POWER SYSTEM DATASET INTRODUCTION AND APPLICATIONS

Presenter: Hussein Alkhafaji
Advisor: Dr. mohamed Mahmoud
Graduate Mentor (if any): Dr. islam

Research Objective and Goals

- To build a machine learning model that can accurately classify power system events as either:

  -Natural occurrences (e.g., equipment failures, weather issues)

  -Cyberattacks (e.g., data manipulation, false signals)

- Explore whether deep learning, especially Convolutional Neural Networks (CNNs), can outperform traditional models in cyber-physical system security.

- Explore using model for federated learning.

Data Set Overview

Overview

The dataset includes 15 sub-datasets with a total of 37 power system scenarios.
Scenarios are divided into:

- 8 natural events
- 1 no-event (normal)
- 28 cyber attack events.

The dataset is grouped into three classification types:

- Binary classification (normal vs. attack)
- Three-class (normal, natural, attack)
- Multi-class (each attack/natural event type is its own class).

The 5 main event types

- Short-circuit faults
- Line maintenance
- Remote tripping command injection (attack)
- Relay setting change (attack)
- Data injection (attack)

Goal

The main objective is to detect and classify power system events, including attacks.
Events are categorized into scenarios and classes.
The dataset is constructed using data from two key sources:

Phasor Measurement Unit (PMU): Records real-time measurements such as voltage, current, and frequency.

Control Logs: Record actions taken by the system or operators. Each event or scenario is recorded every 2 seconds over a duration of 180 seconds.

Structure of the 3 different datasets classifications

Binary Class (Two-Class) Structure: Normal state or Attack

Feature: State label – Attack: Yes or No

TABLE V. BINARY CLASSIFICATION

	Attack Events	Normal Operation
Scenarios	7,8,9,10,11,12,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,35,36,37,38,39,40	1,2,3,4,5,6,13,14, 41

Three-Class Structure: Normal (no-event) , Natural Event (equipment failure or maintenance) or Attack Event.

Feature: State and Attack Label – Scenario type (e.g., natural fault or cyberattack)

TABLE IV. THREE-CLASS CLASSIFICATION GROUP

	Attack Events	Natural Events	No Events
Scenarios	7,8,9,10,11,12,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,35,36,37,38,39,40	1,2,3,4,5,6,13,14	41

Multi-class

Structure: 37 unique scenarios (Normal, Natural Events, and specific Attack types)

Feature: Scenario 1-37, each unique scenario is treated as a sperate feature.

label – Specific event or attack type

Features power system data set

Feature	Description
PA1:VH – PA3:VH	Phase A - C Voltage Phase Angle
PM1: V – PM3: V	Phase A - C Voltage Phase Magnitude
PA4:IH – PA6:IH	Phase A - C Current Phase Angle
PM4: I – PM6: I	Phase A - C Current Phase Magnitude
PA7:VH – PA9:VH	Pos. – Neg. – Zero Voltage Phase Angle
PM7: V – PM9: V	Pos. – Neg. – Zero Voltage Phase Magnitude
PA10:VH - PA12:VH	Pos. – Neg. – Zero Current Phase Angle
PM10: V - PM12: V	Pos. – Neg. – Zero Current Phase Magnitude
F	Frequency for relays
DF	Frequency Delta (dF/dt) for relays
PA:Z	Appearance Impedance for relays
PA:ZH	Appearance Impedance Angle for relays
S	Status Flag for relays

Total of 128 features.

29 types of which are recorded from the phasor measurement unit or synchrophasor.
There are a total of 4 PMUs which contribute about 116 Measurement columns. First 116 columns in the dataset.

Next 12 columns or features are the control panel logs, snort Alerts and relay logs. Last column is marker or label.

Using power system dataset to develop a Hybrid Intrusion Detection System

Detection system training
•Used to train a Hybrid Intrusion Detection System (HIDS)
•Based on Common Path Mining (CPM), which learns:
- Normal behavior patterns
- Malicious (cyberattack) patterns
- Natural fault patterns

Uses FP- growth algorithm to train on data set.
Common path mining is extremely similar to the FP- growth Algorithm expect it finds system states.

Detection & Classification
•Once sensor readings are received, the system:
- Checks if the new sequence matches a normal path
- Or an attack path
- Or is it an unknown/unseen behavior?

•This allows the system to:
- Detect known attacks
- Identify anomalies or unknown threats
- Distinguish between natural faults and malicious activity

How It Works: Common Path Mining
•CPM learns typical sequences from sensor data
•Builds "paths" that represent the system's behavior over time
•Helps identify deviations from expected patterns

Features used from the binary class data set:
Total of 15 features
PM4: I – PM6: I : Phase A - C Current Phase Magnitude
S : Status Flag for relays
Snort alert status for each relay
control panel remote trip status

Why Use the Power System Dataset?
•Provides labeled data for:
- Normal operations
- Natural events
- Cyberattacks (across 37 scenarios)

•Enables evaluation of:
- Accuracy in detecting attacks
- False positives and false negatives
- System's ability to detect new/unknown attacks

A stacked deep learning approach to cyber-attacks detection in industrial systems: application to power system

Methodology Summary – Stacked Deep Learning Model

- A stacked deep learning model of five feedforward networks
- Each network has three fully connected hidden layers
- Final prediction is made by averaging the outputs of all models

Implementation Details (Binary Classification)
Input Features: 122
Output Layer: 1 unit
Total Parameters: 19,221
Frameworks: TensorFlow + Keras
Optimizer: RMSprop
Epochs: 4,000
Batch Size: 128

Activation Functions

Hidden Layers: ReLU

Output Layer:
- Sigmoid for binary classification
- Softmax for multi-class classification

Important features include:

- Voltage Phase Magnitudes
- Voltage Phase Angles
- Current Phase Magnitudes
- Zero Voltage Phase Angles
- Zero Current Phase Magnitudes
- Appearance Impedance Measurements

Features excluded:
- 'R2-PM8:V', 'R1-PM8:V', 'R2-PM9:V'
- 'R4-PA9:VH', 'R2-PA8:VH', 'R3-PA8:VH'

