Course Name : Deep Learning

Project Name : Anomaly Detection on FashionMNIST

Submitted By:

Name: Sameer Ahmed

Matricola: 2047250

Email : ahmed.2047250@studenti.uniroma1.it

Submitted To:

Professor: Fabrizio Silvestri

Introduction:

Anomaly detection in images refers to the process of identifying unusual or anomalous patterns within a collection of images. It involves detecting images that deviate significantly from the expected or normal visual characteristics exhibited by the majority of the images in a dataset. In anomaly detection for images, the goal is to distinguish abnormal or anomalous images from the normal ones. Anomalies in images can take various forms, such as objects or regions that are rare, unexpected, or inconsistent with the overall patterns in the dataset. Detecting these anomalies can have numerous applications, including quality control in manufacturing, surveillance systems, medical imaging, and anomaly detection in satellite imagery.

Task Description:

Our task is to find out anomaly detection in FashionMNIST dataset. Anomaly detection on FashionMNIST refers to the process of identifying unusual or anomalous patterns in the FashionMNIST dataset. FashionMNIST is a popular benchmark dataset in the field of computer vision, consisting of 60,000 labeled images of 10 different fashion categories, such as T-shirts, dresses, shoes, and bags. Each image is a 28x28 grayscale picture.

Anomaly detection aims to find instances that deviate significantly from the normal patterns present in the dataset. In the context of FashionMNIST, anomalies can be images that do not belong to any of the predefined fashion categories or images that exhibit unusual or unexpected characteristics compared to the majority of the dataset.

What is our approach:

Here to find out anomalies in FashionMNIST dataset, I am considering one FashionMNIST class (label) as normal and rest of nine classes as anomaly.

1. Normal
    a. Class 0 (T-shirt/top)
2. Anomaly
    a. Class 1 (Trouser)

b. Class 2 (Pullover)
c. Class 3 (Dress)
d. Class 4 (Coat)
e. Class 5 (Sandal)
f. Class 6 (Shirt)
g. Class 7 (Sneaker)
h. Class 8 (Bag)
i. Class 9 (Ankle boot)

Here, I am taken two approaches,

One is the baseline approach, where the model is based on Normal GAN, and Generator is based on autoencoder. On the other hand, The second approach is based on a state-of-the-art (SOTA) model, where the model is based on DCGAN, and Generator is based on autoencoder. This time we try to use different loss functions according to the research paper.
https://paperswithcode.com/paper/gan-based-anomaly-detection-in-imbalance

Metrics:

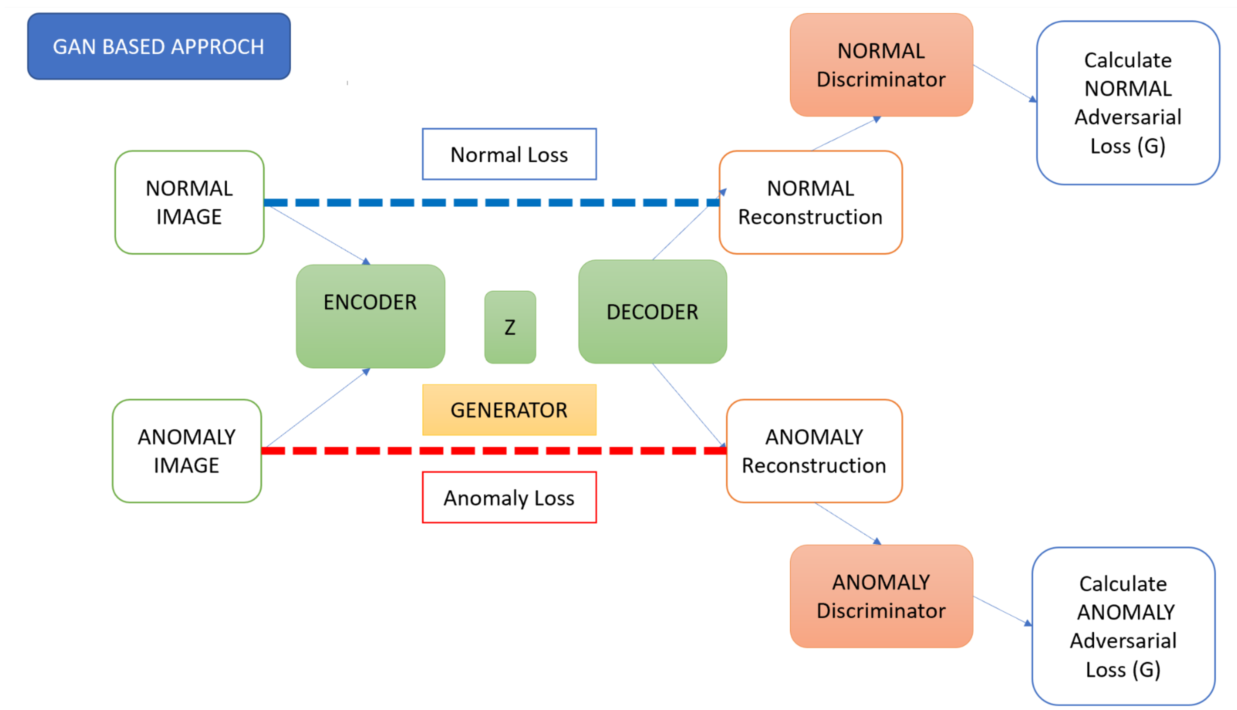
Here the metrics is **"AUROC"** for anomaly detection, Because anomaly detection datasets often have a significant class imbalance, with normal instances outnumbering anomalies. AUROC is robust to imbalanced datasets and provides a comprehensive evaluation of the model's performance in distinguishing between the two classes.

Baseline Model:

The first model is the baseline approach, where the model is based on traditional GAN, and Generator is based on autoencoder. **"I am taken the same**

methodology given in the SOTA model research paper". But here the approach is traditional GAN instead of DCGAN. And here I am using a single loss function, Mean Squared Error (MSE) Loss. No multiple losses mention in the SOTA model paper. The traditional GAN approach and the use of MSE loss provide a straightforward and easy-to-implement solution. This approach may be suitable for initial exploratory research or as a baseline to compare against more complex models.

