deti universidade de aveiro
departamento de eletrónica, telecomunicações e informática

Data Exfiltration via WhatsApp

| Hugo Moinheiro (84931) | Rafael Amorim (98197) |

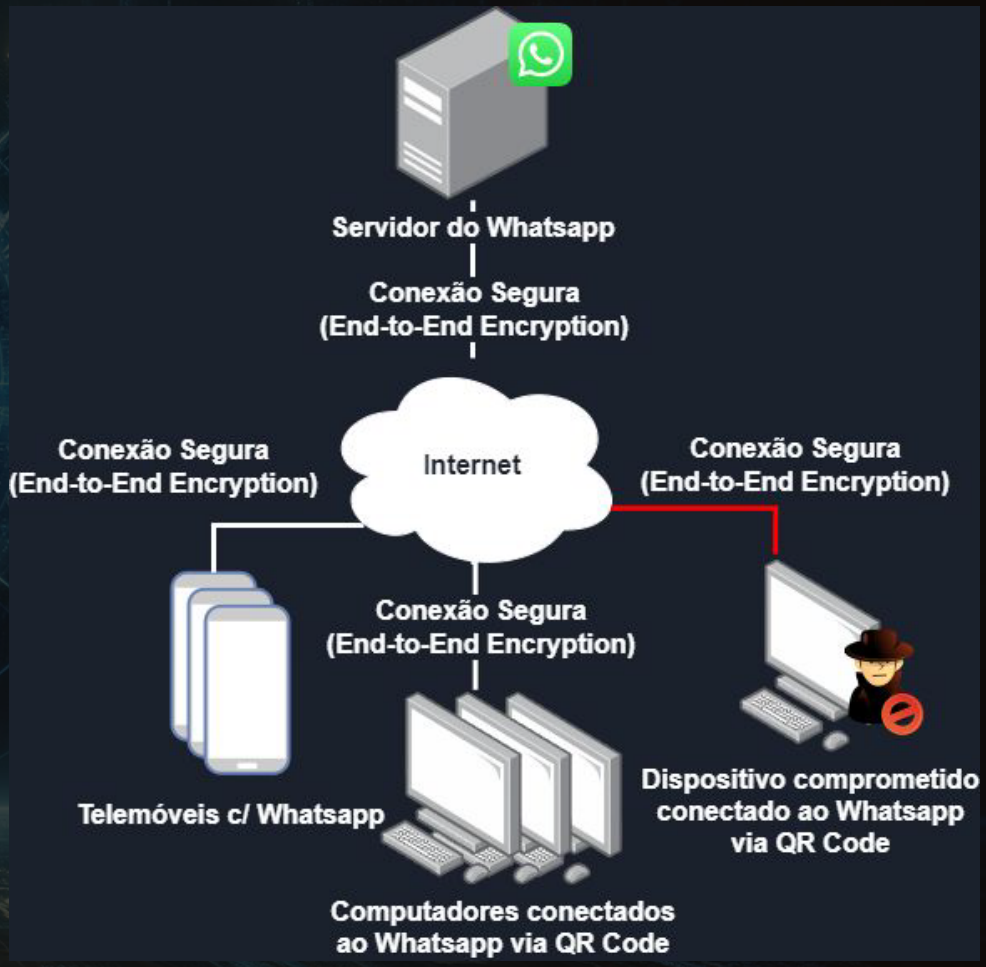
Reformulating 1° Presentation

We considered:

- Using WhatsApp daily is a common practice
- The attacker gained access to a device and uses WhatsApp to exfiltrate data

Suspicious activities:

- Sending messages more frequently and with larger amounts of data than a normal user would.



Good vs Bad Behaviors

```python
def send_message(chat_id, text):
    """
    Send message to chat_id.
    :param chat_id: Phone number + "@c.us" suffix
    :param text: Message for the recipient
    """
    # Send a text message via WhatsApp HTTP API
    response = requests.post(
        "http://localhost:3000/api/sendText",
        json={
            "chatId": chat_id,
            "text": text,
            "session": "default",
        },
    )
    response.raise_for_status()
```

Good Behavior:

- Data for models derived from actual navigation.

Integration with an API after account association.
Two bot scripts: basic and advanced.
Bad Behavior:

- Dumb has a sequence of time and uniform random chars size
- Advanced bot mimics human behavior by mixing short, quick messages with longer, more time-consuming ones. It prioritizes short messages for realistic interaction.

Basic Bot

- Every 10 messages pause for 15 seconds
- 60% chance of messages with 50-100 characters
- 40% chance of messages with 101-500 characters
- Every message pause for 1-3 seconds
- Messages of random strings

Advanced Bot

- Every 10 messages pause for 10-20 seconds
- 40% chance of messages with 10-29 characters and pause for X seconds
- 35% chance of messages with 30-99 characters and pause for Y seconds
- 25% chance of messages with 100-500 characters and pause for Z seconds
- Messages of random strings

- X = Gaussean with mean 10 and standard deviation 5 truncated in interval [0,20]
- Y = Gaussean with mean 20 and standard deviation 10 truncated in interval [0,40]
- Z = Gaussean with mean 30 and standard deviation 15 truncated in interval [0,60]

Observation Window

Considering periodicity interval: 5 seconds

Sliding Window:

- Width of 20 intervals (1 minute and 40 seconds)
- Slide of 4 intervals (20 seconds)

Input Data

Metrics

For each IP in each capture:

- Number of packets upload
- Number of bytes upload
- Number of packets upload
- Number of bytes upload

For activity and silence

Features

For each Metric:

- Sum
- Percentage
- Maximum/ Minimun
- Average/ Median/ Standard Deviation

Note:

Only counting packets of Facebook networks ('157.240.212.0/24') so percentage is irrelevant

Client IP is 192.168.0.164

Captures

Normal behaviour

- 1h30 of capture with browsing and normal WhatsApp messaging
- 5brsg1h30_lowFreq.pcap