A stacked deep learning approach to cyber-attacks detection in industrial systems: application to power system

Desired Evaluation Metrics:

Accuracy: Proportion of correct detections
Sensitivity: Ability to correctly identify attacks (True Positive Rate)
Specificity: Ability to correctly identify actual negatives (True Negative Rate)
Precision: Proportion of relevant (correct) positive predictions
F1-Score: Harmonic mean of Precision and Sensitivity
False Alarm Rate
Highest Difference

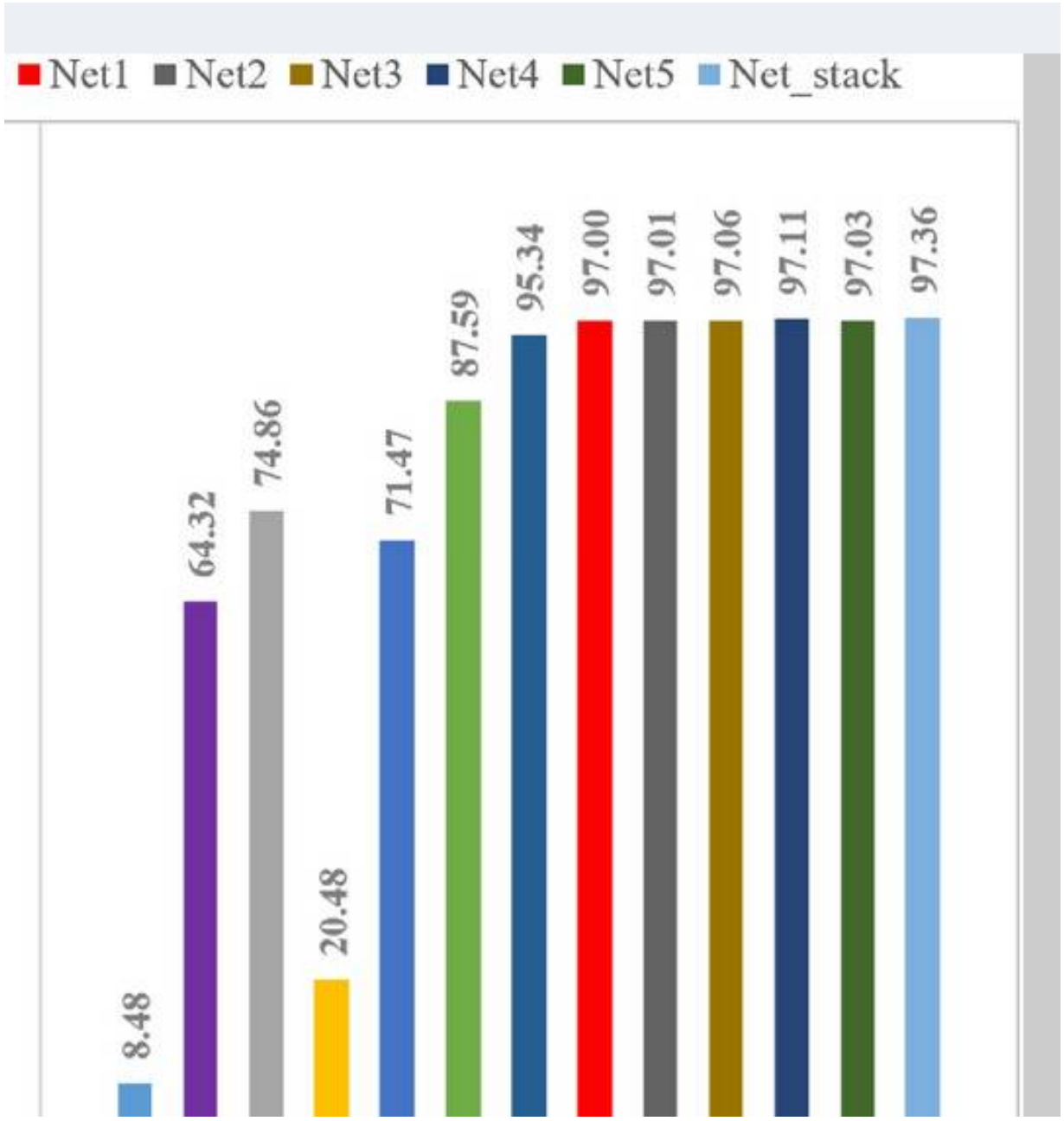
Interesting findings:

- A 3-layer deep network is sufficient for binary, three-class, and multiclass problems.
- Deeper networks tend to overfit due to the small dataset size.
- Model averaging boosts prediction stability and performance.

Data set sampling and results :
Dataset randomly sampled at 1% to test model efficiency on smaller samples.
Average accuracy for each of the five networks:
Network 1: 97.00%
Network 2: 97.01%
Network 3: 97.06%
Network 4: 97.11%
Network 5: 97.03%
(Binary intrusion detection over 15 datasets)

