



Name: **Muhammad Nabeel**

Assignment Type: **Internship Task 4**

Supervisor/Instructor: **Dr. Fehmida Usmani**

CMS: **413522**

Task: To leverage transformer models to enhance the ability to differentiate between class 1 samples and noise. Specifically, you will use a dataset containing only samples of label 1, apply transformers to classify between original class 1 data and artificially generated noise.

Data Description:

The dataset contains 20,468 entries of data containing single column feature (SOPAS) and one output labels (only of Class 1).

Data Preprocessing and Outlier Detection:

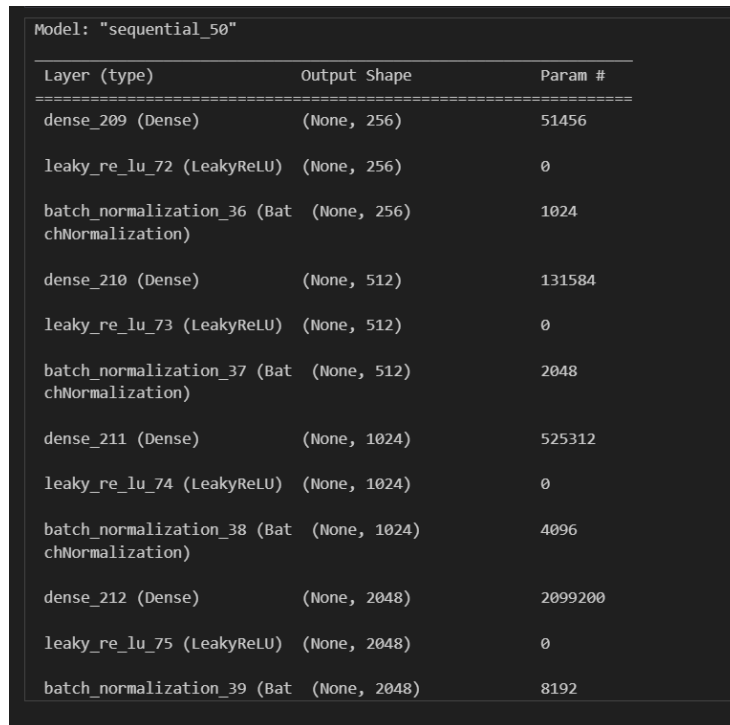
For the Data-Preprocessing, I converted the plain rows of data to window-sequences of size 30 so the transformer model can better understand the sequenced entries. And normalized the entries using Standard-Normalization.

Noise Generation:

For the sake of Noise generation, I trained a GAN for mimicking the Outliers present in the dataset.

GAN Training:

Generator Architecture Used:



```
Model: "sequential_50"
Layer (type)                Output Shape              Param #
=====
dense_209 (Dense)            (None, 256)               51456
leaky_re_lu_72 (LeakyReLU)   (None, 256)               0
batch_normalization_36 (Bat (None, 256)               1024
chNormalization)
dense_210 (Dense)            (None, 512)               131584
leaky_re_lu_73 (LeakyReLU)   (None, 512)               0
batch_normalization_37 (Bat (None, 512)               2048
chNormalization)
dense_211 (Dense)            (None, 1024)              525312
leaky_re_lu_74 (LeakyReLU)   (None, 1024)              0
batch_normalization_38 (Bat (None, 1024)              4096
chNormalization)
dense_212 (Dense)            (None, 2048)              2099200
leaky_re_lu_75 (LeakyReLU)   (None, 2048)              0
batch_normalization_39 (Bat (None, 2048)              8192
chNormalization)
```

Discriminator Architecture:

```
dense_214 (Dense)          (None, 256)          2816
leaky_re_lu_76 (LeakyReLU) (None, 256)           0
dropout_132 (Dropout)       (None, 256)           0
dense_215 (Dense)          (None, 512)         131584
leaky_re_lu_77 (LeakyReLU) (None, 512)           0
dropout_133 (Dropout)       (None, 512)           0
dense_216 (Dense)          (None, 1024)        525312
leaky_re_lu_78 (LeakyReLU) (None, 1024)          0
dropout_134 (Dropout)       (None, 1024)          0
dense_217 (Dense)          (None, 2048)       2099200
leaky_re_lu_79 (LeakyReLU) (None, 2048)          0
dropout_135 (Dropout)       (None, 2048)          0
dense_218 (Dense)          (None, 1)            2049

=====
Total params: 2,760,961
Trainable params: 2,760,961
Non-trainable params: 0
```

- The GAN was trained on 5000 Epochs.

