Topic:

Anomaly Detection in Dataset Using Different Technique

Submitted by:

Name: Shahkar Javid

Matricola: 2047305

Email: javid.2047305@studenti.uniroma1.it

Submitted To:

Prof. **Fabrizio Silvestri**

1. INTRODUCTION

Our focus in this work is to detect an anomaly in a dataset, in particular we will test our approaches on Fashion MNIST dataset. To this end, we adopt the anomaly detection method to identify unusual patterns to address such challenging problems. We have tried three approaches that have shown effective result in this field. 1) auto-encoder based technique, 2) Anomaly Detection using GANs, 3) a novel Sota method that was proposed in the paper [1] which also consist of GAN based architecture with two Discriminators and optimization was performed using different losses.

2. Description

2.1 Dataset and Preprocessing

We use well-known Fashion-MNIST dataset for our experiments, it consists of 28×28 grayscale images of 70,000 fashion products from 10 categories, with 7,000 images per category. The training set has 60,000 images and the test set has 10,000 images. Fashion-MNIST shares the same image size, data format and the structure of training and testing splits with the original MNIST.

To design an anomaly detection algorithm, what we did is that we need to keep on class of the dataset as normal data and rest of the classes of the dataset as anomaly. For instance, we consider class 0 of Fashion-MNIST data set i.e., “Shirt” a normal class and rest of the classes of the dataset as anomaly data. This setup give rise to another problem during training that might be the distribution between normal data and anomaly data because the quantity of anomaly dataset is way more than normal to solve this problem after some research on internet we use data sampling technique which try to distribute the batches equally during training between normal class and anomaly class.

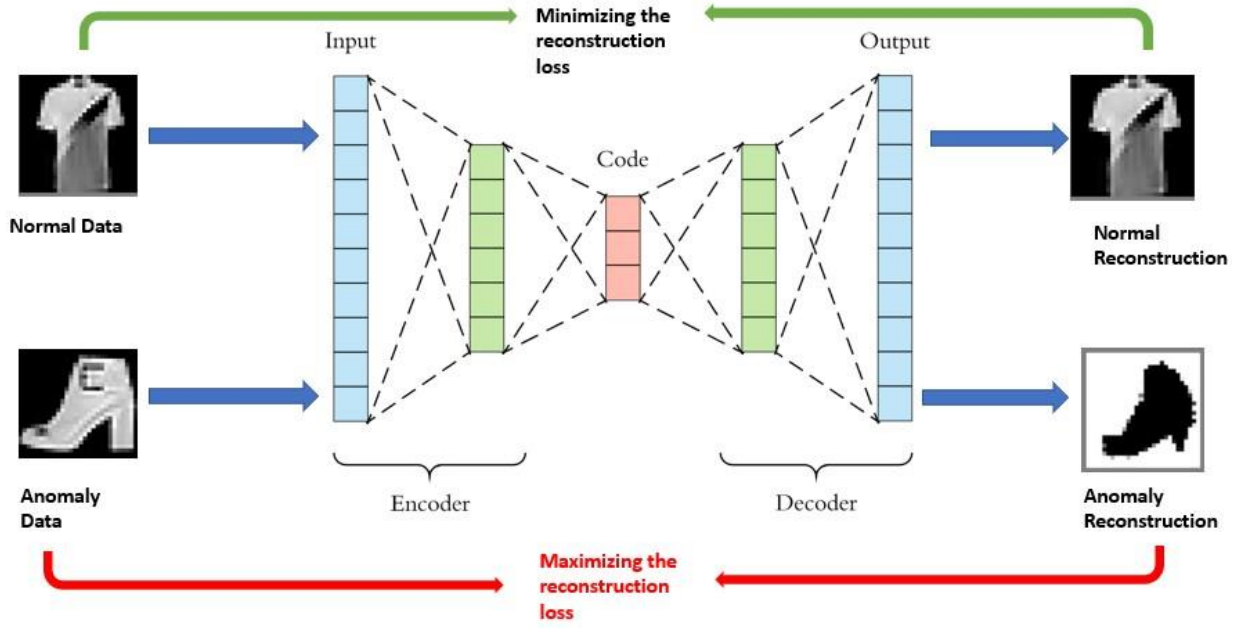
2.2 Hyper Parameters

We tried many different hyperparameters but below were found to be working best for our models;