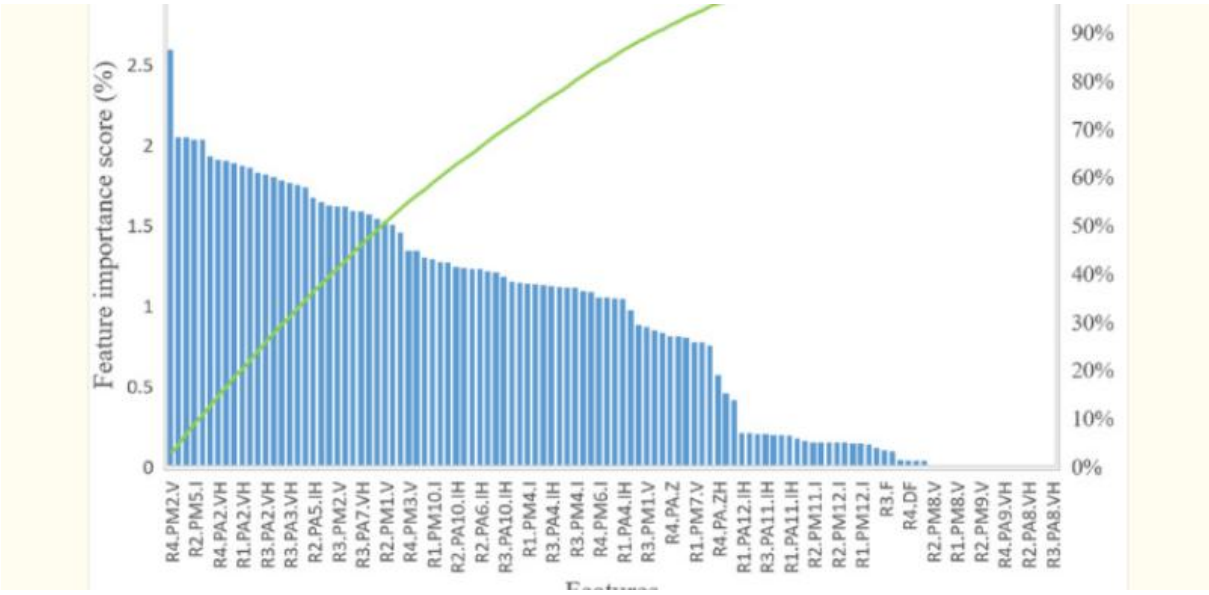
LAYER CONFIGURATIONS (HIDDEN NEURONS)

Model	Layer 1	Layer 2	Layer 3
1	80	60	60
2	80	80	60
3	100	80	80
4	120	100	80
5	180	120	80



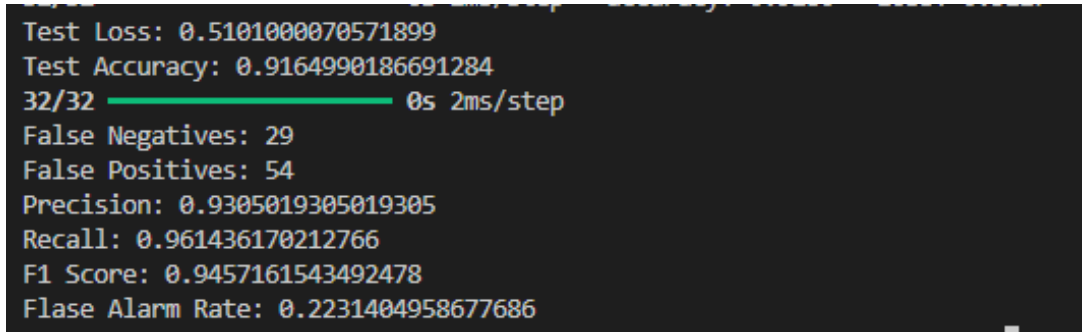
(c) Binary classification

Dataset 1	6.699	66.75	79.43	21.77	77.99	90.19	96.17	97.1	97.18	97.42	97.12	97	97.48
Dataset 2	9.33	57.42	73.21	16.27	70.33	85.65	94.74	96.69	96.88	96.8	96.82	97.02	97
Dataset 3	9.091	71.05	74.88	16.27	71.05	88.76	95.45	97.12	97.03	97.01	97.01	97.01	97.43
Dataset 4	10.29	55.26	71.05	22.73	66.27	86.6	95.93	96.94	97.12	97.19	97.37	97.23	97.56
Dataset 5	8.134	67.7	77.27	17.46	72.25	88.76	95.69	97.44	97.36	97.48	97.75	97.71	97.93
Dataset 6	6.459	61.72	72.97	17.23	70.57	84.69	94.98	97.22	97.16	97.28	97.3	97.3	97.7
Dataset 7	6.459	70.81	77.51	27.51	74.88	88.28	95.45	97.61	97.5	97.56	97.65	97.31	97.73
Dataset 8	9.569	71.77	74.88	30.14	71.29	89.23	95.45	96.95	96.88	97.1	97.18	97.06	97.25
Dataset 9	11.96	57.66	71.05	20.34	67.46	84.21	93.54	95.71	95.77	95.94	96.1	95.81	96.31
Dataset 10	9.33	67.7	74.16	16.75	71.05	89.47	95.69	96.77	96.32	96.62	96.73	96.64	97.06
Dataset 11	5.502	72.01	77.27	21.53	75.6	87.8	95.69	97.56	97.51	97.35	97.16	97.45	97.6
Dataset 12	11.24	57.66	70.1	16.51	66.27	87.32	95.22	96.55	96.57	96.55	96.65	96.38	96.9
Dataset 13	6.699	67.46	80.86	16.27	77.99	89.47	95.93	97.74	97.78	97.59	97.7	97.88	97.99
Dataset 14	4.306	68.18	76.32	27.99	73.92	87.32	95.69	97.32	97.36	97.44	97.5	97.46	97.63