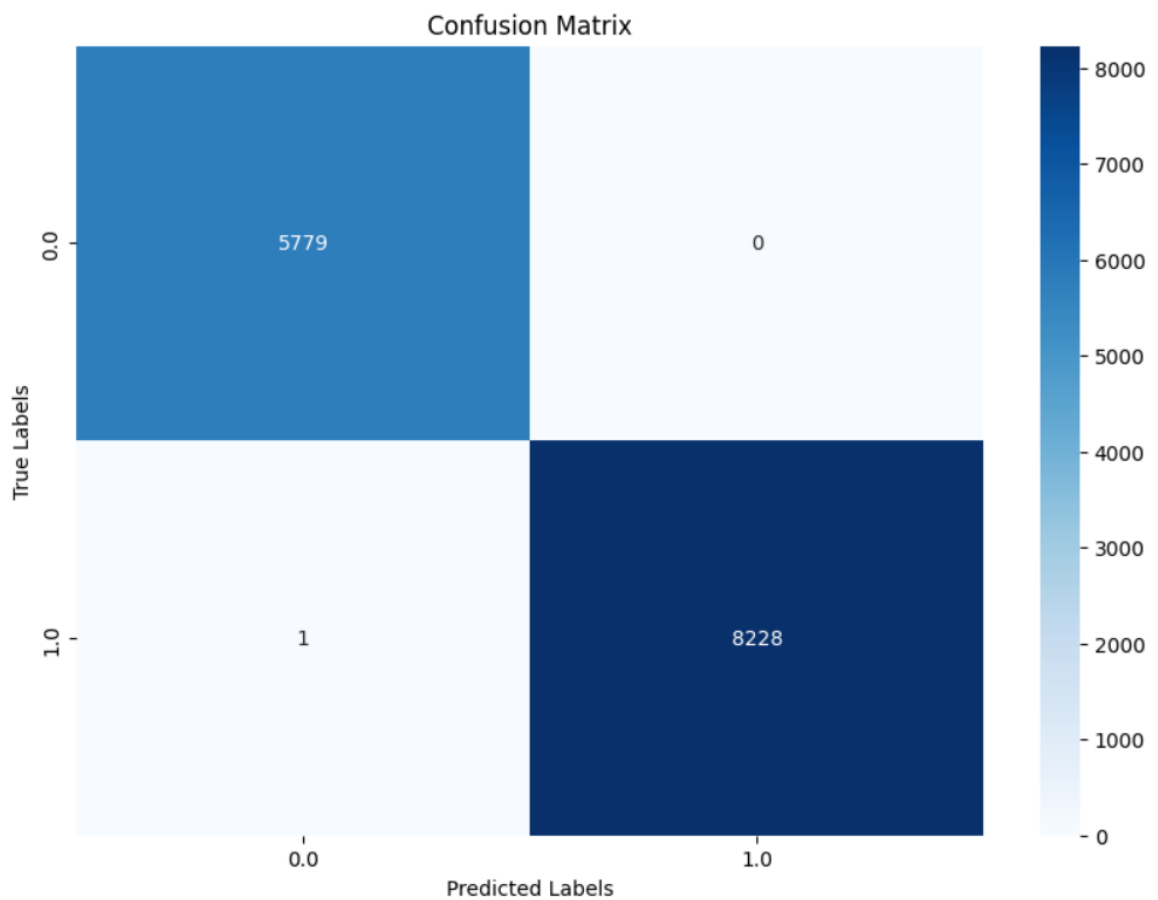
Transformers Architecture:

... Model: "model_22"

Layer (type)	Output Shape	Param #
=====		
input_23 (InputLayer)	[(None, 1, 10)]	0
dense_192 (Dense)	(None, 1, 4)	44
positional_encoding_22 (PositionalEncoding)	(None, 1, 4)	0
transformer_block_22 (TransformerBlock)	(None, 1, 4)	136
global_average_pooling1d_22 (GlobalAveragePooling1D)	(None, 4)	0
dropout_124 (Dropout)	(None, 4)	0
dense_195 (Dense)	(None, 20)	100
dropout_125 (Dropout)	(None, 20)	0
dense_196 (Dense)	(None, 1)	21
=====		
Total params: 301		
Trainable params: 301		
Non-trainable params: 0		

A transformers architecture consists of a single head Attention Layer along with Positional Encoding layer for Context analysis.

Confusion Matrix:



Classification Report:

	Class	precision	recall	f1-score	support
0.0	0.0	0.9998269896193772	1.0	0.9999134873258932	5779.0
1.0	1.0	1.0	0.9998784785514643	0.9999392355836422	8229.0
accuracy	accuracy	0.9999286122215877	0.9999286122215877	0.9999286122215877	0.9999286122215877
macro avg	macro avg	0.9999134948096886	0.9999392392757321	0.9999263614547678	14008.0
weighted avg	weighted avg	0.9999286245724144	0.9999286122215877	0.9999286131406431	14008.0

The Transformer model and GAN model is attached in the zip.