distinguishes latent vector generated by Encoder from the IID Gaussian random value(②). Encoder deceives Discriminator(③) and proceeds with the learning in the direction of performing source image restoration at the same time. In inference, the IID Gaussian random feature is entered into Decoder as z to generate the image.

3.2. Entanglement and Disentanglement

In AAE, the latent vector z has different purposes in the learning phase and the inference phase. In training, information to restore the original image should be well stored, and in inference, an image with desired characteristics should be well represented. In order to satisfy the former condition, z must compress information as much as possible. However, the latent vector represented by such compressed information cannot change the output image in the desired direction by adjusting the vector value. This is because the variables are not independent. we call these vectors an entangled vector. In order to satisfy the latter condition, z must be an independent vector. If z is independent we should be able to change the desired properties of the image by adjusting one attribute of z. We call this a disentangled vector. For the disentangled representation of z, when a single latent unit changes, only a single characteristic of the generated output changes, while other characteristics remain unchanged [2]. For example, in the case of a completely disentangled latent vector z, z1 only affects the color of the person's skin, and z2 only affects the direction of the person's face. However, the amount of information contained in z, which is easy to adjust image features in this way, is relatively reduced, and the quality of the image restored by the decoder is lowered.

4. Related Work

4.1. Improvement pixel-by-pixel loss

In order to make the image generated by Adversarial Learning more realistic, there is a study by Hou *et al.* [6] that applies an additional loss term to the pixel-by-pixel loss. In Hou's study, as shown in Figure 2, a loss term was added to minimize the difference between the output features by putting the original image and the image regenerated by the generative model as inputs into the pre-trained CNN model, respectively. This allows the generated image to exhibit the special characteristics of the original image. However, this model has the disadvantage that there is no method for generating a plausible image from an image that does not exist in the original data.

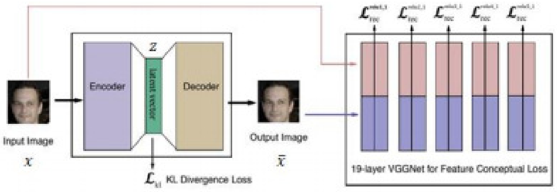

Figure 2. Learning structure in Hou's research

Larsen *et al.* [11] proposed a VAE-GAN model that properly combined the models of VAE and GAN. The VAE-GAN model allows Autoencoder to further learn the similarity of features in the data space. Figure 3 is the structure of the VAE-GAN model. By combining VAE and GAN, this model can use the learned feature representation in the GAN's discriminator. Disl(x) is the hidden of the lth layer of the discriminator representation. The pixel-wise error is replaced by the error between the two features (output from Disl(x)) when the actual x is inserted and the fake x is inserted. Therefore, they can learn better data distributions and furthermore make them invariant to transformations such as translation. do. VAE-GAN has the advantages of using feature-wise errors, but because the model is large, it requires a lot of time and resources to learn.Testing with

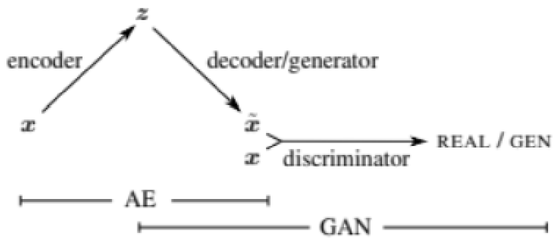

Figure 3. Structure of VAE-GAN and example of CelebA output

CelebA data showed that the generated image quality of the AAE model was better for the initial epoch. Figure 4 shows the generated images of each model of AAE (left) and VAE-GAN (Right) in epoch 50. It can be seen that the image generated from AAE is clearer than that of VAE-GAN.

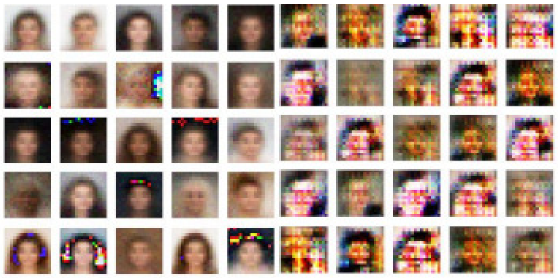

Figure 4. Comparison of generated images from AAE and VAE-GAN at epoch 50

4.2. Beta-VAE

The learning methods of AAE and VAE have many things in common. Therefore, in order to have disentanglement characteristics in AAE, the study of Higgins et al. [3], a research paper for obtaining disentangle latent vectors in VAE, can be used.