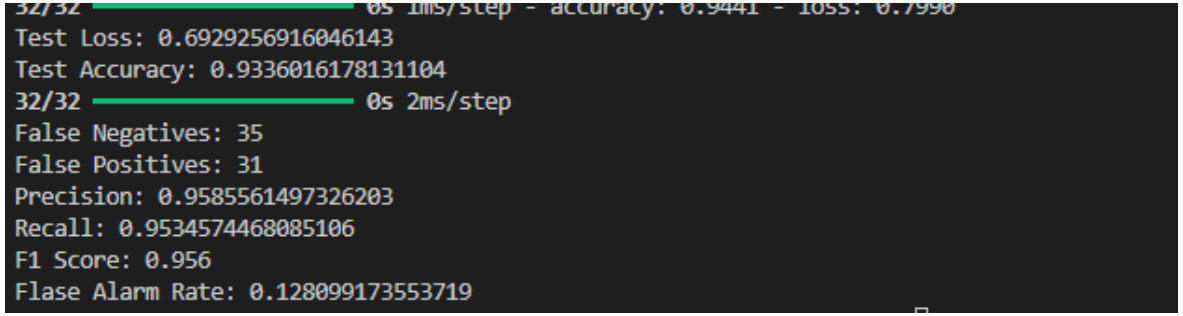


TEST RESULTS 1 FEEDFORWARD NN

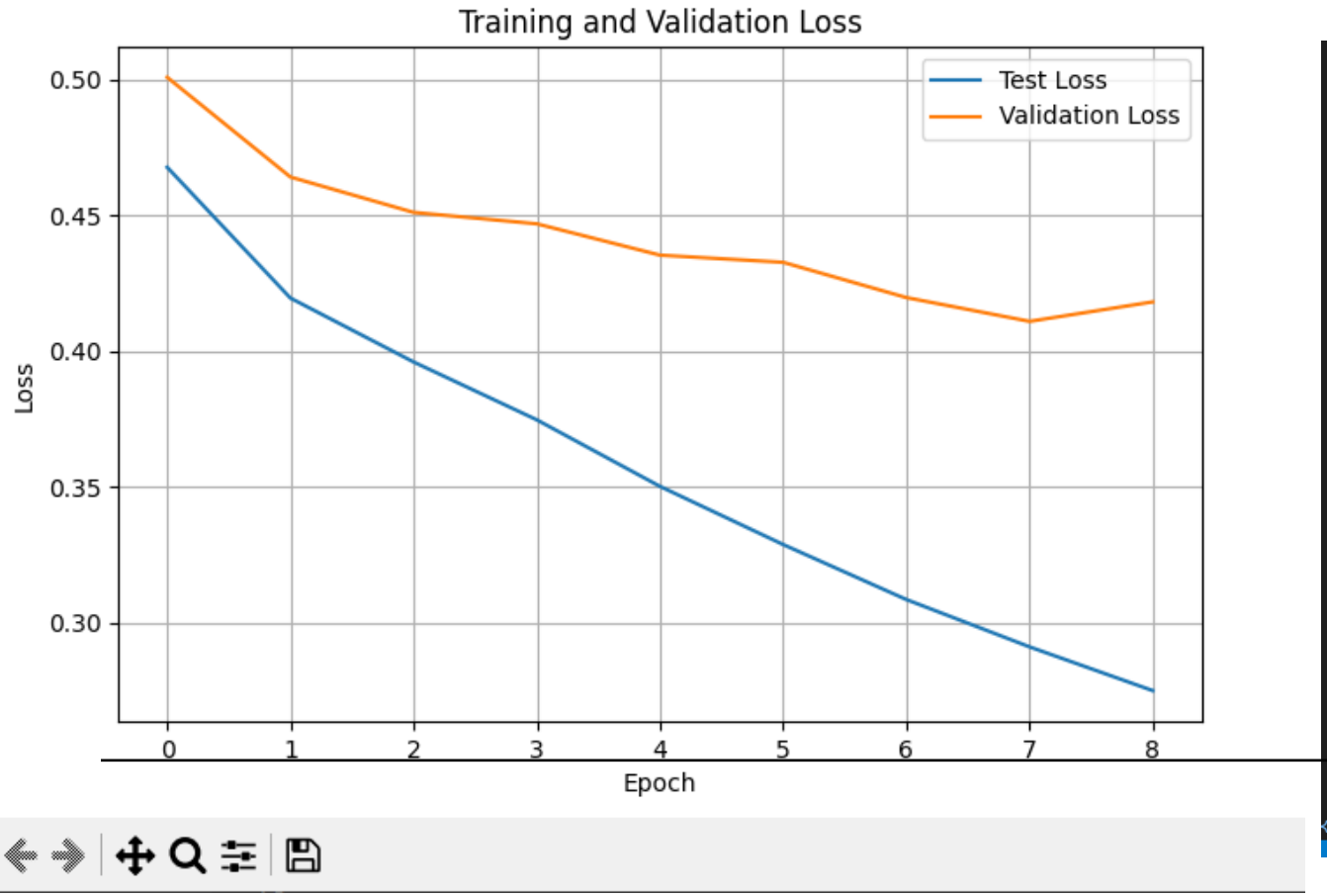
Epochs 100, batch size 64



Epochs 200, batch size 64



TESTING AFTER 6/27 : TRAINING SPLIT 80% TRAIN , 10% VALIDATION , 10% TEST , HYPERPARAMETER TUNING RESULTS

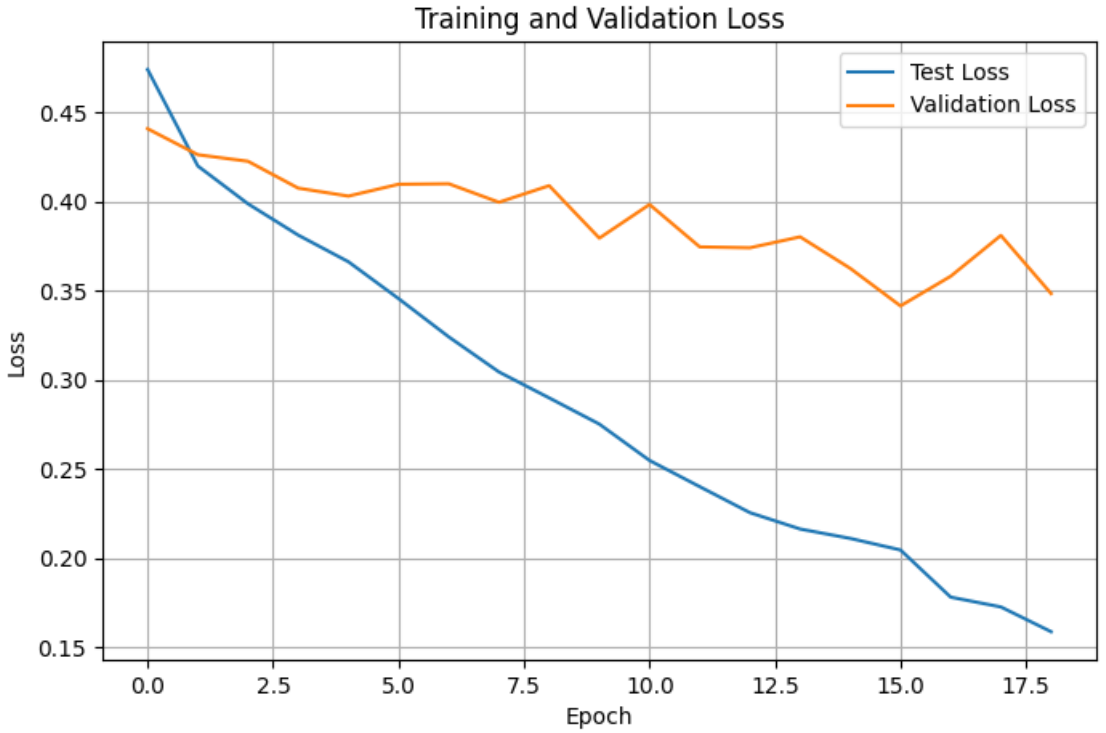



```
Lowest False Alarm Rate from top models: 0.0583
Input dim: 122
Model: "sequential_1"
```