Model Architecture:

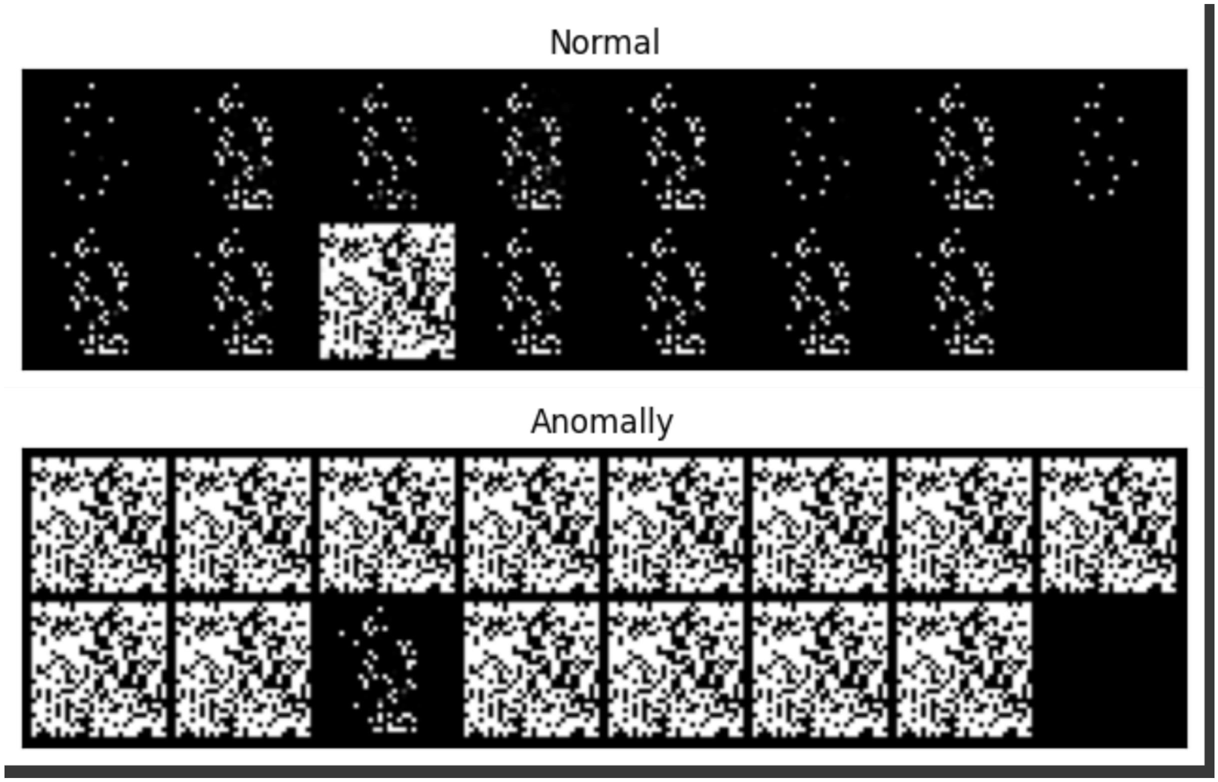


Hyperparameter of the Model:

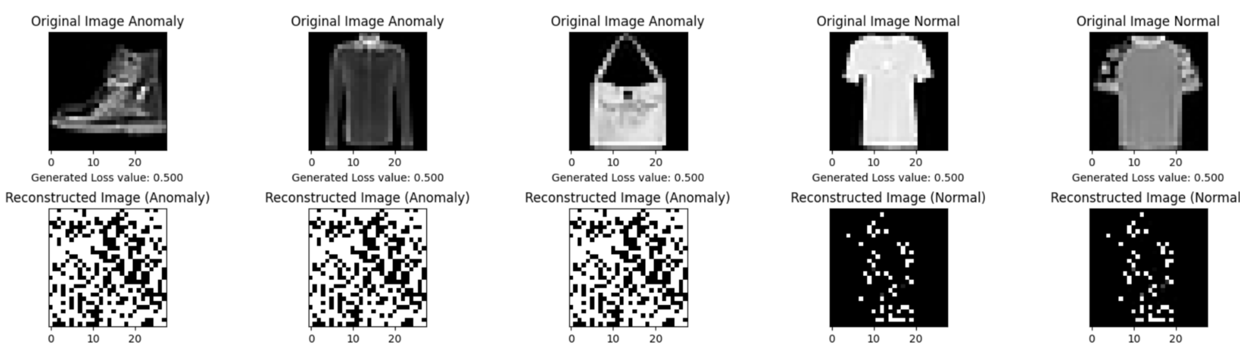
- Hidden layer sizes (Generator): 256, 128, and 64
- Hidden layer sizes (Discriminator): 256, 128, and 1
- Activation Function: ReLU, and Sigmoid (For binary classification - Normal or Anomaly)
- Optimizer of Generator : Adam (learning rate =1e-3, weight_decay=1e-5)

- Optimizer of Discriminator 1: Adam (learning rate =1e-3, weight_decay=1e-5)
- Optimizer of Discriminator 2: Adam (learning rate =1e-3, weight_decay=1e-5)
- Loss Function: Mean Square Error (MSE)

Model Results (Training Data):



Model Results (Testing Data):



'The model have some noise, It's not generating the accurate image of Normal'

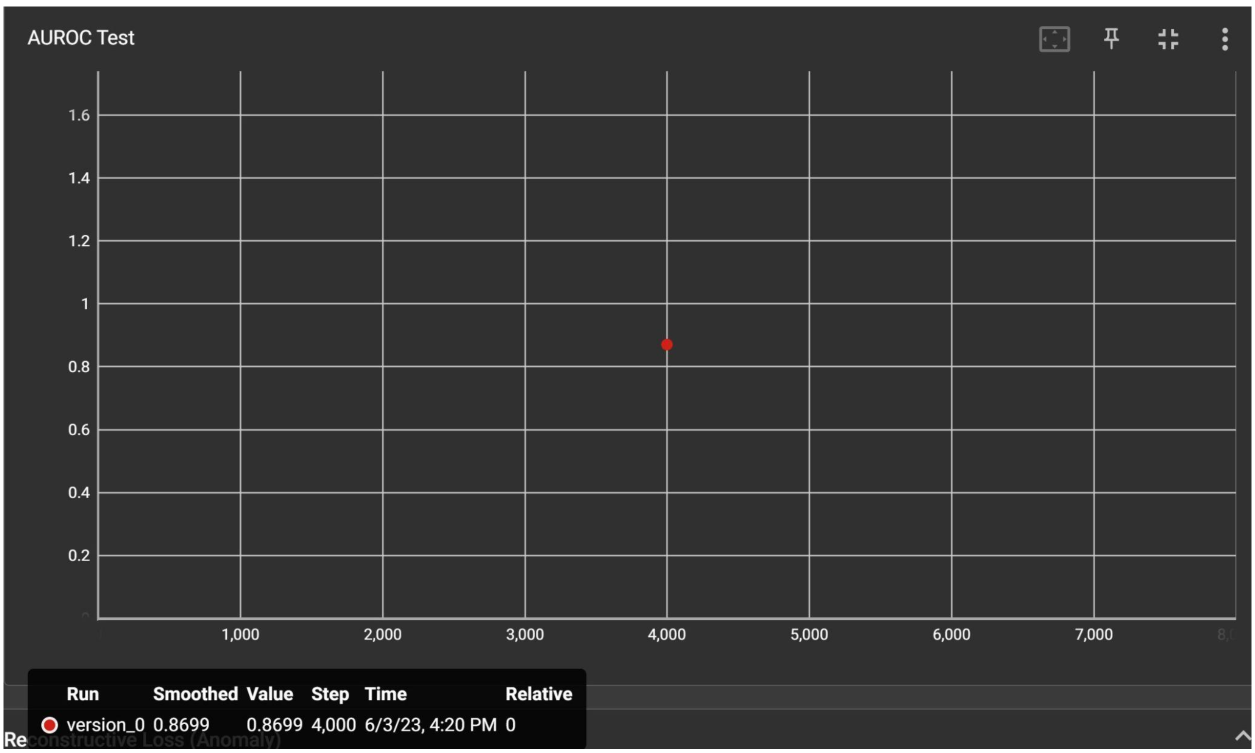
Model AUROC Result:

Test metric	DataLoader 0
AUROC Test	0.8698956370353699

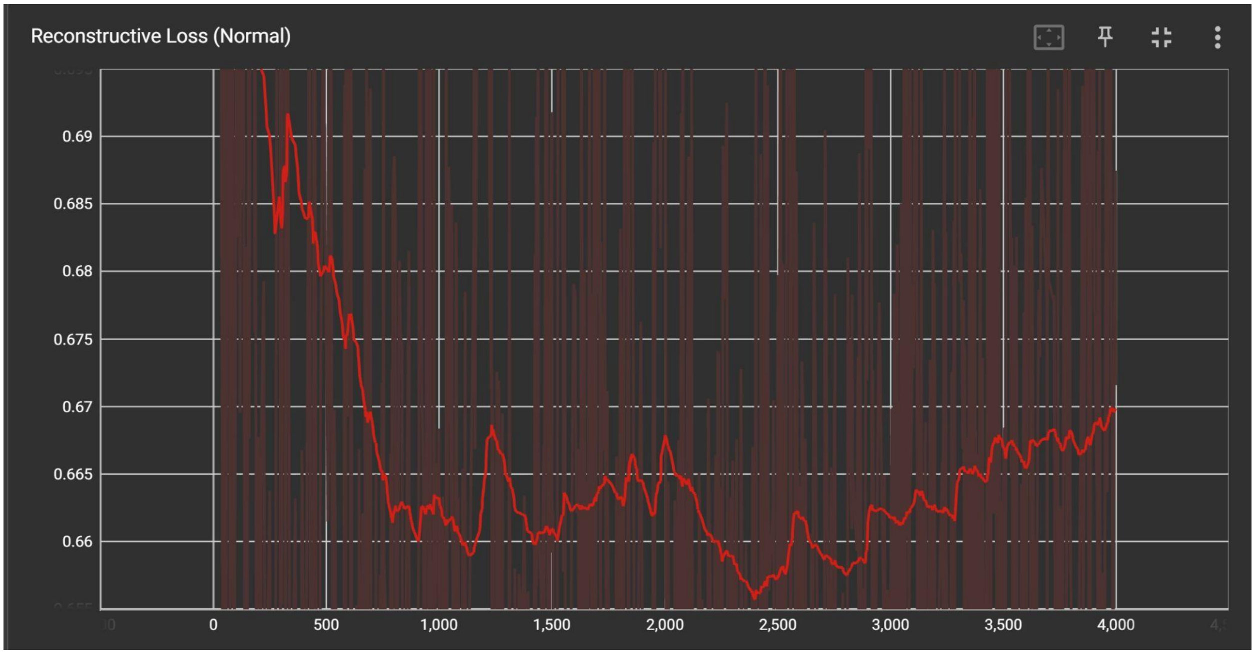
AUROC result = Approx 87%

Tenorboard Results:

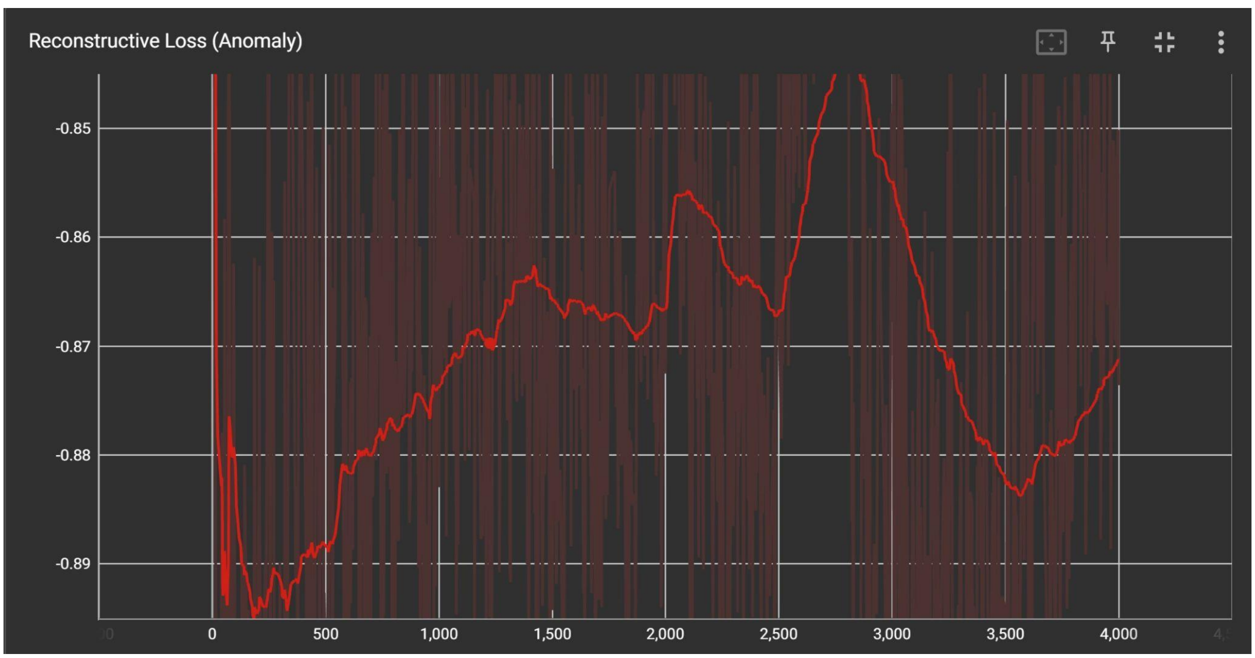
AUROC Test:



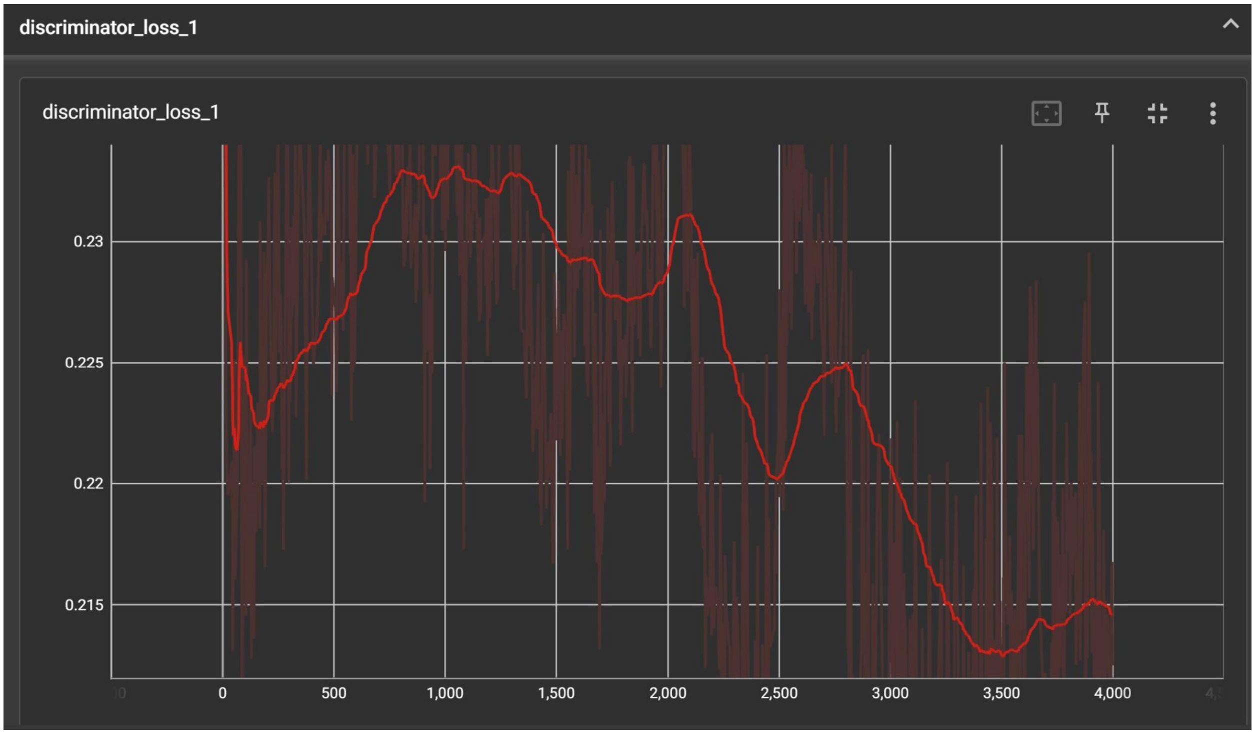
Generative Loss (Normal)



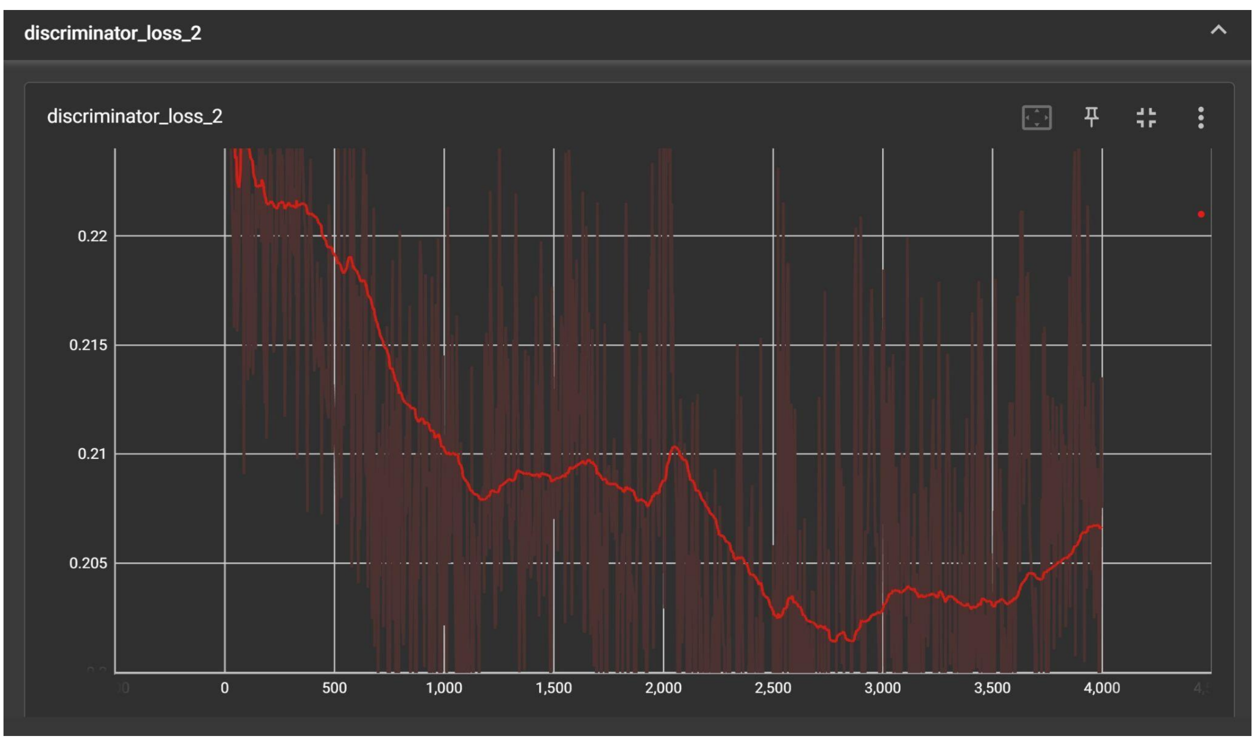
Generative Loss (Anomaly)