The general VAE loss function is as follows.

$$L = -E_{z \sim q(z|x)}[\log p(x \mid z)] + D_{KL}(q(z \mid x)||p(z))$$

In beta-VAE, the loss function of VAE is transformed into solving the following optimization problems.

$$\begin{aligned}&\max \phi, \theta \mathbb{E}\mathbf{x} \sim D\left[\mathbb{E}\mathbf{z} \sim q\phi(\mathbf{z} \mid \mathbf{x}) \log p_\theta(\mathbf{x} \mid \mathbf{z})\right]\\ &\text{subject to } D_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \| p_\theta(\mathbf{z})\right) < \delta\end{aligned}$$

The above optimization equation maximizes the reconstruction performance and limits the difference between the distribution of the prior and the aggregated posterior within delta. If the KKT condition is applied to solve the equation (lagrangian multiplier beta), the following equation can be obtained, which is the lower limit of the optimization equation.

$$\begin{aligned}\mathcal{F}(\theta, \phi, \beta) &= \mathbb{E}\mathbf{z} \sim q\phi(\mathbf{z} \mid \mathbf{x}) \log p_\theta(\mathbf{x} \mid \mathbf{z}) - \beta\left(D_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \| p_\theta(\mathbf{z})\right) - \delta\right)\\ &= \mathbb{E}\mathbf{z} \sim q\phi(\mathbf{z} \mid \mathbf{x}) \log p_\theta(\mathbf{x} \mid \mathbf{z}) - \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \| p_\theta(\mathbf{z})\right) + \beta\delta\\ &\geq \mathbb{E}\mathbf{z} \sim q\phi(\mathbf{z} \mid \mathbf{x}) \log p_\theta(\mathbf{x} \mid \mathbf{z}) - \beta D_{\mathrm{KL}}\left(q_\phi(\mathbf{z} \mid \mathbf{x}) \| p_\theta(\mathbf{z})\right)\end{aligned}$$

Here, the larger the value of beta, the more likely the posterior will fit prior ∼ N(0, I), which makes the latent vector disentangle.

5. Proposed Method

5.1. Vector Mapping

In our proposed work, we attempt to ensure that the latent vector has disentangled properties and at the same time that the quality of the generated image is not qualitatively inferior to that of the input of the entangled z. As shown in Figure 5, instead of using Z extracted from the Gaussian distribution as input, the value after passing through the mapping network will be used as the input value of the generative model. In StyleGAN, 8 Fully Connected Layer were used, but we try to find an optimized mapping network through experiments with various layers.

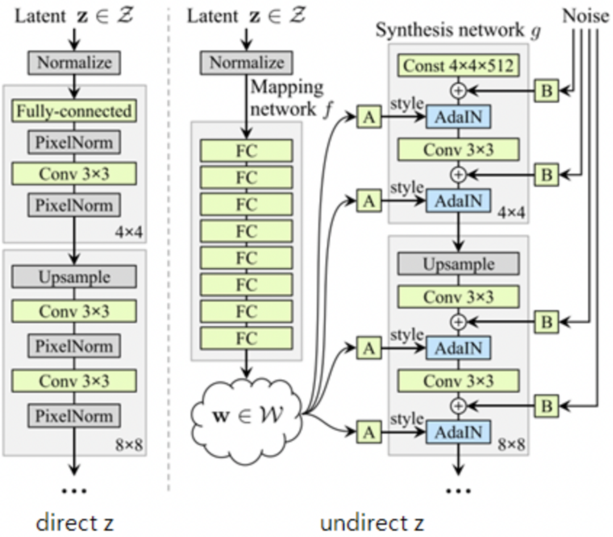

Figure 5. z mapping of StyleGAN

5.2. Disentanglement

We try to apply the idea of beta-VAE by transforming it to fit AAE. beta-VAE makes the latent vector more disentangle by adding a constraint on the latent vector to the loss function. The method of constraining the loss of AAE can be implemented by adding regularization such as KLD to the discriminator loss during training. The discriminator loss of the original AAE is as follows.

$$-\mathrm{E}_{\mathbf{z}\sim N(0,I)}[\log D(\mathbf{z})] + \mathrm{E}_{\mathbf{x}\sim p_{\text{data}}}[\log(1-D(q(\mathbf{z}|\mathbf{x}))]$$

Here, the following discriminator loss is used for training by adding the KLD loss of prior z and posterior $q(z|x)$.

$$-\mathrm{E}_{\mathbf{z}\sim N(0,I)}[\log D(\mathbf{z})]+\mathrm{E}_{\mathbf{x}\sim p_{\text{data}}}[\log(1-D(q(\mathbf{z}|\mathbf{x}))]+KLD(z|q(z|x))$$

In another way, the beta-VAE beta penalty can be applied by repeating the training corresponding to regularization several times. Since the part corresponding to regularization in AAE is the discriminator loss, the update of the discriminator is repeated n times.