Batch Size = 32
Input Size = 28
Epoch =15
Optimizer = Adam optimizer
Loss Function = MSE loss
Learning rate = 1e^-3

2.3 Baseline Technique (Auto-Encoder)

The first method that we tried was using an autoencoder, Autoencoder is an unsupervised learning technique for neural networks that learns efficient data encoding by training the network to ignore signal noise. It simply consist of a simple encoder decoder network that when given an input image tries to reconstruct out the same image with minimum reconstruction loss.



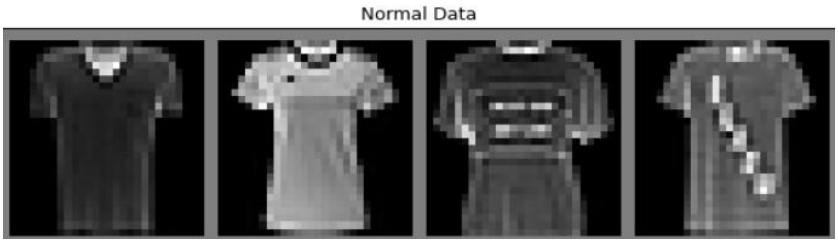
So, what we did is after splitting the data between anomaly and normal data we feed that data to our autoencoder based architecture and during training we try to minimize the reconstruction loss for normal data and maximize the reconstruction loss for Anomaly data. We were also able to get a decent losses for our model as seen from figure below;

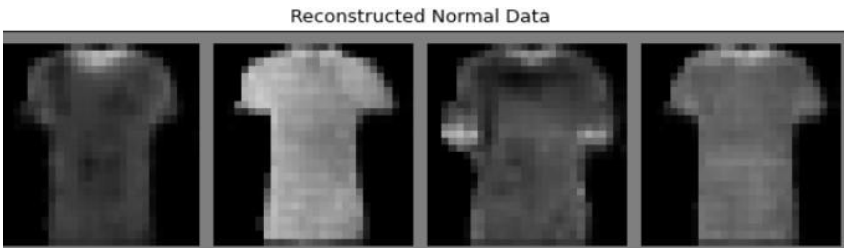
Test metric	DataLoader 0
Anomaly_loss	0.8319480419158936
Normal_loss	0.05370621383190155

The result that we got were pretty good both in term of visual and AUROC [2] which can be observed below;

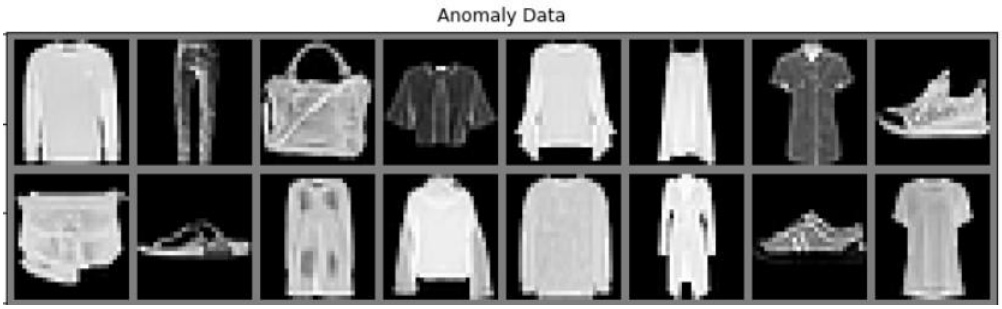
2.3.1 Results:

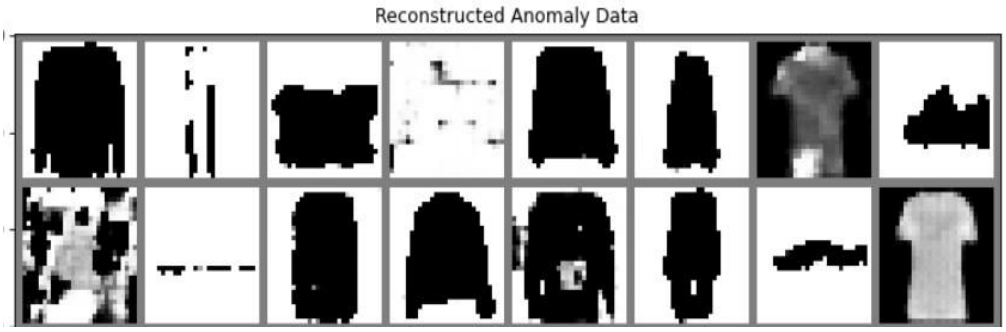
Normal Data



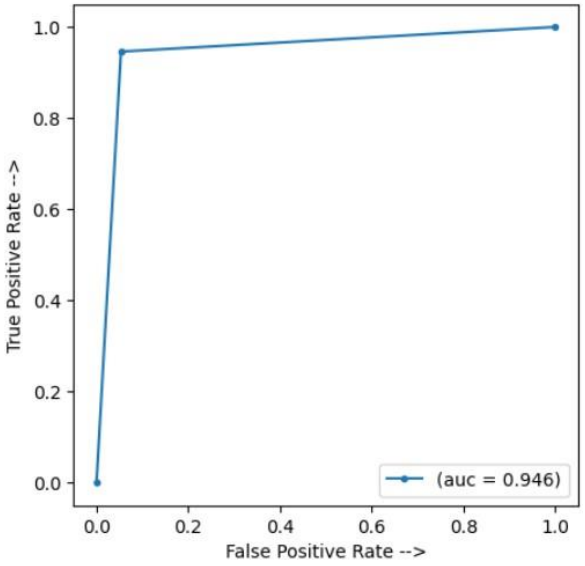


Anomaly data



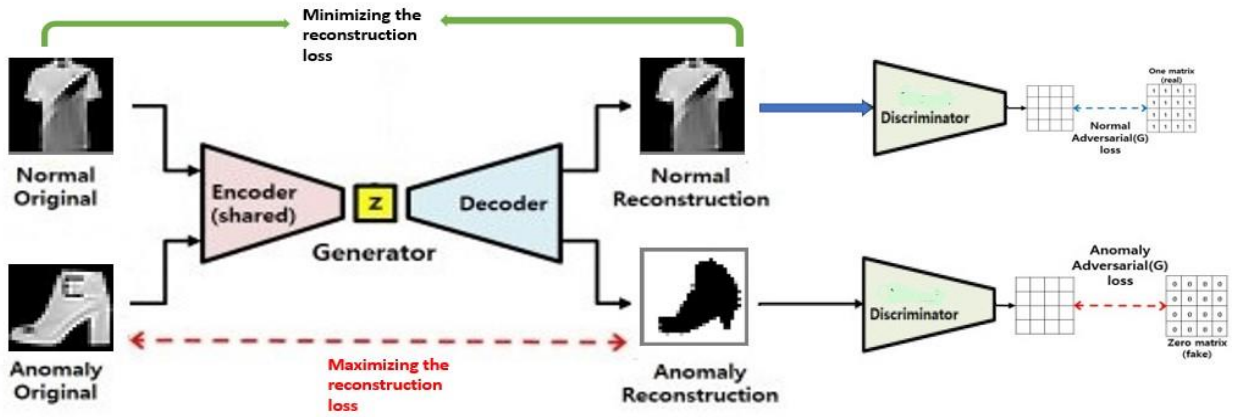


AUROC for class 0



2.4 Additional Non-Sota Techniques (GAN)

The next method we tried was using GAN model which consist of simple generator and discriminator architecture. The generator's job is to create realistic-looking data, such as images, while the discriminator's role is to distinguish between real and fake data. The discriminator basically takes this generated sample and tries to determine if it's real or fake. The generator aims to generate samples that the discriminator cannot distinguish as fake, while the discriminator tries to become better at identifying fake samples. This back-and-forth training continues until the generator becomes skilled at producing realistic data that can fool the discriminator into thinking it's real.