Layer (type)	Output Shape	Param #
dense_4 (Dense)	(None, 160)	19,680
dense_5 (Dense)	(None, 140)	22,540
dense_6 (Dense)	(None, 60)	8,460
dense_7 (Dense)	(None, 1)	61

```
Total params: 101,484 (396.43 KB)
Trainable params: 50,741 (198.21 KB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 50,743 (198.22 KB)
16/16 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step
16/16 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step - accuracy: 0.9380 - loss: 0.5493
Test Loss: 0.4909776747226715
Final Validation Loss: 0.41821253299713135
Test Accuracy: 0.9315895438194275
Highest Difference (max absolute error): 0.8700486295313882
False Negatives: 27
False Positives: 7
Precision: 0.9803921568627451
Recall: 0.928381962864721 5
F1 Score: 0.9536784741144414
False Alarm Rate: 0.058333333333333334

Best Hyperparameters Found:
Best units: 160, 140, 60
Best batch size: 64
Best epochs: 140
Best learning rate (lowest FAR model): 0.0004571155244989825
PS C:\Users\alkha\OneDrive\Documents\cyber security detection model>
```

TESTING AFTER 6/27 : TRAINING SPLIT 80% TRAIN , 10% VALIDATION , 10% TEST , HYPERPARAMETER TUNING RESULTS OPTIMIZER: ADAM

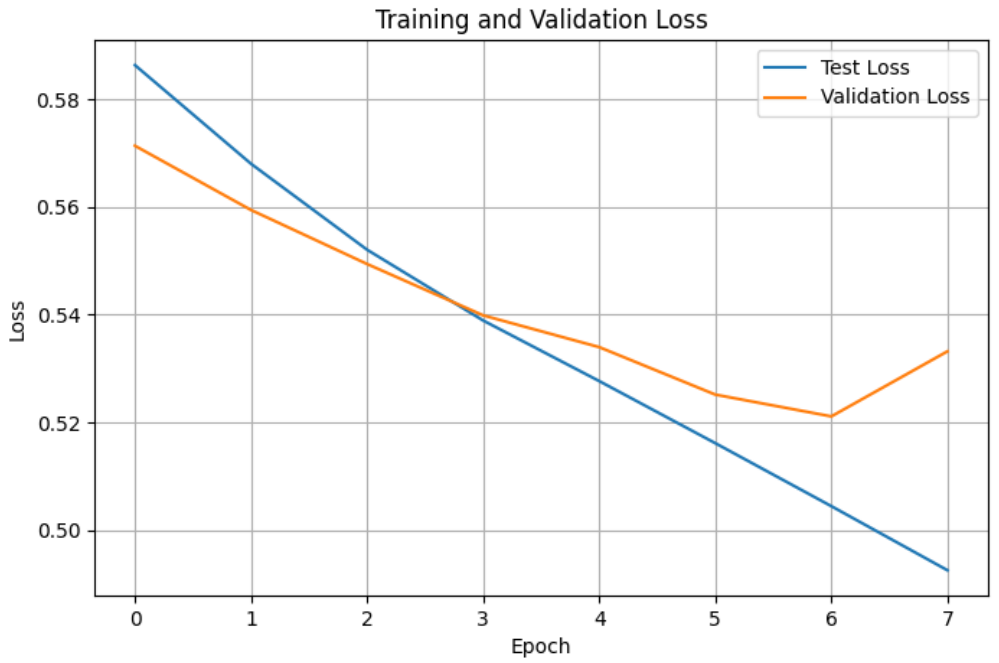



```
Lowest False Alarm Rate from top models: 0.0847
Input dim: 122
Model: "sequential_1"
```

Layer (type)	Output Shape	Param #
dense_4 (Dense)	(None, 160)	19,680
dense_5 (Dense)	(None, 140)	22,540
dense_6 (Dense)	(None, 60)	8,460
dense_7 (Dense)	(None, 1)	61

```
 Total params: 152,225 (594.63 KB)
 Trainable params: 50,741 (198.21 KB)
 Non-trainable params: 0 (0.00 B)
 Optimizer params: 101,484 (396.43 KB)
16/16 ━━━━━━━━━━━━━━━━━━━━ 0s 1ms/step
16/16 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.9450 - loss: 0.2215
Test Loss: 0.21948130428791046
Final Validation Loss: 0.3485465943813324
Test Accuracy: 0.953722357749939
Highest Difference (max absolute error): 0.8809534457314073
False Negatives: 13
False Positives: 10
Precision: 0.973404255319149
Recall: 0.9656992084432717
F1 Score: 0.9695364238410596
False Alarm Rate: 0.0847457627118644

Best Hyperparameters Found:
Best units: 160, 140, 60
Best batch size: 64
Best epochs: 140
Best learning rate (lowest FAR model): 0.002051386718289359
```

TESTING AFTER 6/27 : TRAINING SPLIT 80% TRAIN , 10% VALIDATION , 10% TEST , HYPERPARAMETER TUNING RESULTS OPTIMIZER: ADAM , ENTIRE DATASET




```
Lowest False Alarm Rate from top models: 0.3293
Input dim: 122
Model: "sequential_1"
```