5.3. Techniques using feature-wise error

AAE improved the KL Divergence term of VAE, but still It uses a pixel-wise error between input and output, which allows for fast convergence but produces blurry images. To improve this, we try to improve by mixing the pixel-wise error of the reconstruction error with the feature-wise error. In this paper, based on the idea that latent vector learns the hidden representation of the input, the distance between the latent vector of the real image and the generated images is used. As a result, we try to obtain an invariant effect on image translation (Figure 6).

5.4. Image improvement with weak normalized

The original AAE used a very strongly regularized input inter-neural net by batchnorm [7]. However, there exists a state-of-the-art study [8] shows that strong normalization make the quality of the image messy. Therefore, instead of batchnorm, we choose to use an equal layer that uses only very weak normalized ones.

$$\gamma = 1/\sqrt{D_{in}}$$
$$\hat{A} = \gamma A,\ \hat{b} = \gamma b$$
$$EqualLinear(x) = x\hat{A}^T + \hat{b}$$

Instead of normalizing the feature of input x, equal linear scales the weights and biases of the linear layer. Scaling factor gamma is the square root of dim in the input dimension.

6. Experiments

6.1. Dataset and Evaluation Metric

To evaluate the proposed scheme, various experiments are performed in FFHQ [8] and CIFAR10, MNIST, EMNIST, and Fashion-MNIST datasets. The experiment was performed from 32x32 images to 64x64 images. The performance evaluation was performed using the most widely used Inception Score [1] and Frechet Inception distance[3] when evaluating the image quality of the image generation model. Inception Score(IS) has higher values as the quality of the image is better and the diversity of the image is greater. The Frechet acceptance distance (FID) calculates the Frechet distance using the feature of the intermediate layer of the Inception Network. We will use FID as a slightly more important metric because FID captures the disturbance level very well by monotonically increasing whereas inception score fluctuates, stays flats or even, in the worst case, increases [5]. In the case of IS and FID, the values change depending on the batch size, so a batch size of 1024 for 32 pixels and 256 batch sizes for 64 pixels was used.

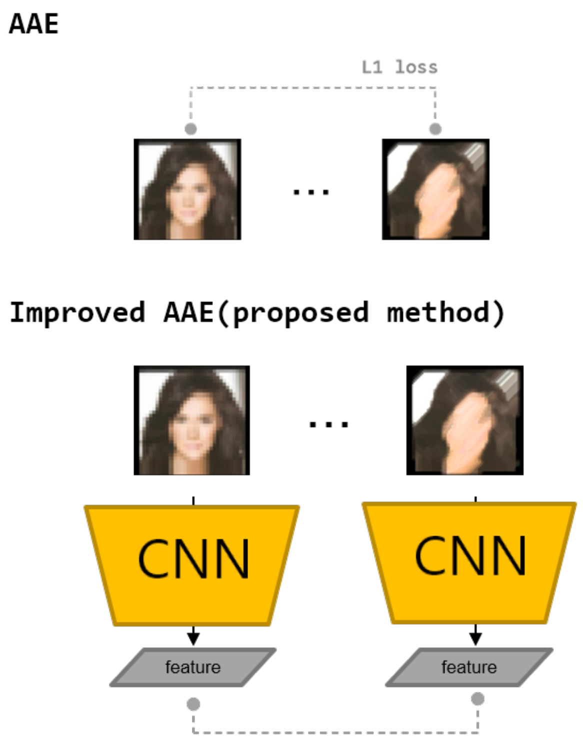

Figure 6. feature-IAAE idea

6.2. Proposed Method Result

Figure 7 shows that images generated by AAE and our models(latent-mapping-IAAE, n-iter-IAAE, KLD-loss-IAAE, equal-linear-IAAE) for several data sets: CIFAR-10, FFHQ32, FFHQ64, MNIST, EMNIST and Fashion MNIST. Since the resolution of the experimental data was not increased, there was no significant difference in quality with the naked eye. However, since our models confirmed that the image quality index, FID, is improved for a specific data set, I think that better quality images can be obtained by experimenting with high resolution.