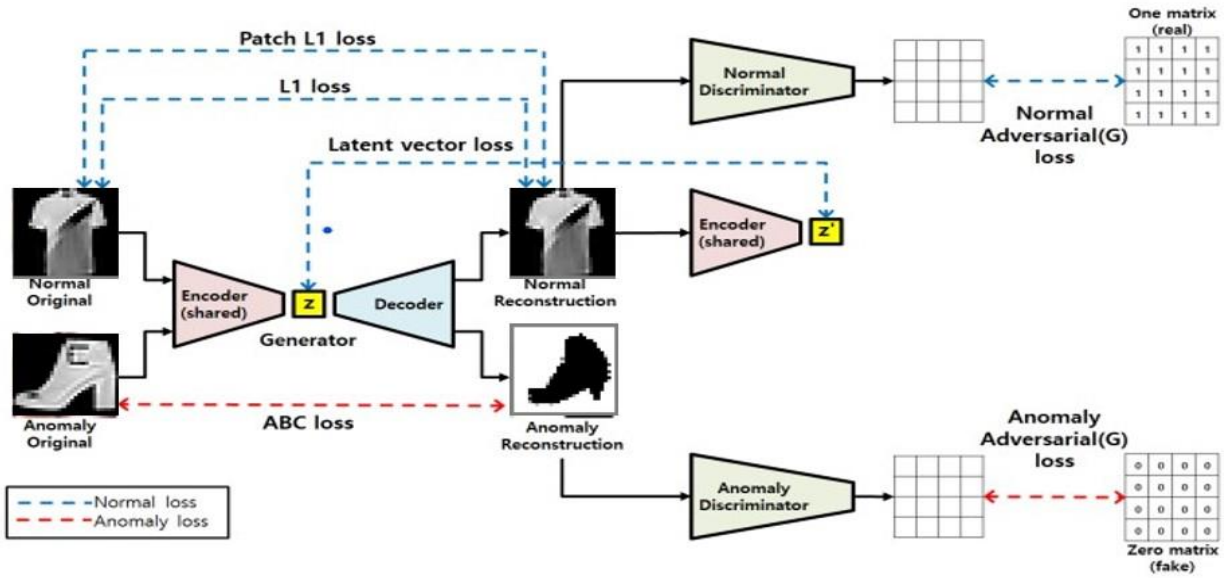
To use GAN based architecture for anomaly detection here we did the same as we did for our autoencoder network, we tried to minimize the reconstruction loss for normal data and maximize the reconstruction loss for Anomaly data. The result was almost similar to the autoencoder model both in term of visual and AUROC terms

2.5 SOTA Approach

So, after trying the simple method to detect an anomaly we also tried the method that was described in the sota paper[1]. Which is also a Generative Adversarial Network (GAN) based architecture consist of a generator and two separate discriminators. Generator model is just like a simple U-net or an autoencoder and discriminator is a decoder like network.

As per the official published paper [1], When the discriminator receives the reconstructed data, the generator is trained to produce a score of 1 for normal data and 0 for anomaly data. So, when training the model with both normal and anomaly data, using just one discriminator only teaches the model to classify normal images accurately. To overcome this issue, it is necessary to have separate discriminators for normal data and anomaly data. This method only adds more parameters or computations during the training phase, but it doesn't affect the parameters or computations during the inference phase

During training for anomaly detection a lot of different kind of losses were considered which can be shown from the figure below.



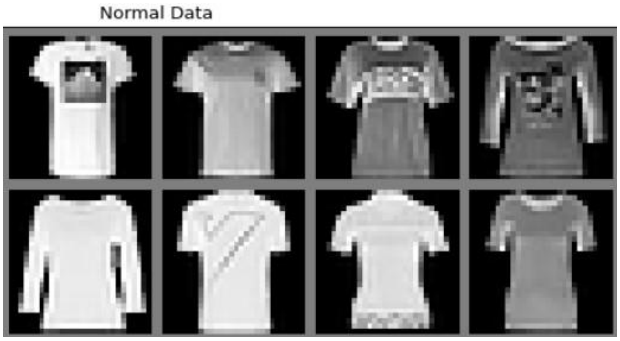
2.5.1 Results:

Below you can observe the losses , images with ground truth and after training for both anomaly and normal data.