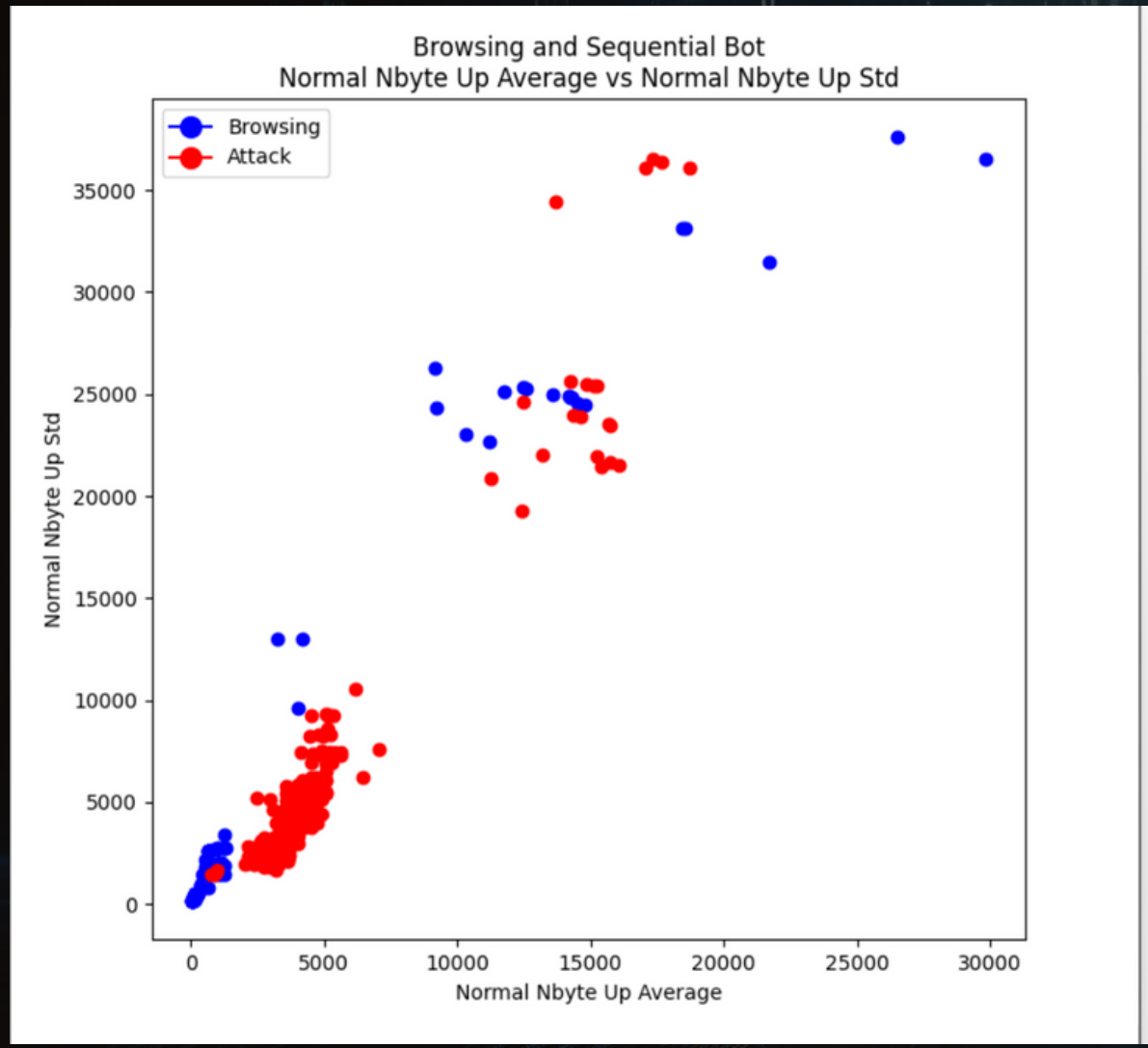
Basic Attack

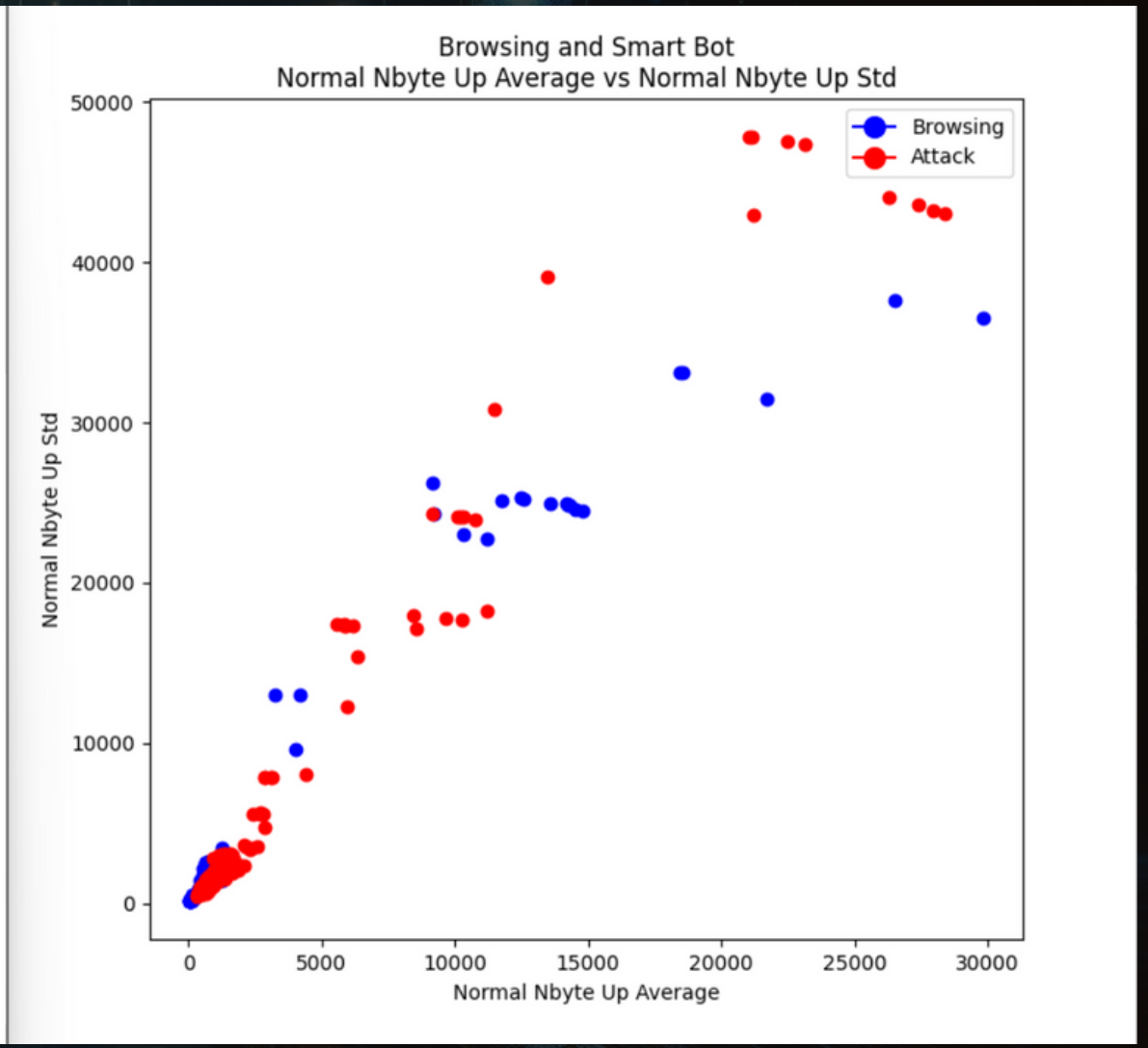
- 1h30 of capture with browsing and normal WhatsApp messaging + basic script
- 4seq1h30.pcap