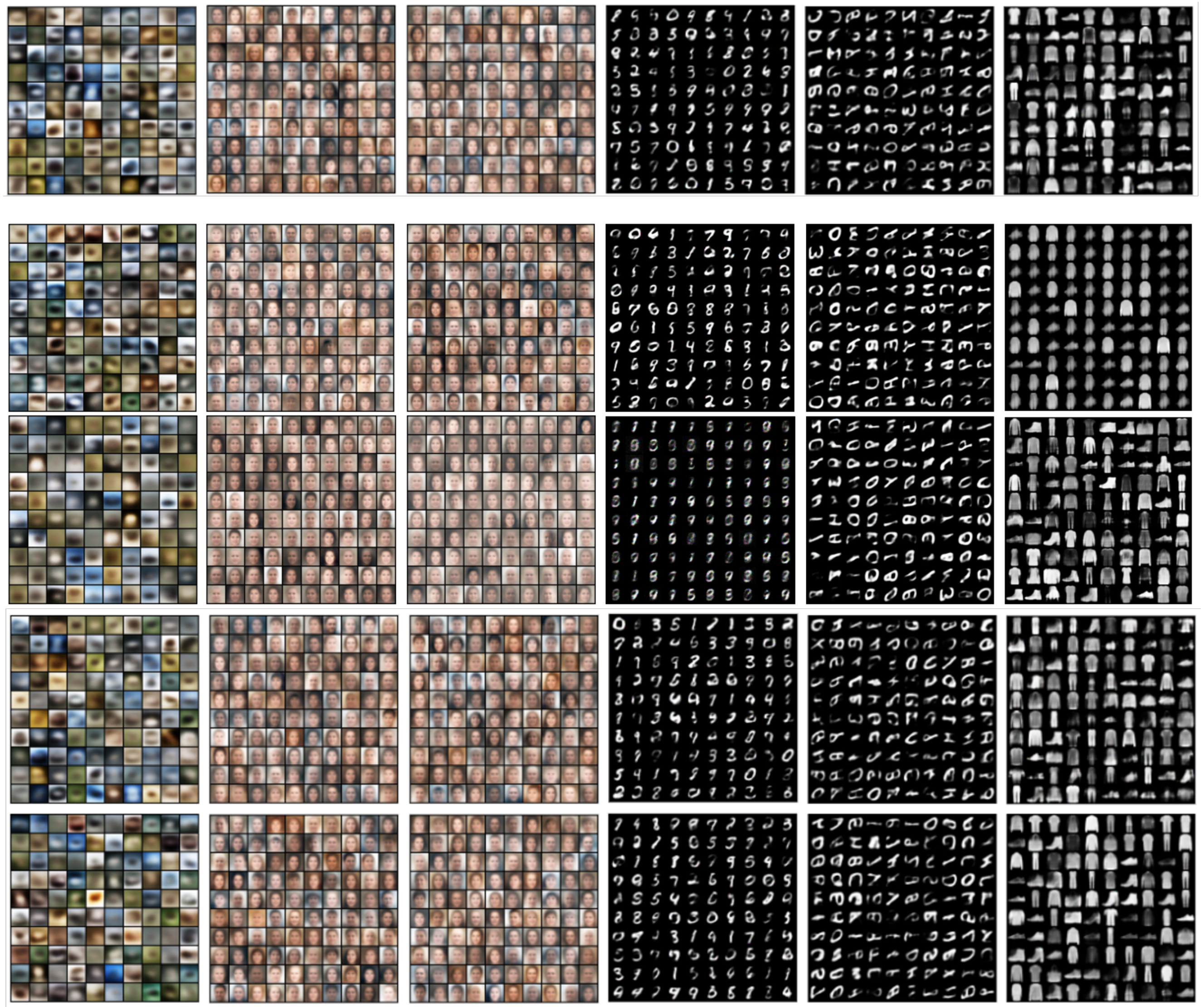

Figure 7. Generated images of models. From top to bottom, generation by AAE, generation by latent-mapping-IAAE, generation by n-iter-IAAE, generation by KLD-loss-IAAE, generation by equal-linear-IAAE. From left to right, CIFAR-10, FFHQ32, FFHQ64, MNIST, EMNIST, fashion MNIST.