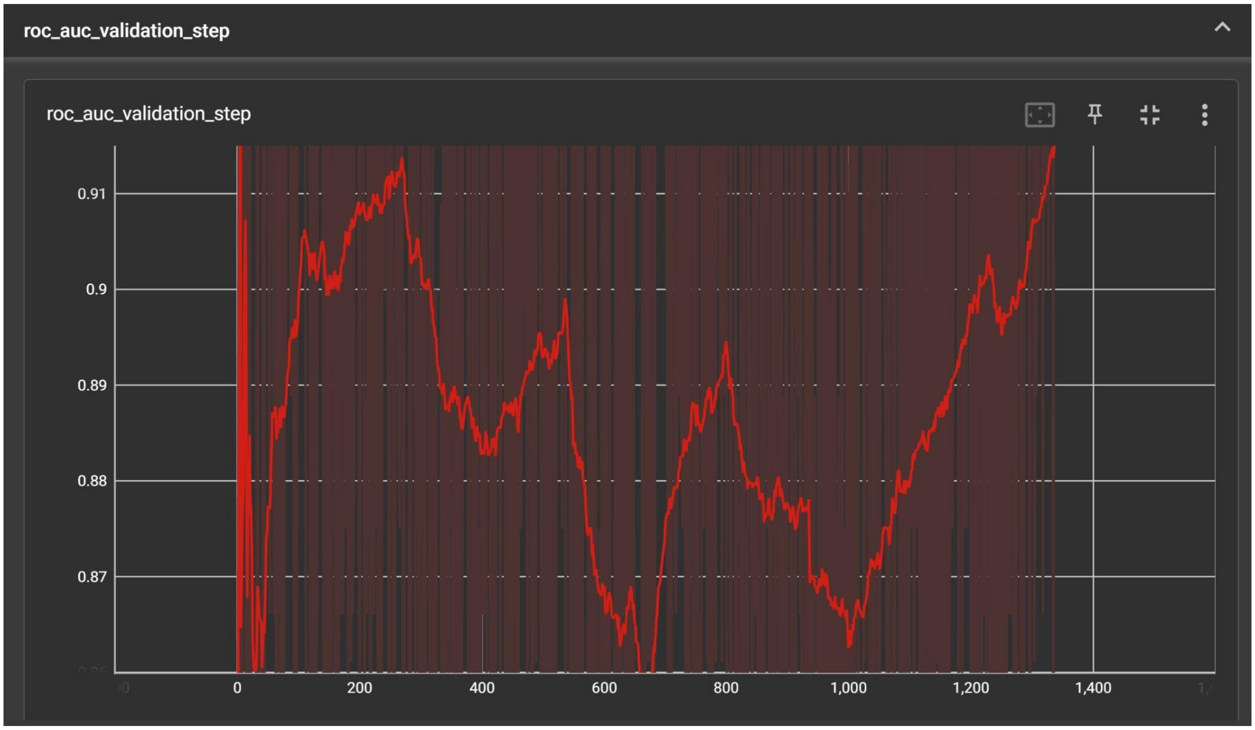
Discriminator Loss 1



Discriminator Loss 2



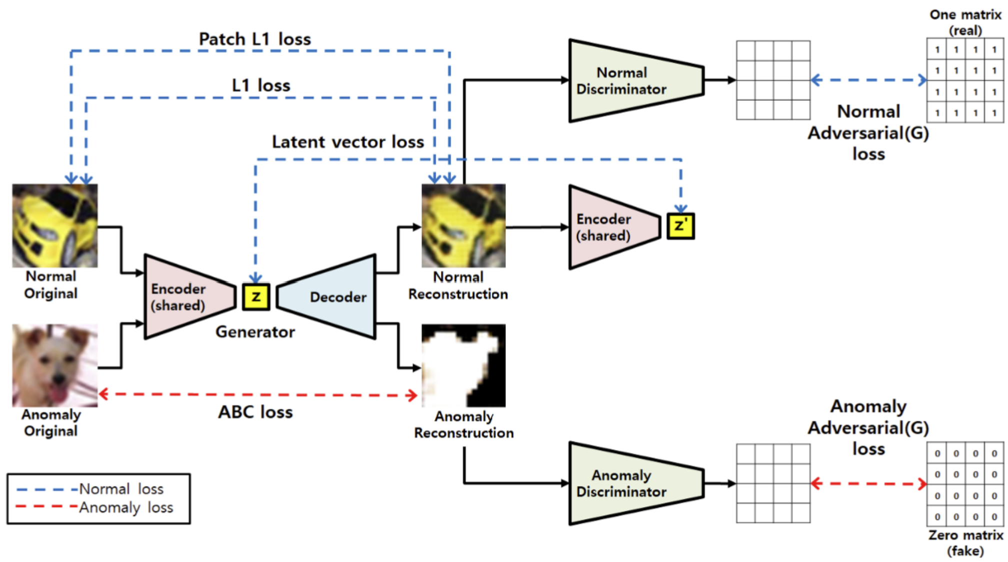
Validation AUROC



SOTA Model:

Here, I am using DCGAN instead of GAN. And the losses will be calculated according to the SOTA model paper.

Model Architecture:

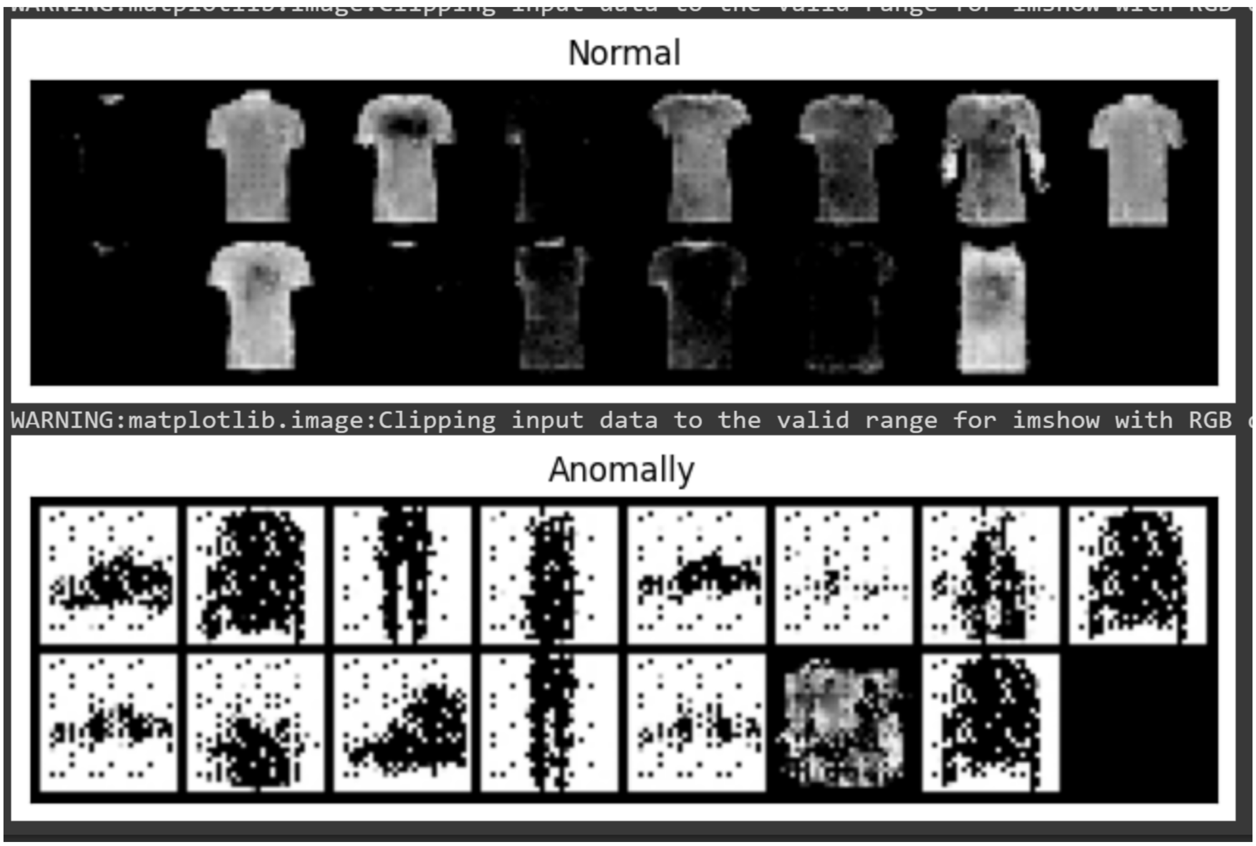


Hyperparameter of the Model:

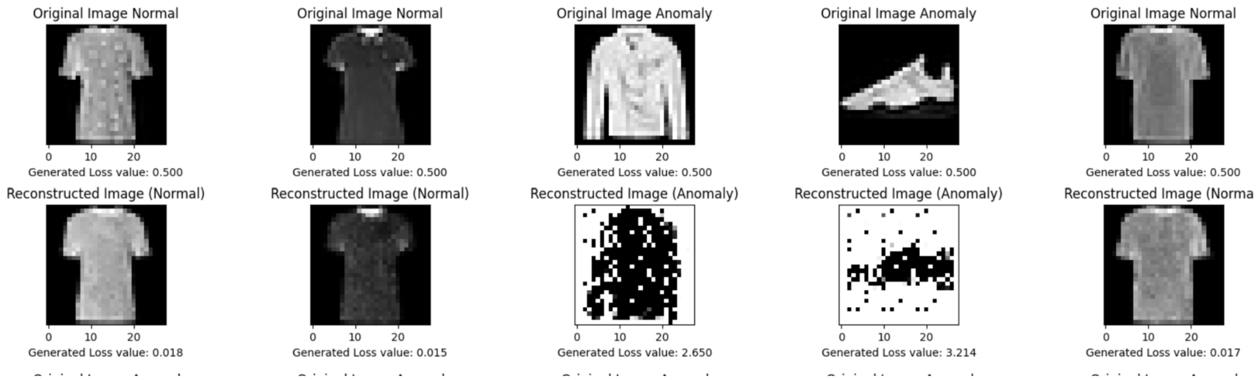
- The convolutional layers have kernel sizes of 3x3, strides of 2, and padding of 1
- The number of output channels for these layers are 32, 64, and 128
- Activation Function: ReLU, and Sigmoid (For binary classification - Normal or Anomaly)
- Optimizer of Generator : Adam (learning rate =1e-3, weight_decay=1e-5)
- Optimizer of Discriminator 1: Adam (learning rate =1e-3, weight_decay=1e-5)
- Optimizer of Discriminator 2: Adam (learning rate =1e-3, weight_decay=1e-5)
- Loss Function (Discriminator):
  - A-b coding scheme loss
- Loss Functions (Normal Generator):
  - L1 reconstruction

    - Patch loss
    - Latent vector loss
    - Adversarial Loss
- Loss Functions (Anomaly Generator):
    - ABC loss
    - Anomaly adversarial loss

Model Results (Training Data):



Model Results (Testing Data):



‘The model has not generated noise now, It’s generating approx the same result as a Normal image and changing the image for an anomaly image.

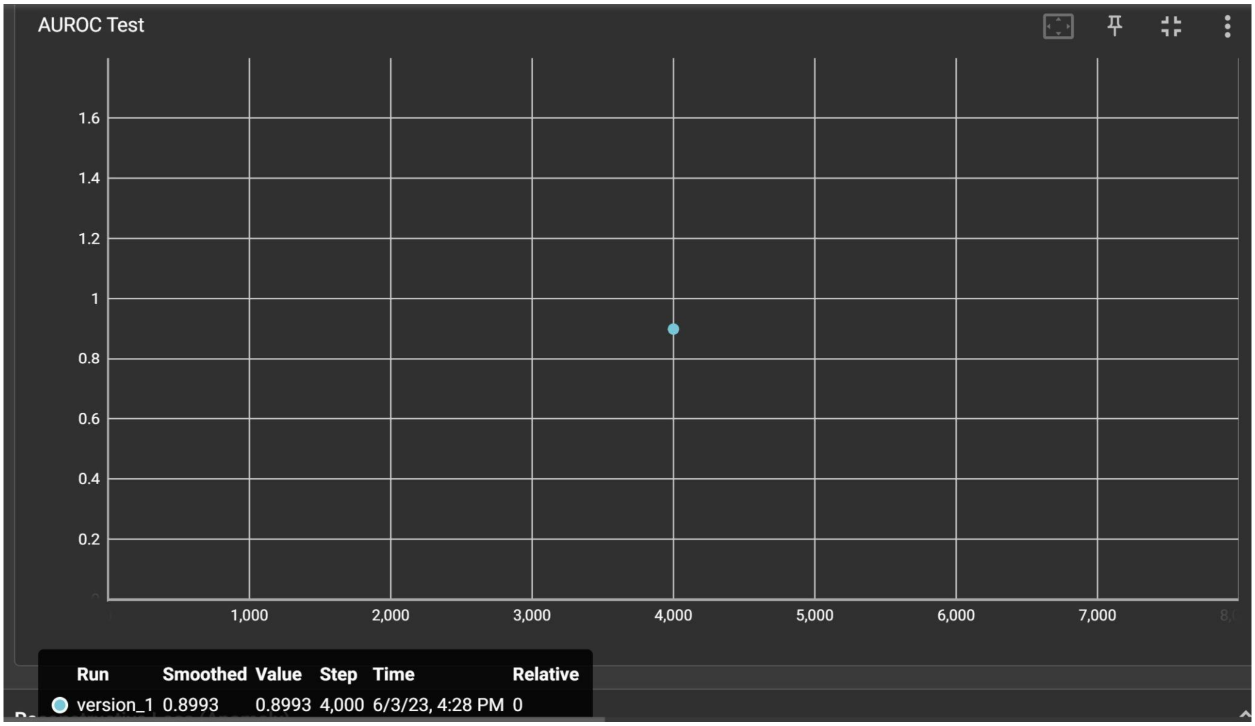
Model AUROC Result:

Test metric	DataLoader 0
AUROC Test	0.8992706537246704

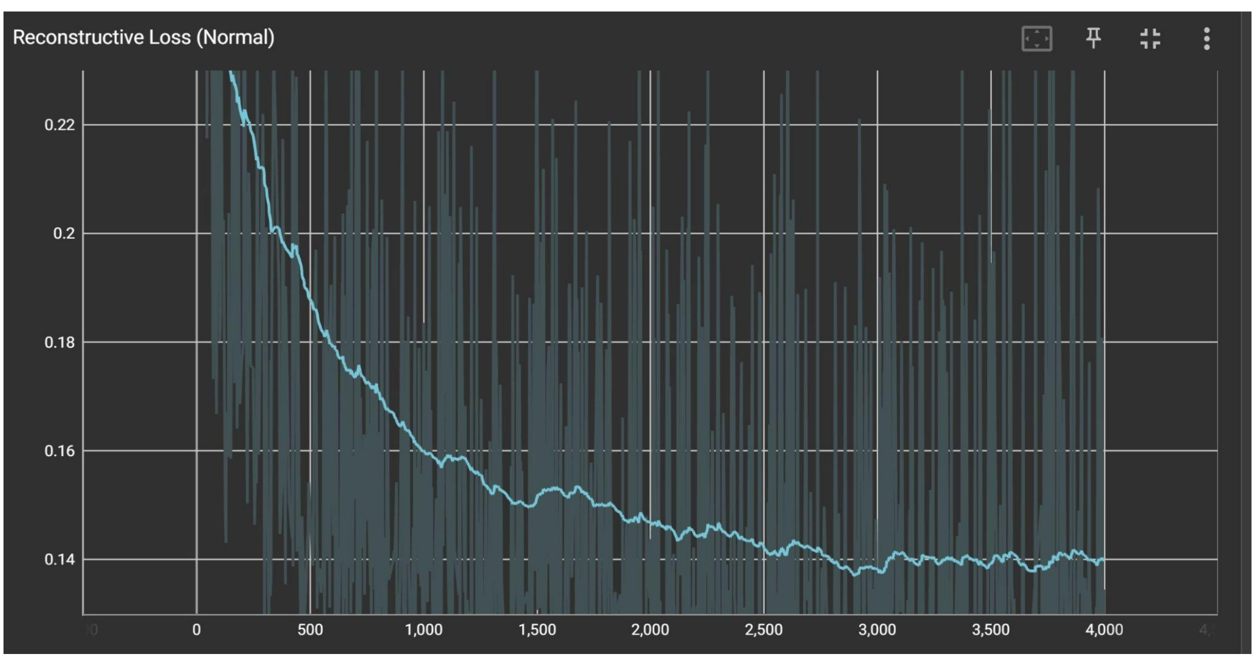
AUROC result = Approx 90%

Tenorboard Results:

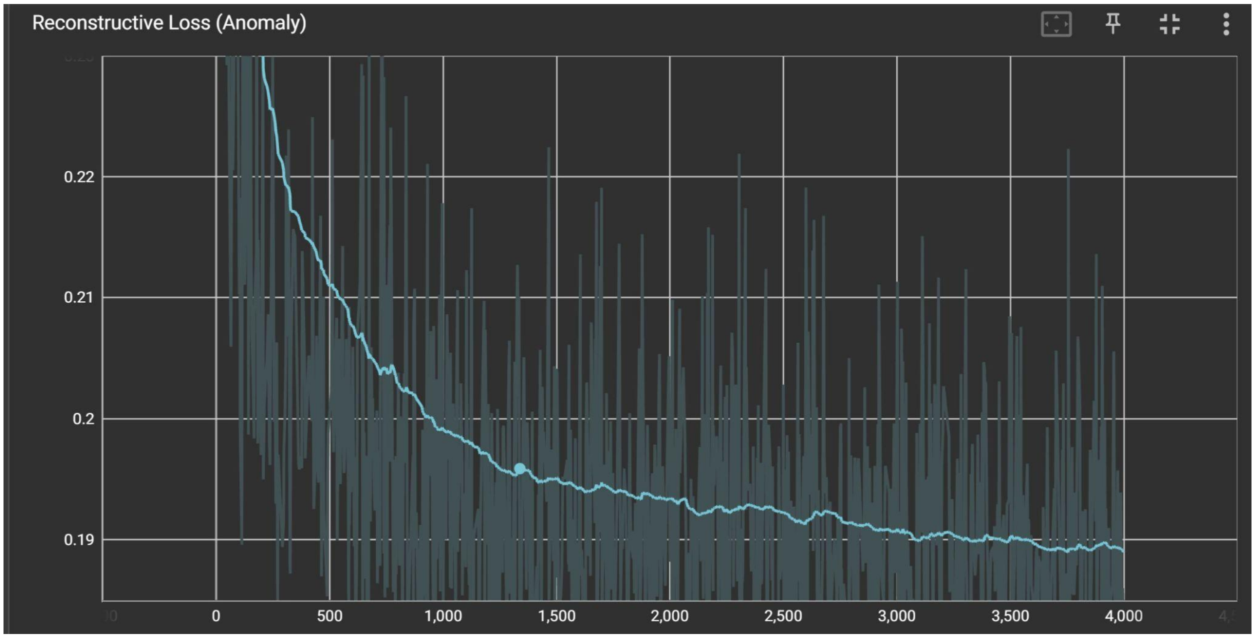
AUROC Test:



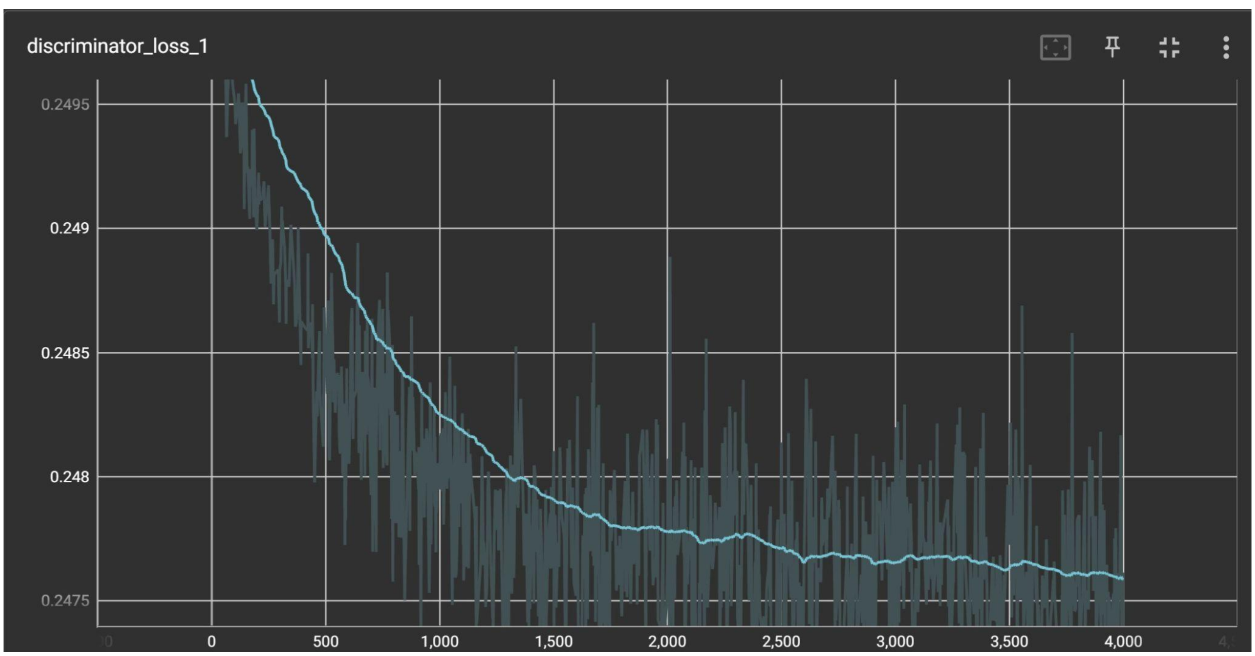
Generative Loss (Normal)