Advanced Attack

- 1h30 of capture with browsing and normal WhatsApp messaging + advanced script
- 4smart1h30.pcap

Plots


Browsing and Sequential Bot
Normal Nbyte Up Average vs Normal Nbyte Up Std
Browsing
Attack
Normal Nbyte Up Std
Normal Nbyte Up Average
0
5000
10000
15000
20000
25000
30000
35000


Browsing and Smart Bot
Normal Nbyte Up Average vs Normal Nbyte Up Std
Browsing
Attack
Normal Nbyte Up Std
Normal Nbyte Up Average
0
10000
20000
30000
40000
50000
5000
15000
25000

Plots

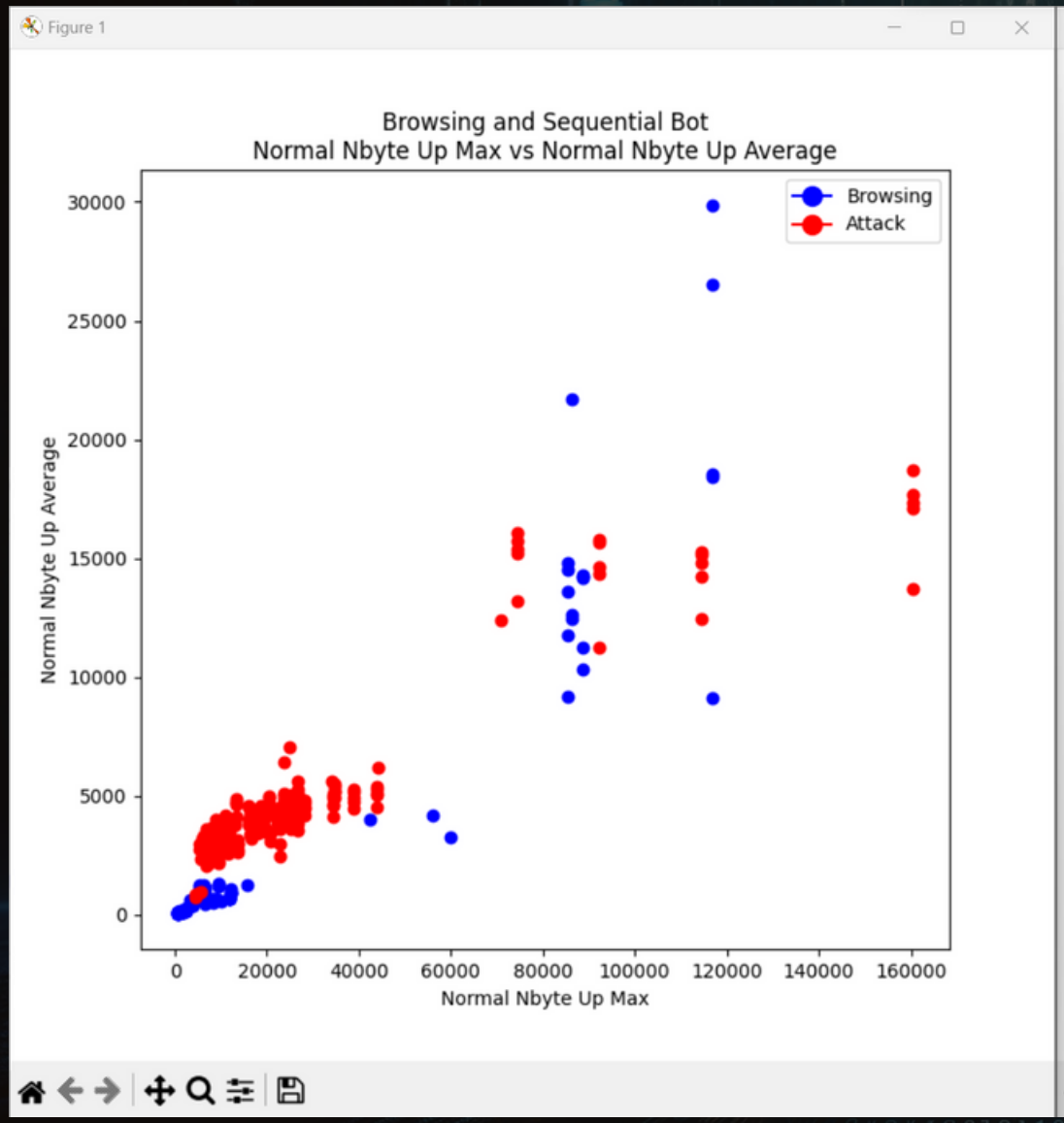


Figure 1
Browsing and Sequential Bot
Normal Nbyte Up Max vs Normal Nbyte Up Average
Browsing
Attack
Normal Nbyte Up Average
Normal Nbyte Up Max
0
5000
10000
15000
20000
25000
30000
0
20000
40000
60000
80000
100000
120000
140000
160000