From table 1 to 5 show the FID and inception score for each model in each data set. The experiment was performed at 32 pixel in all datasets, FFHQ dataset was additionally tested with 64 pixel. This is because the FFHQ dataset is the most standard image set for image generation and is the easiest to compare the results of the generated images.

In the CIFAR-10 dataset, KLD-loss-IAAE shows performance improvement among disentanglement methods(see Table 1), but it is difficult to say that it improves significantly. This can also be confirmed through the generated image. Since CIFAR-10 has a lot of labels, images produced by most AAE architecture seems to be of poor quality.

Method	FID	IS
AAE	334.017	1.701
latent-mapping-IAAE	332.404	1.646
n-iter-IAAE	334.712	1.689
KLD-loss-IAAE	***331.479***	1.662
feature-IAAE	336.938	***1.806***
equal-linear-IAAE	344.172	1.733

Table 1. FID and inception score overview for CIFAR-10 data set. (bold : best performance, underline : baseline)

Table 2 represents the result of experiments with FFHQ dataset. With the 32-pixel FFHQ dataset, the proposed model shows no performance improvement. At 64 pixels,

Method	pixel	FID	IS
AAE	32	***228.989***	2.156
n-iter-IAAE	32	236.342	2.131
latent-mapping-IAAE	32	233.336	2.179
KLD-loss-IAAE	32	229.355	2.122
feature-IAAE	32	276.216	***2.203***
equal-linear-IAAE	32	256.897	2.168
AAE	64	185.864	1.528
latent-mapping-IAAE	64	183.698	1.558
n-iter-IAAE	64	***182.961***	1.536
KLD-loss-IAAE	64	185.005	***1.570***
feature-IAAE	64	188.746	1.508
equal-linear-IAAE	64	186.876	1.527

Table 2. FID and inception score overview for FFHQ data set. (bold : best performance, underline : baseline)

Method	FID	IS
AAE	74.041	2.52
latent-mapping-IAAE	70.451	***2.597***
n-iter-IAAE	***68.92***	2.537
KLD-loss-IAAE	71.304	2.509
feature-IAAE	80.557	2.475
equal-linear-IAAE	76.870	2.468

Table 3. FID and inception score overview for MNIST data set. (bold : best performance, underline : baseline)

the disentanglement models n-iter-IAAE and KLD-loss-IAAE show slight performance improvement in FID and IS, respectively. The generated images of n-iter-IAAE and AAE with performance improvement in FID can be seen in Figure 8.

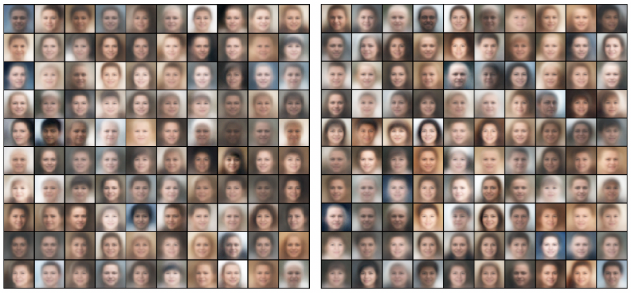

Figure 8. The generated image having 64 x 64 pixel (left : AAE, right : n-iter-IAAE)

From Table 3 to Table 5, there are results showing significant performance improvement of the proposed IAAE techniques. In particular, n-iter-IAAE shows great performance improvement in MNIST and Fashion MNIST, and latent-mapping-IAAE shows performance improvement in EMNIST. In conclusion, there was no IAAE method that show robust performance improvement for all datasets, but there are some methods that show sufficient performance improvement for some datasets.

Method	FID	IS
AAE	76.454	2.385
latent-mapping-IAAE	***72.478***	2.361
n-iter-IAAE	75.539	2.377
KLD-loss-IAAE	74.594	2.374
feature-IAAE	84.102	***2.414***
equal-linear-IAAE	79.478	2.372

Table 4. FID and inception score overview for EMNIST data set. (bold : best performance, underline : baseline)

Method	FID	IS
AAE	131.259	***3.257***
latent-mapping-IAAE	126.278	3.173
n-iter-IAAE	***117.331***	3.082
KLD-loss-IAAE	134.738	3.253
feature-IAAE	149.097	3.231
equal-linear-IAAE	136.425	3.241

Table 5. FID and inception score overview for Fashion MNIST data set. (bold : best performance, underline : baseline)

6.3. Analysis

As we employ ideas from the latest SoTA techniques of generative model, we expect that the proposed techniques will clearly improve performance. However, the performance improvements in each technique did not exist as expected, and we investigate its causes closely.