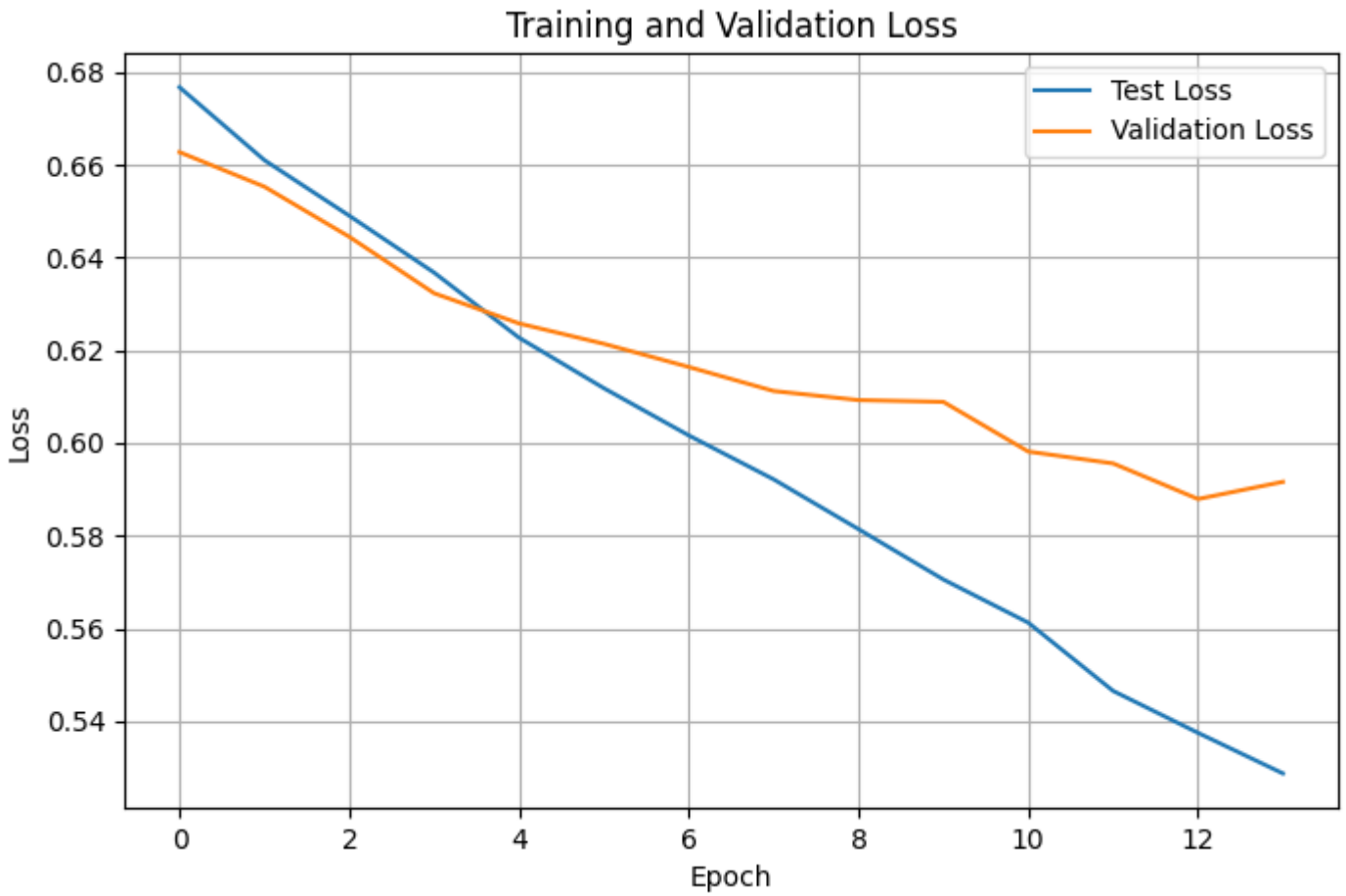
Layer (type)	Output Shape	Param #
dense_4 (Dense)	(None, 60)	7,380
dense_5 (Dense)	(None, 180)	10,980
dense_6 (Dense)	(None, 180)	32,580
dense_7 (Dense)	(None, 1)	181

```
 Total params: 153,365 (599.09 KB)
 Trainable params: 51,121 (199.69 KB)
 Non-trainable params: 0 (0.00 B)
 Optimizer params: 102,244 (399.39 KB)
245/245 ━━━━━━━━━━━━━━━━━━━━ 0s 543us/step
245/245 ━━━━━━━━━━━━━━━━━━━━ 0s 748us/step - accuracy: 0.8204 - loss: 0.5818
Test Loss: 0.6317077279090881
Final Validation Loss: 0.5332104563713074
Test Accuracy: 0.8120694160461426
Highest Difference (max absolute error): 0.5418831765417038
False Negatives: 712
False Positives: 761
Precision: 0.8635222381635581
Recall: 0.871177854170436
F1 Score: 0.8673331532018373
False Alarm Rate: 0.32929467762873216

Best Hyperparameters Found:
Best units: 60, 180, 180
Best batch size: 128
Best epochs: 180
Best learning rate (lowest FAR model): 0.0005935157243286463
```

```
Lowest False Alarm Rate from top models: 0.1415
Input dim: 122
Model: "sequential_1"
```

Layer (type)	Output Shape	Param #
dense_4 (Dense)	(None, 60)	7,380
dense_5 (Dense)	(None, 180)	10,980
dense_6 (Dense)	(None, 180)	32,580
dense_7 (Dense)	(None, 1)	181

```
 Total params: 153,365 (599.09 KB)
 Trainable params: 51,121 (199.69 KB)
 Non-trainable params: 0 (0.00 B)
 Optimizer params: 102,244 (399.39 KB)
213/213 ━━━━━━━━━━━━━━━━━━━━ 0s 1ms/step
213/213 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.7012 - loss: 0.5857
Test Loss: 0.6229376196861267
Final Validation Loss: 0.5916130542755127
Test Accuracy: 0.6927366256713867
Highest Difference (max absolute error): 0.38908106618201244
False Negatives: 1617
False Positives: 477
Precision: 0.793058568329718
Recall: 0.5306240928882439
F1 Score: 0.6358260869565218
False Alarm Rate: 0.14154302670623145

Best Hyperparameters Found:
Best units: 60, 180, 180
Best batch size: 128
Best epochs: 3350
Best learning rate (lowest FAR model): 0.0037228446291984307
Figure(800x500)
```



```
Average over 5 trials:
loss: 4.2836 ± 0.3474
accuracy: 0.8458 ± 0.0053
precision: 0.8648 ± 0.0086
recall: 0.8240 ± 0.0101
f1: 0.8438 ± 0.0055
false_alarm_rate: 0.1318 ± 0.0103
Model: "sequential_4"
```