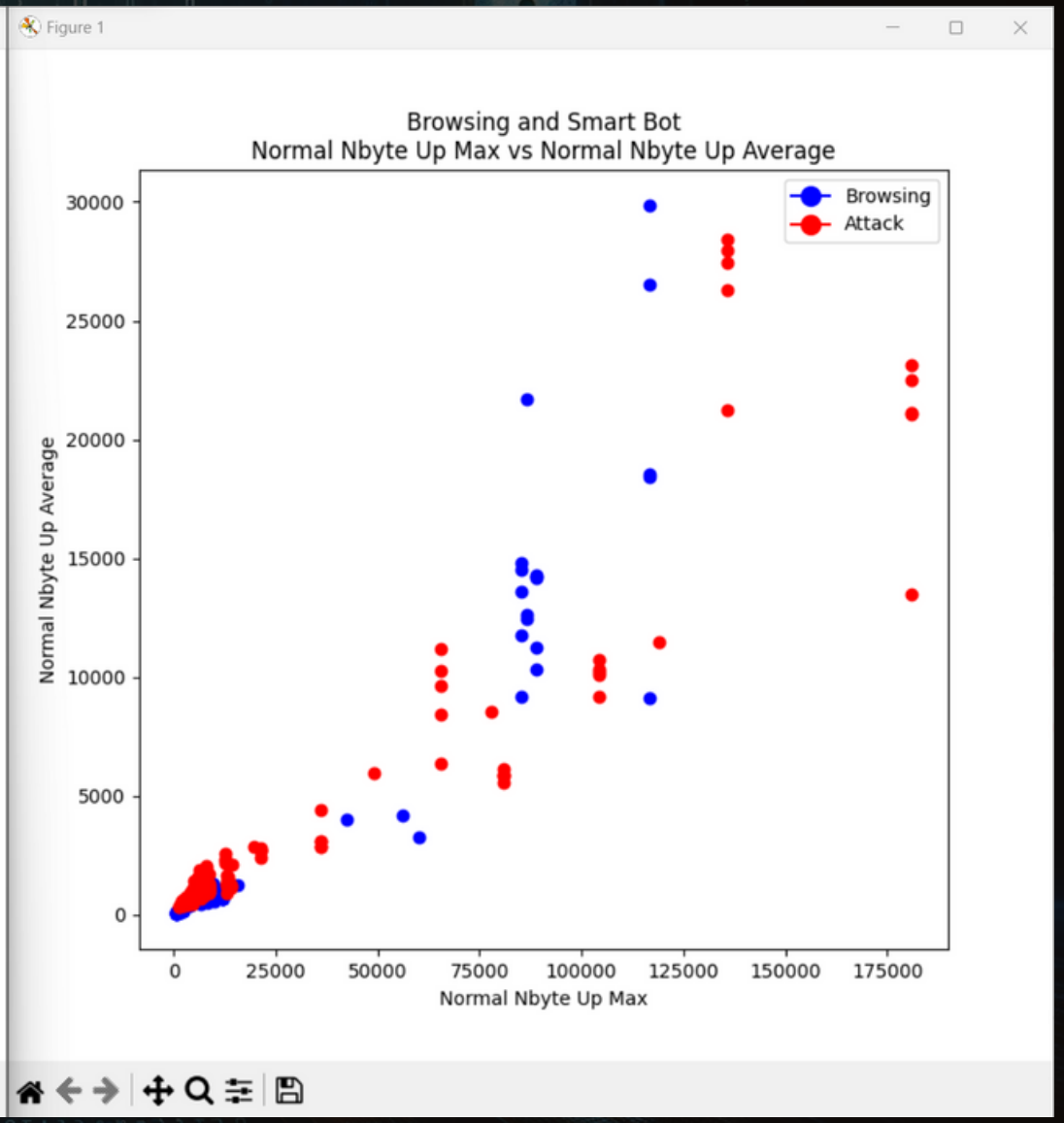


Figure 1
Browsing and Smart Bot
Normal Nbyte Up Max vs Normal Nbyte Up Average
Browsing
Attack
Normal Nbyte Up Average
Normal Nbyte Up Max
0
5000
10000
15000
20000
25000
30000
0
25000
50000
75000
100000
125000
150000
175000

Training & Test Features

First 50%: Train Features
Second 50%: Test Features

SumF				PercentageF				MaxF				MinF				AverageF				MedianF				StdF			
N_Pkt_Upload	N_Bytes_Upload	N_Pkt_Download	N_Bytes_Download	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By	N_Pl	N_By
Train																											
Test																											

Classes Defined

Two classes:

- Client – 0
  - Only one class created to represent human behavior patterns
- Attacker – 1
  - Only one class to detect anomalies

Methods

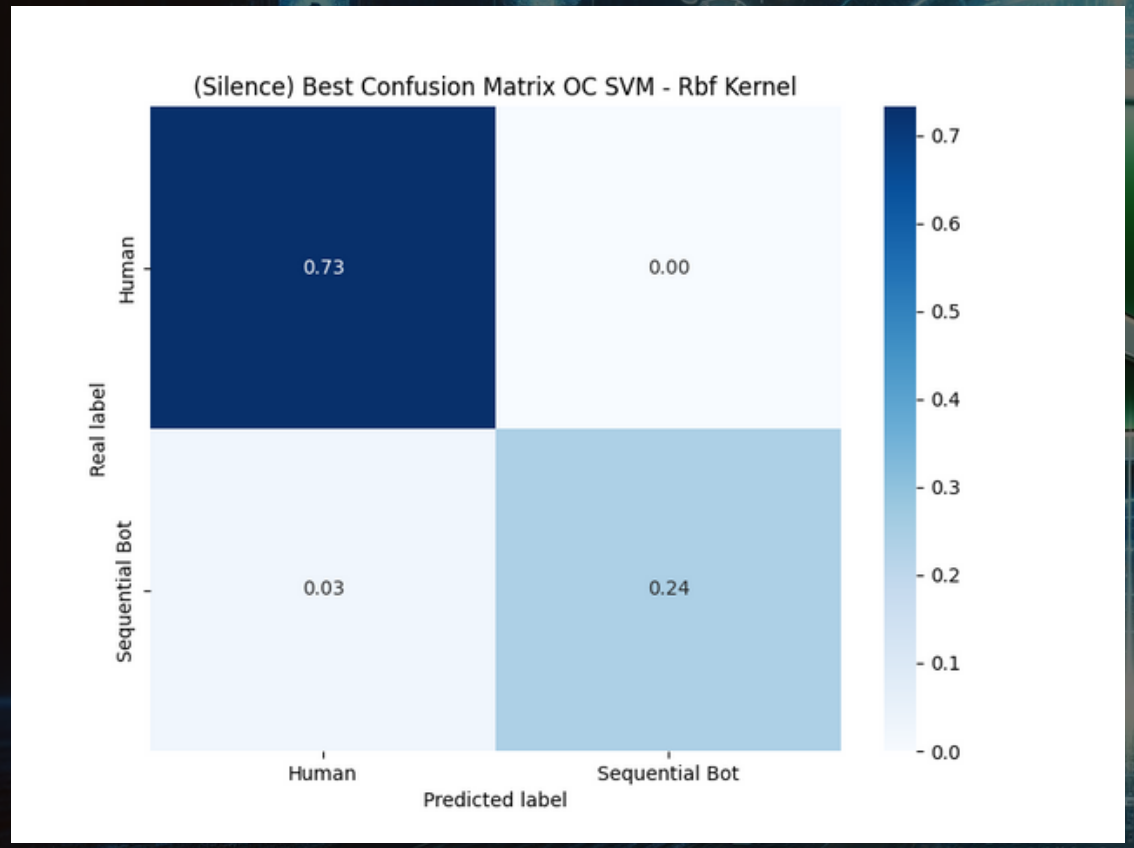
Machine Learning based on:

- One Class SVM with and without PCA features
  - Linear, RBF, Poly
- Local Outlier Factor with and without PCA features
- Isolation Forest with and without PCA features
- Gaussian Mix Model with and without PCA features
- Robust Covariance with and without PCA features

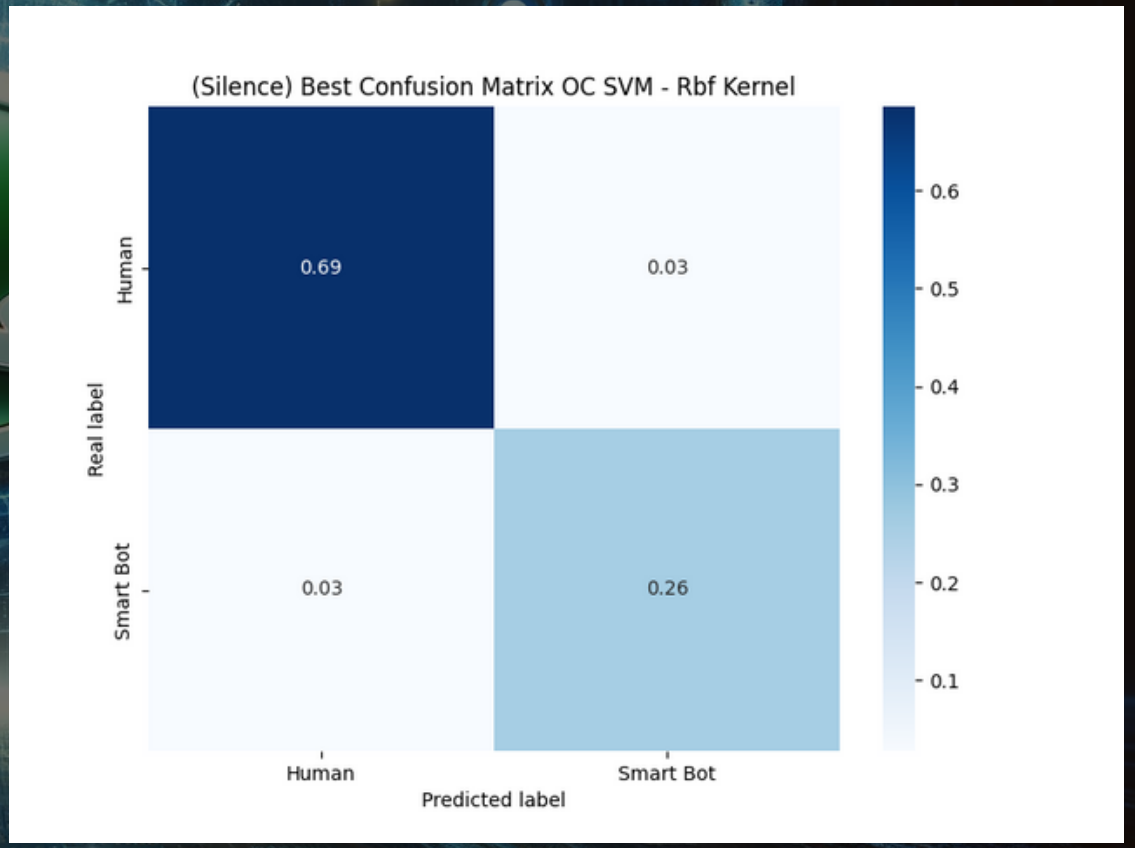
Basic vs Advanced Results

One-Class Support Vector Machine

F1 Score: 98.25



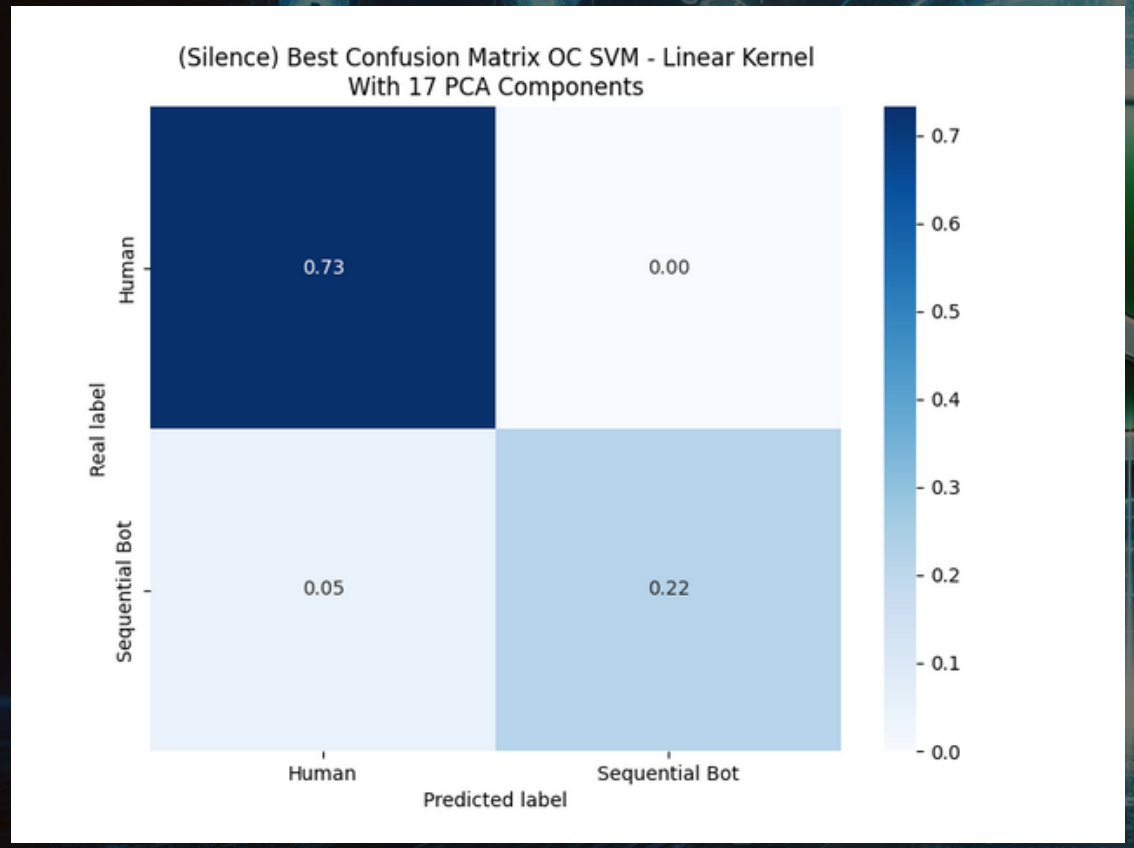
F1 Score: 96.06



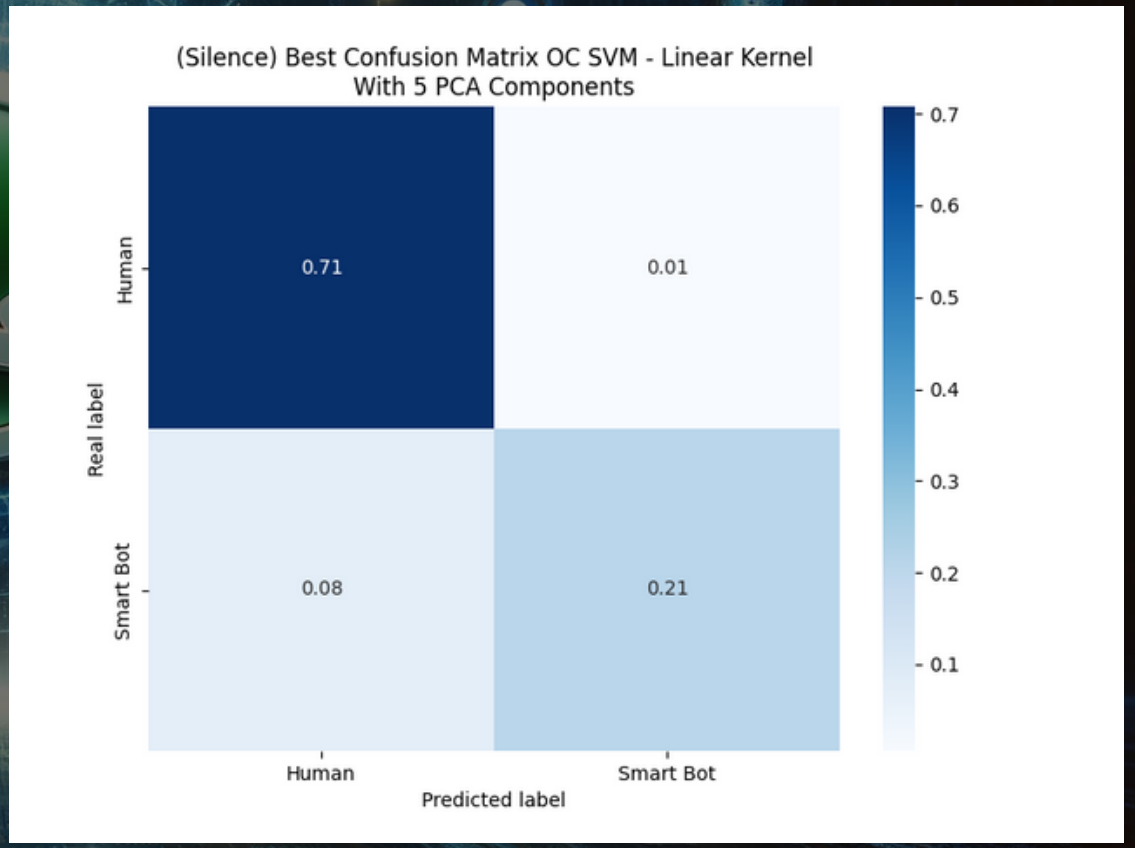