Figure 9 shows that there are several common features in various FID performance graphs, with rapid performance improvements from the beginning of learning to around 20 epochs. However, after that, the decrease in FID values is rapidly reduced or converged, and sometimes it is increasing.

This phenomenon can be explained through the distribution of posterior that we estimate as learning progresses. In the assumption of AAE, the prior assumes a normal distribution with an average of 0 and a standard deviation of 1. Therefore, if we learn the posterior well, the standard deviation should have a value of 1. However, in the Figure 10, the standard deviation of AAE is actually converged to zero. This results in the creation of a deteministic latent vector because it gives little variation of the latent vector that we produce. That is, AAE is no longer a generative model, but only plays the role of embedding into a low-dimensional vector like an autoencoder. In future studies, we need to verify exactly what negative effects the standard deviation convergence of AAE has on the proposed methodologies and continue to study ideas that can prevent this phenomenon.

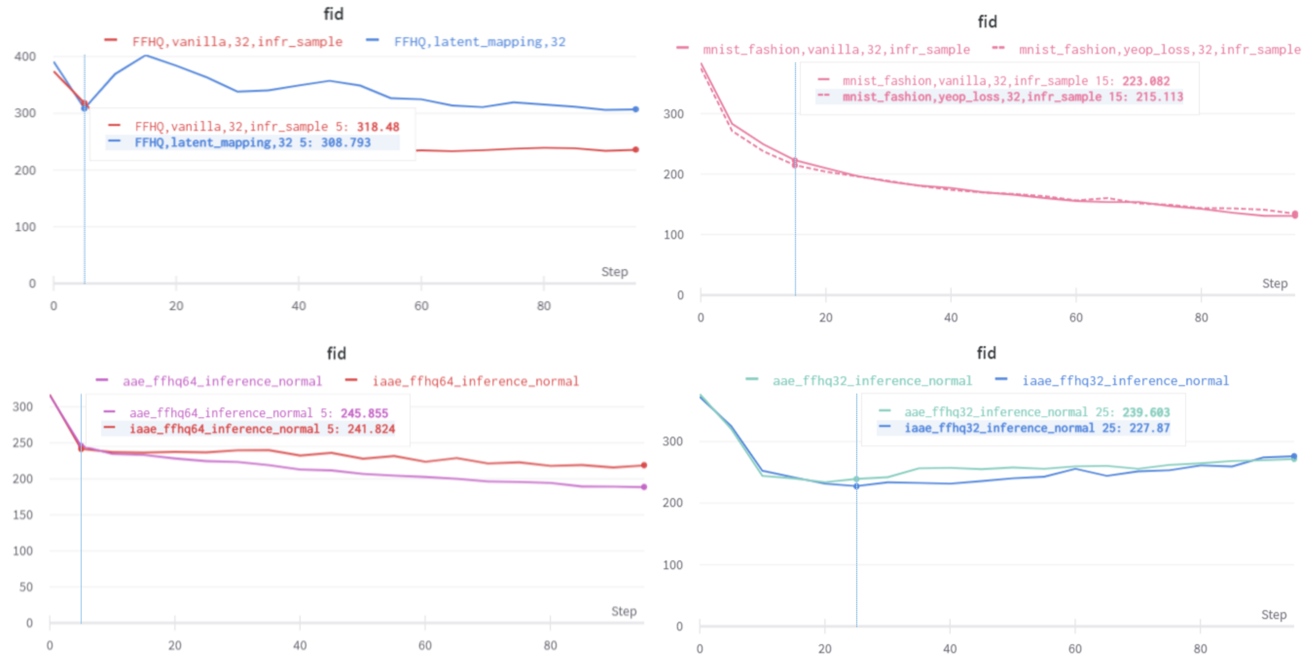

Figure 9. Example of FID Value Change according to Learning

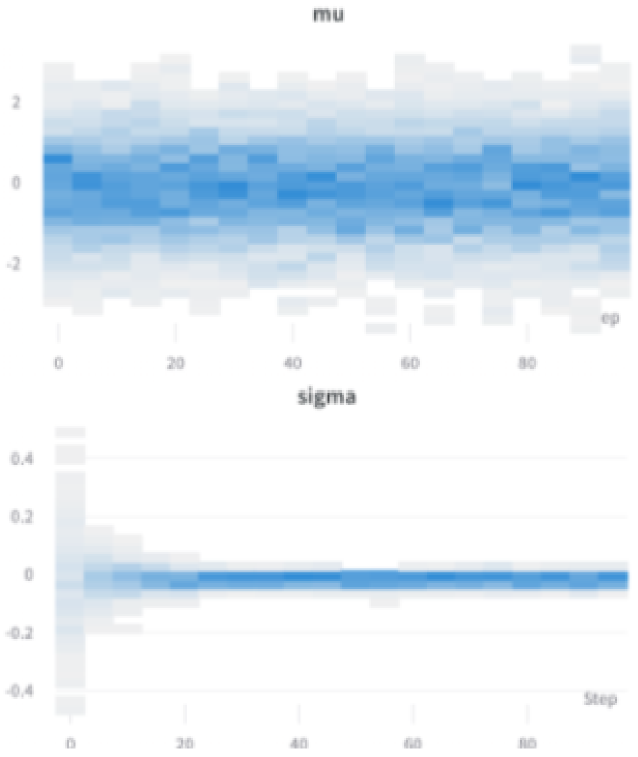

Figure 10. Changes in mu and sigma distributions with learning (mu on the top, sigma on the bottom)

We originally tried to compare performance by creating a final model combining each technique. However, since AAE does not work as a generative model, it is meaningless to create a final model and evaluate its performance with a generative model.

7. Conclusion

In this study, we try to summarize the problems of AAE into four categories. First, AAE assumes that the latent vector is Gaussian. Second, AAE was assumed that posterior would change easily. Third, AAE uses pixel-by-pixel loss. Finally, AAE uses powerful normalize methods such as batchnorm.

The latent-mapping-IAAE, KLD-loss-IAAE, and n-iter-IAAE methods are proposed as methods to solve the problem of the strong assumption that the prior follows a Gaussian distribution and iid. Latent-mapping-IAAE solves the problem of distribution assumptions by mapping to a non-Gaussian distribution. On the other hand, KLD-loss-IAAE and n-iter-IAAE solve the problem of the above assumption by finding a way to make them learn the prior better.

The problem that AAE is sensitive to image translation due to pixel-wise error is solved through feature-IAAE. To improve this, we propose feature-IAAE method by mixing the pixel-wise error of the reconstruction error with the feature-wise error.