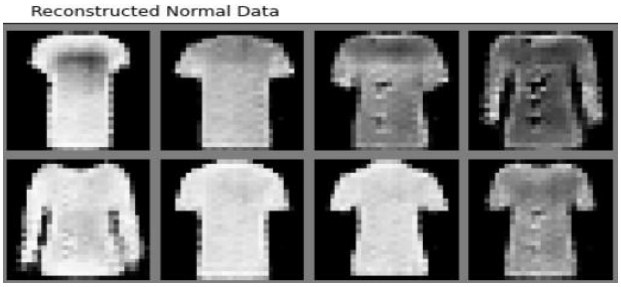
Test metric	DataLoader 0
Anomaly_loss	[illegible]
Normal_loss	[illegible]

Anomaly loss= 0.824, normal data loss= 0.052.

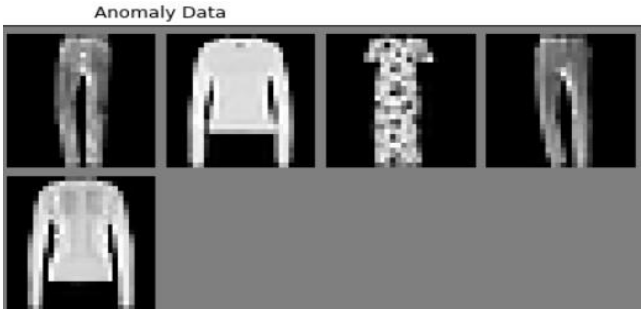
Ground truth



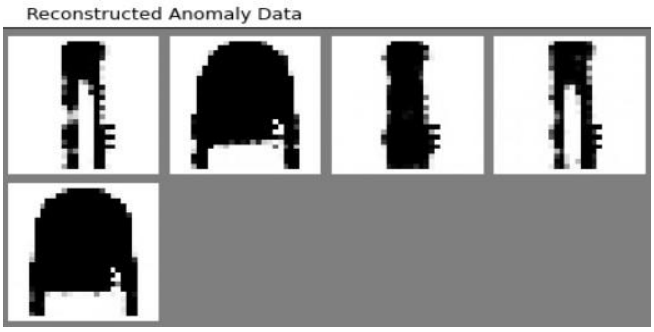
Reconstruction of Normal Class



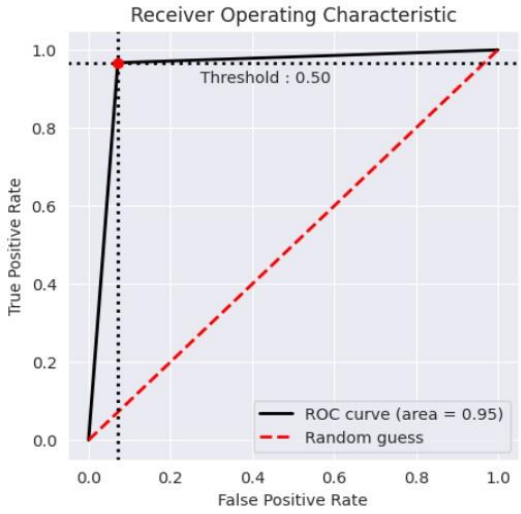
Anomaly Ground Truth



Anomaly Reconstruction after Training



AUROC for Class 0



3. Conclusion

So after performing same experiment on different model, we observed that result was almost the same including the baseline technique which was pretty simple architecture of auto encoder still result was almost same as the Sota.

4. References

Below are the references that were used in this project;

[1]. GAN-based Anomaly Detection in Imbalance Problems (https://paperswithcode.com/paper/gan-based-anomaly-detection-in-imbalance)

[2]. Area Under the Receiver-Operating Characteristics (https://scikitlearn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html)

[3].