Basic vs Advanced Results

One-Class Support Vector Machine PCA Components

F1 Score: 96.89



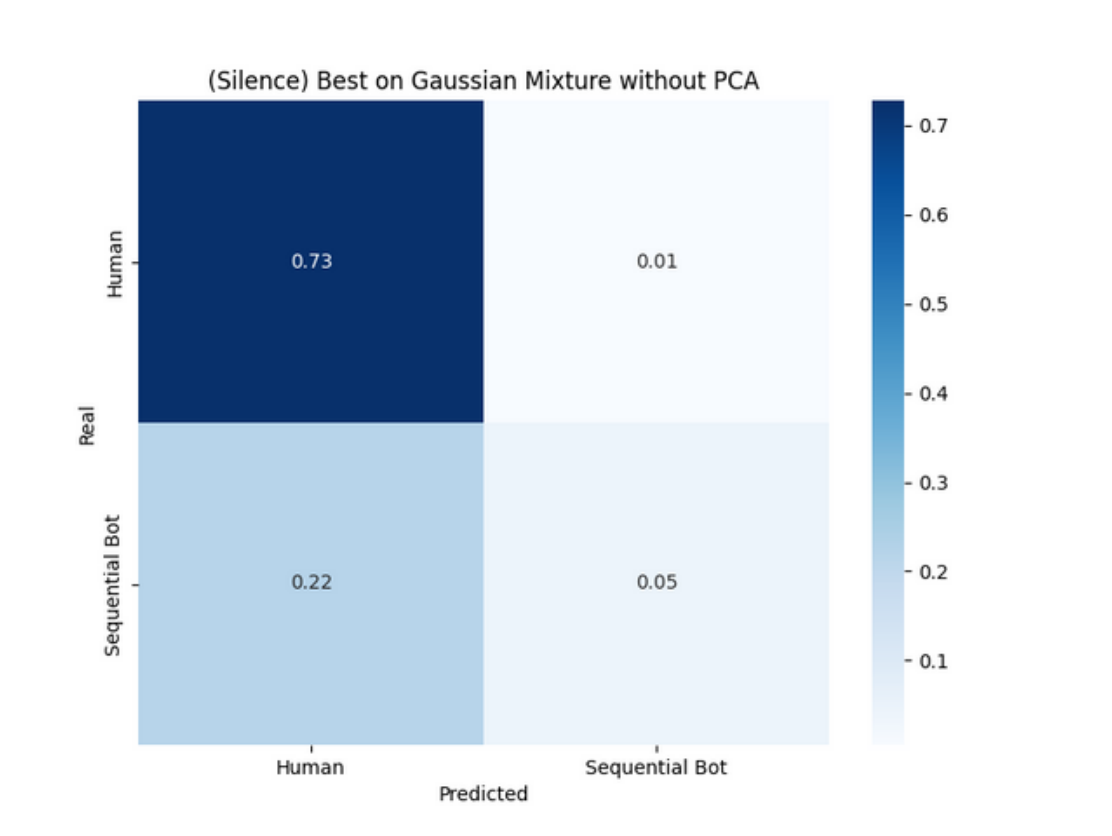
F1 Score: 94.38



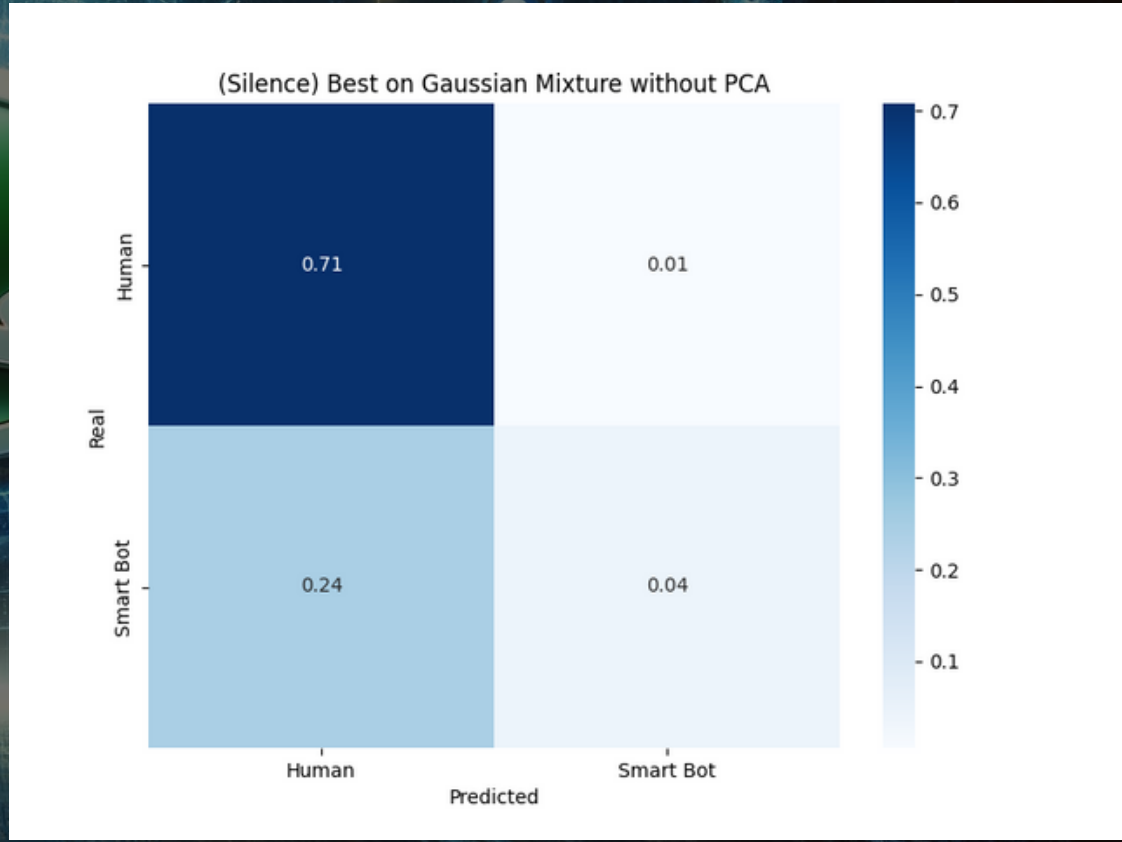
Basic vs Advanced Results

Gaussian Mixture

F1 Score: 86.6