The last normalize problem is solved through equal-linear-IAAE using an equal linear layer that weakens the strength of normalization.

As a result of verifying the effectiveness of the proposed models on five datasets, some datasets show performance improvement, but we cannot find a methodology that shows performance improvement in all datasets. However, we can clearly identify the limitations of AAE, and the study is meaningful in that it suggests various methodologies for performance improvement in specific data sets. In addition, as a result of analyzing the performance improvement, it was possible to newly discover the problem of AAE that the standard deviation converges to 0 as learning progresses. Therefore, it is expected that more meaningful results can be found if we conduct a study to solve the problem of convergence of standard deviations in future studies and verify the effect of IAAE once again. Starting with testing a model that combines the ideas of KLD-loss-IAAE and feature-IAAE, whose performance has improved compared to AAE, We plan to apply the ideas to overcome the limitations of AAE directly to many types of data and combine them into a model based on the results obtained.

8. Contribution and Limiation

We found a key problem with learning in AAE. We points out the implicit assumptions of existing AAEs and proposes ways to improve them.

However, since we have not solved the core problem of AAE, we do not fully validate the performance of the proposed technique.

9. Future Work

We will address the core problem of AAE, which fails to proceed with learning with a generative model. It then plans to reapply the four techniques proposed in this study and measure the degree of improvement.

References

[1] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018. 4

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013. 1, 2

[3] DC Dowson and BV Landau. The fréchet distance between multivariate normal distributions. *Journal of multivariate analysis*, 12(3):450–455, 1982. 4

[4] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014. 1

[5] 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. *arXiv preprint arXiv:1706.08500*, 2017. 4

[6] Xianxu Hou, Linlin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1133–1141. IEEE, 2017. 2

[7] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. 4

[8] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 1, 4

[9] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020. 2

[10] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1

[11] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning*, pages 1558–1566. PMLR, 2016. 2