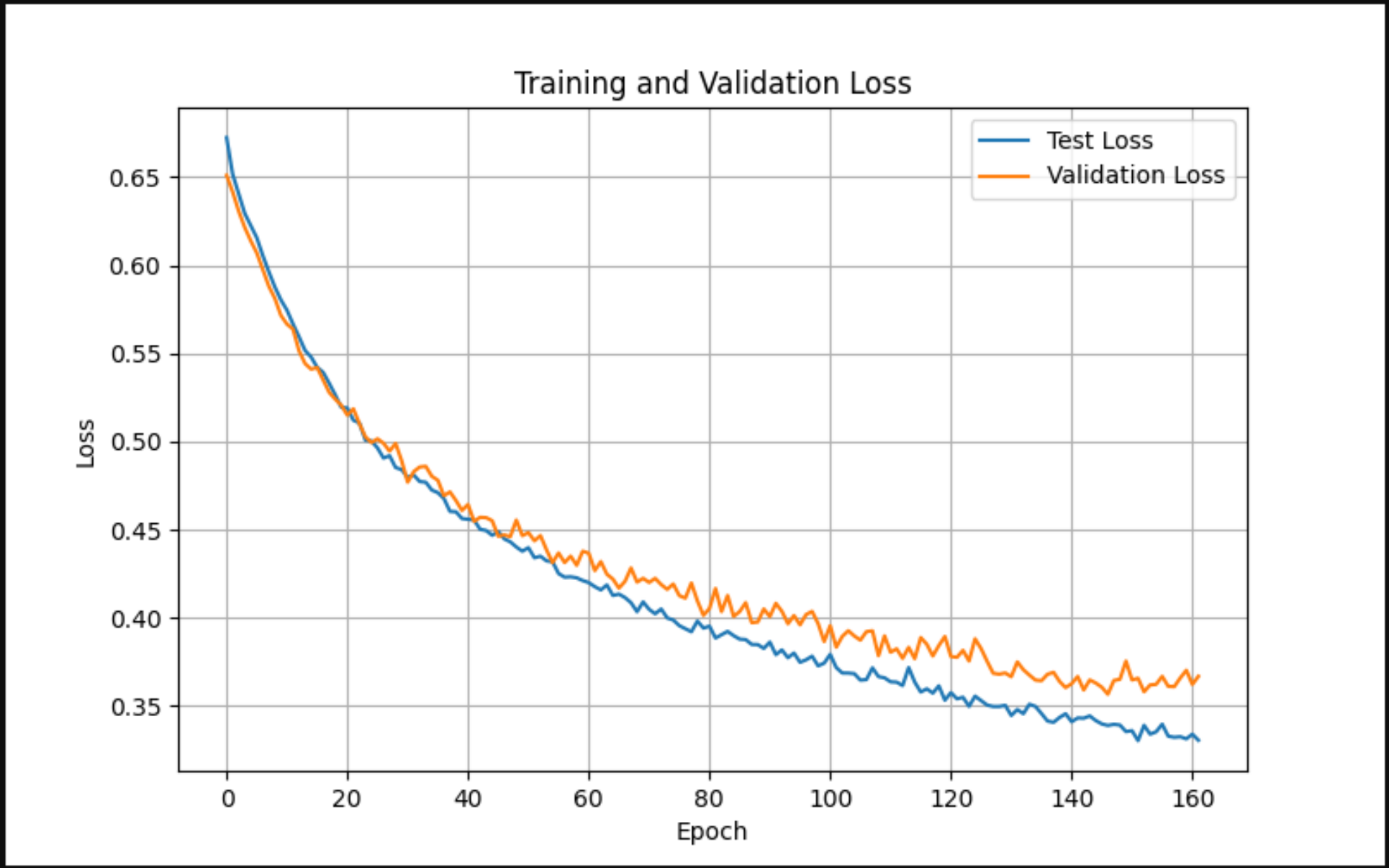
Layer (type)	Output Shape	Param #
conv1d_4 (Conv1D)	(None, 120, 112)	448
flatten_4 (Flatten)	(None, 13440)	0
dense_16 (Dense)	(None, 200)	2,688,200
dense_17 (Dense)	(None, 100)	20,100
dense_18 (Dense)	(None, 150)	15,150
dense_19 (Dense)	(None, 1)	151

```
Total params: 8,172,149 (31.17 MB)
Trainable params: 2,724,049 (10.39 MB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 5,448,100 (20.78 MB)
```

Average over 1 trials:
loss: 0.3055 ± 0.0000
accuracy: 0.9176 ± 0.0000
precision: 0.9443 ± 0.0000
recall: 0.8859 ± 0.0000
f1: 0.9141 ± 0.0000
false_alarm_rate: 0.0513 ± 0.0000
Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 121, 80)	240
flatten (Flatten)	(None, 9680)	0
dropout (Dropout)	(None, 9680)	0
dense (Dense)	(None, 250)	2,420,250
dropout_1 (Dropout)	(None, 250)	0
dense_1 (Dense)	(None, 200)	50,200
dropout_2 (Dropout)	(None, 200)	0
dense_2 (Dense)	(None, 100)	20,100
dense_3 (Dense)	(None, 1)	101