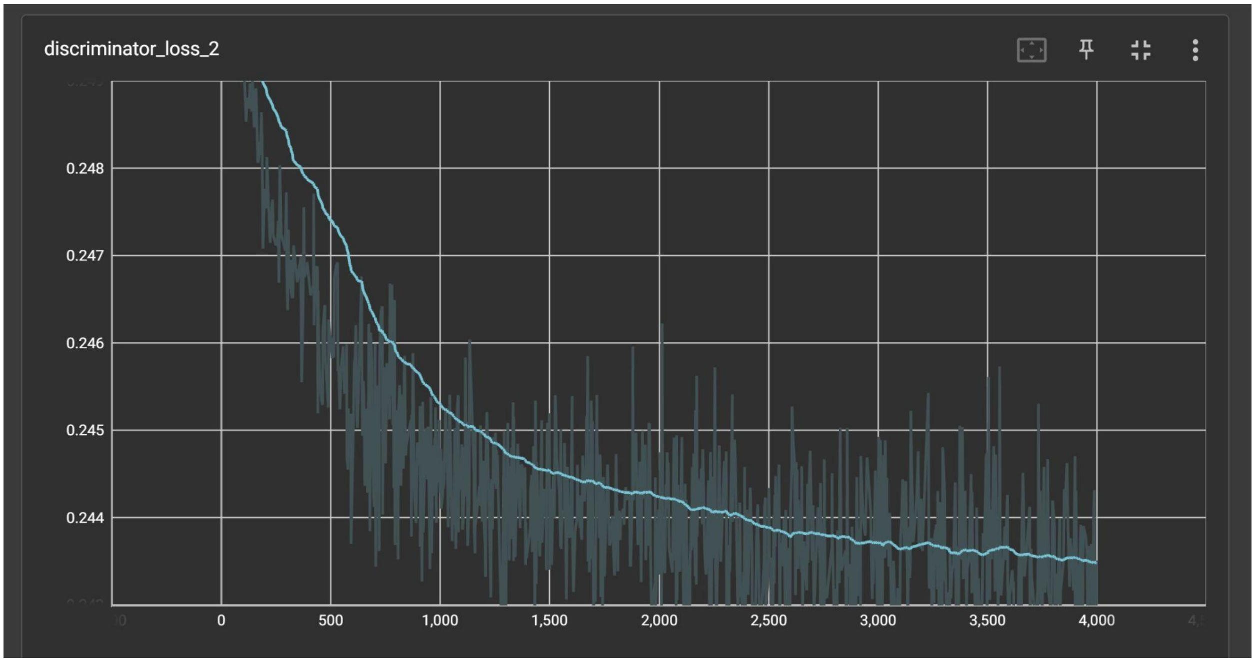
Generative Loss (Anomaly)



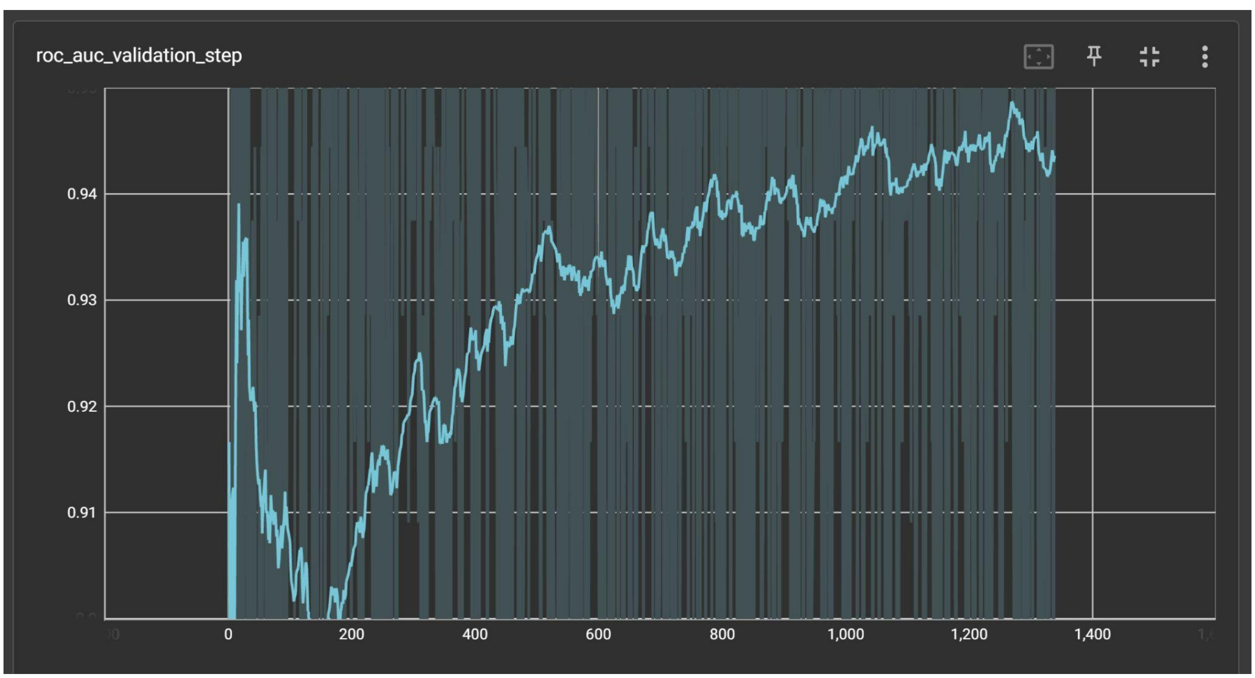
Discriminator Loss 1



Discriminator Loss 2



Validation AUROC



Conclusion:

The anomaly detection task for the FashionMNIST dataset is that the SOTA model performs significantly better compared to the baseline model. The SOTA model, based on DCGAN architecture and utilizing specific loss functions designed for anomaly detection, demonstrates superior performance in accurately identifying anomalies within the dataset. By employing convolutional layers, the SOTA model captures spatial features and patterns in the images more effectively, resulting in higher-quality and more realistic image generation. Additionally, the adoption of specialized loss functions tailored for anomaly detection enhances the model's ability to distinguish between normal and anomalous instances, leading to improved AUROC results. Overall, the SOTA model presents a more advanced and effective approach for anomaly detection in the FashionMNIST dataset compared to the baseline model. The results suggest that utilizing DCGAN architecture and incorporating specific loss functions can significantly enhance the model's ability to identify anomalies, providing valuable insights for practical applications in the fashion domain.