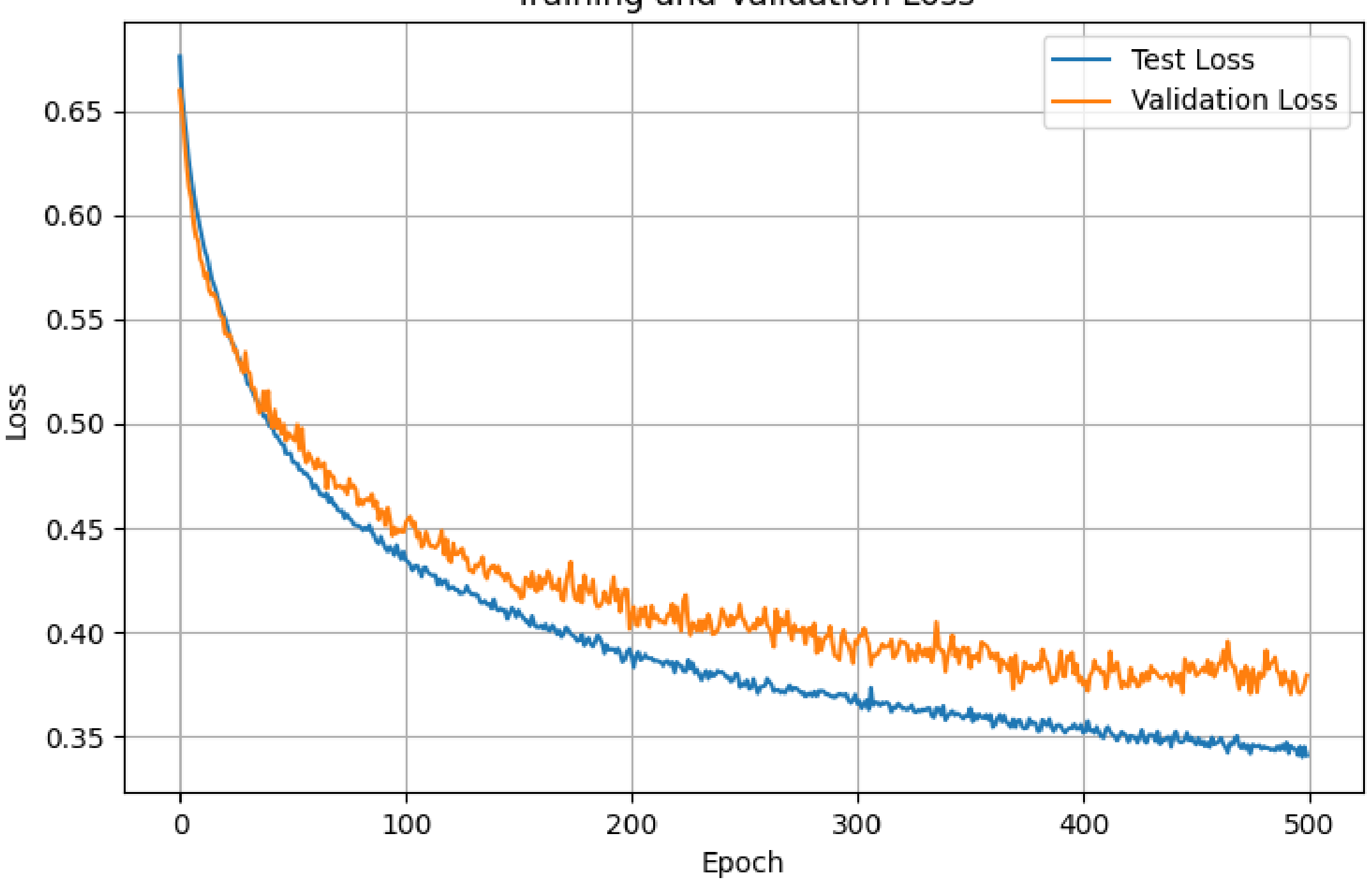
Total params: 7,472,675 (28.51 MB)
Trainable params: 2,490,891 (9.50 MB)
Non-trainable params: 0 (0.00 B)
Optimizer params: 4,981,784 (19.00 MB)


Training and Validation Loss
Test Loss
Validation Loss
Loss
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0
20
40
60
80
100
120
140
160
Epoch

loss: 0.3743 ± 0.0000
accuracy: 0.8444 ± 0.0000
precision: 0.9014 ± 0.0000
recall: 0.7672 ± 0.0000
f1: 0.8289 ± 0.0000
false_alarm_rate: 0.0811 ± 0.0000
highest_diff: 0.6861 ± 0.0000
Model: "sequential"

Layer (type)	Output Shape	Param #
conv1d (Conv1D)	(None, 121, 48)	144
flatten (Flatten)	(None, 5808)	0
dropout (Dropout)	(None, 5808)	0
dense (Dense)	(None, 200)	1,161,800
dropout_1 (Dropout)	(None, 200)	0
dense_1 (Dense)	(None, 150)	30,150
dropout_2 (Dropout)	(None, 150)	0
dense_2 (Dense)	(None, 50)	7,550
dense_3 (Dense)	(None, 1)	51

Total params: 3,599,087 (13.73 MB)


Training and Validation Loss
Test Loss
Validation Loss
Loss
0.65
0.60
0.55
0.50
0.45
0.40
0.35
0
100
200
300
400
500
Epoch

Federated Learning Best Result So Far (5 datasets or 5 clients)

Model	Communication rounds	Accuracy	Epoch
CNN	150	.7401	1
CNN	200	.7564	1
CNN	200	.7840	1
CNN	200	.8802	5
CNN	200	.9017	10

Future Work (Federated Learning)

- Hyperparameter Tuning
  - Refine learning rates, batch sizes, and optimizers to improve model convergence.

- Optimization of Communication Strategies
  - Increase number of communication rounds.
  - Experiment with adaptive round scheduling and communication-efficient algorithms.

- Client Sampling & Aggregation Enhancements
  - Explore smarter client selection methods.
  - Investigate weighted aggregation based on data quality or client representativeness.

- Improving Global Accuracy
  - Aim to match or exceed centralized CNN performance through iterative tuning.

- Security & Robustness Testing
  - Simulate evasion attacks to evaluate model resilience.
  - Focus on adversarial robustness for cybersecurity applications.