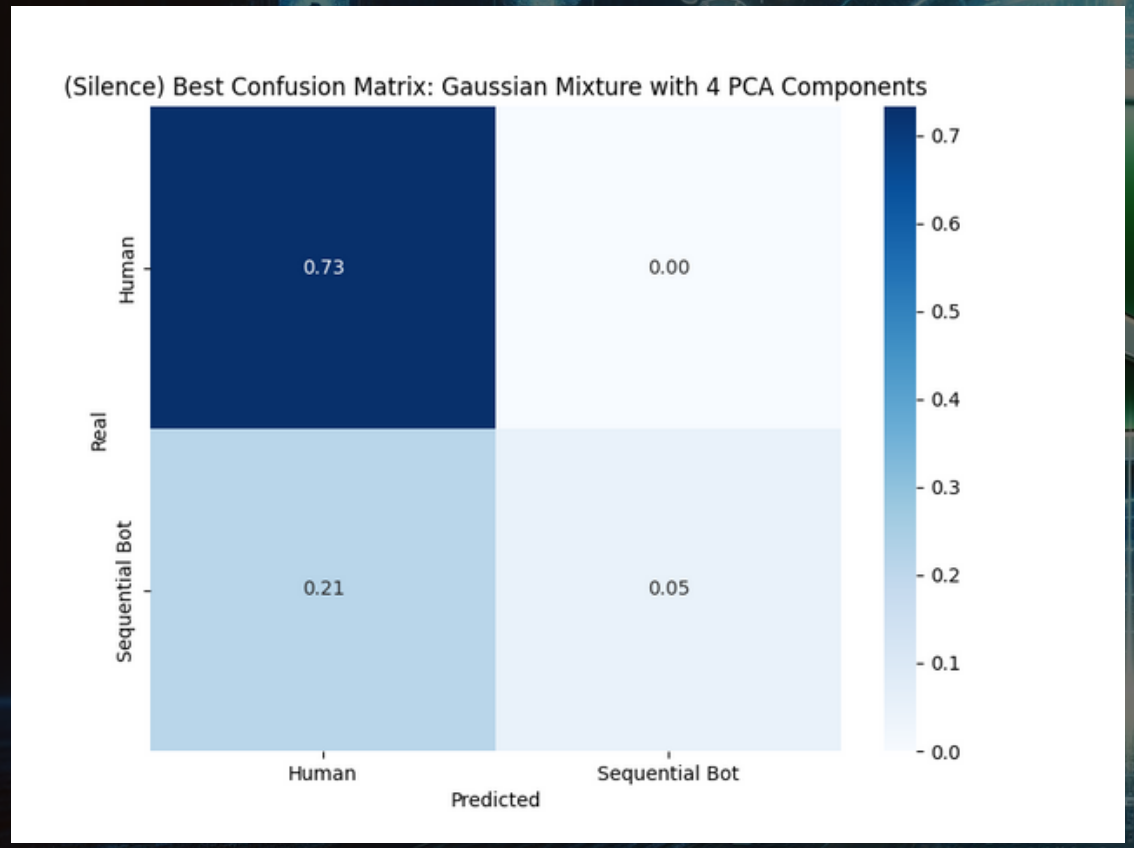
F1 Score: 85.14



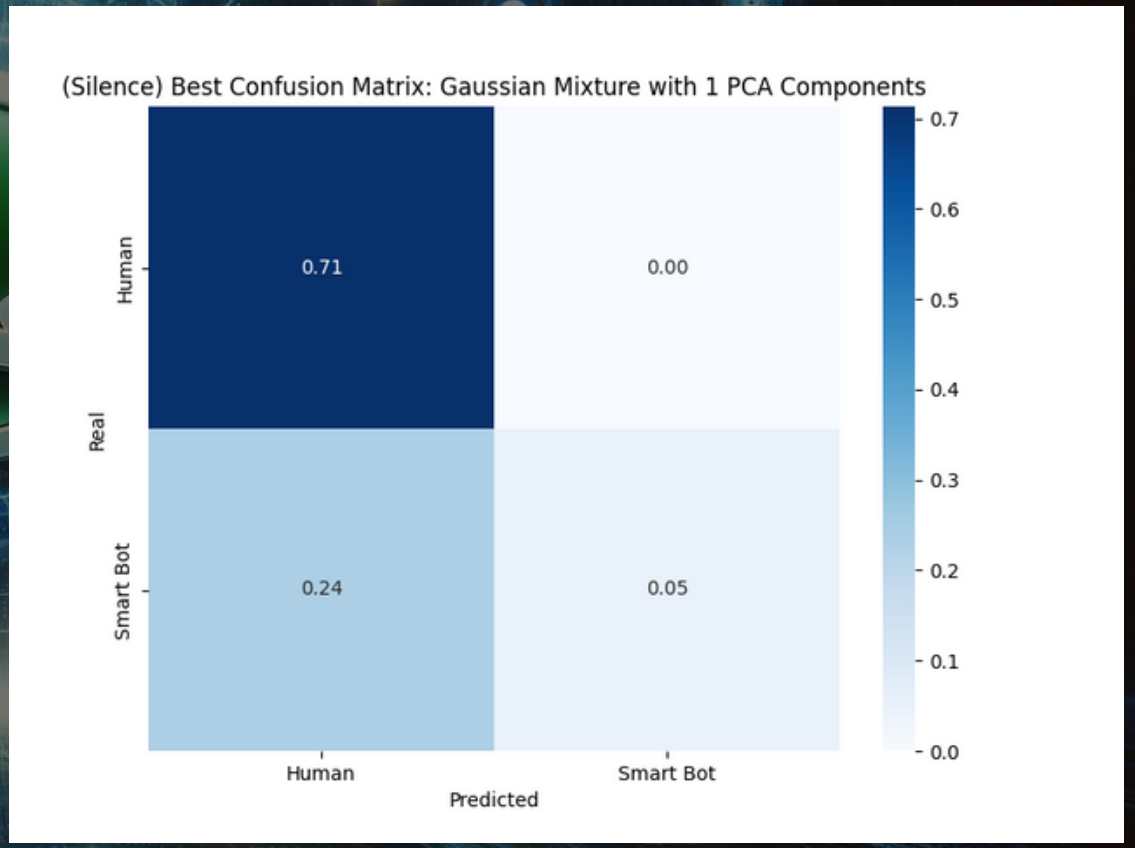
Basic vs Advanced Results

Gaussian Mixture PCA Components

F1 Score: 87.23



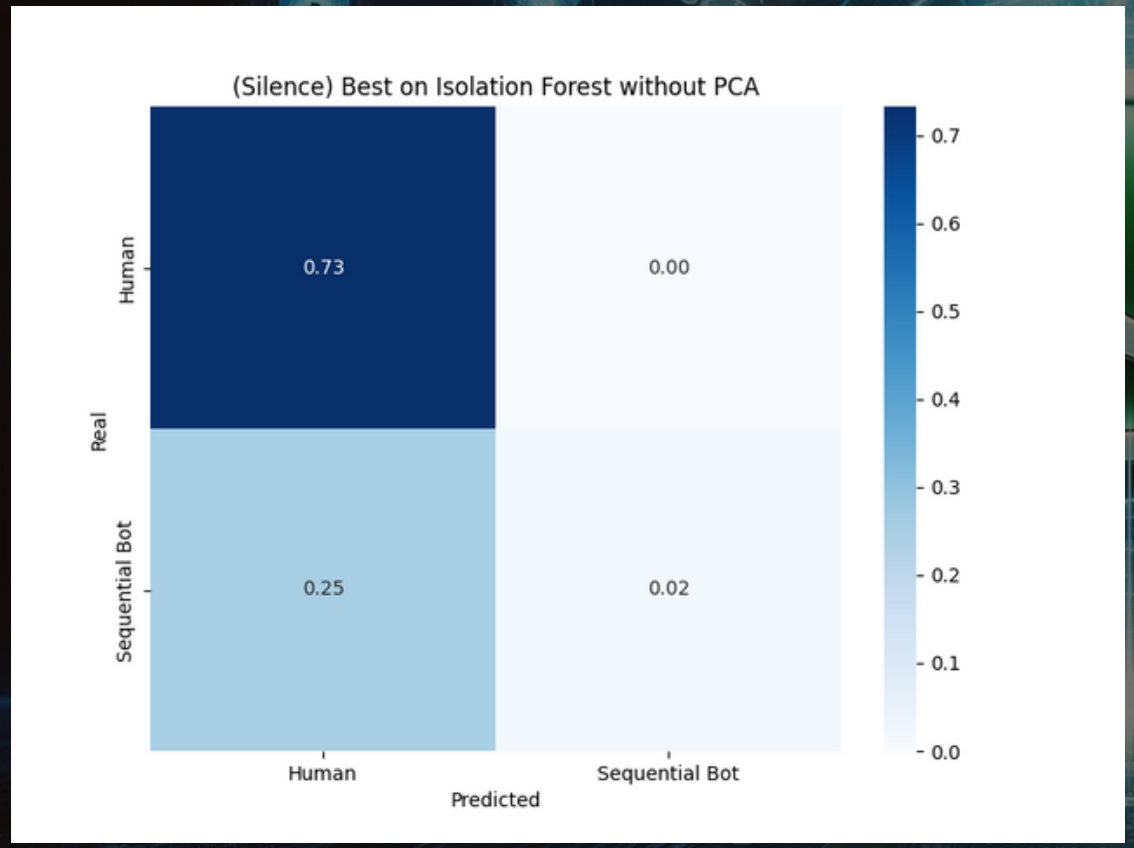
F1 Score: 85.81



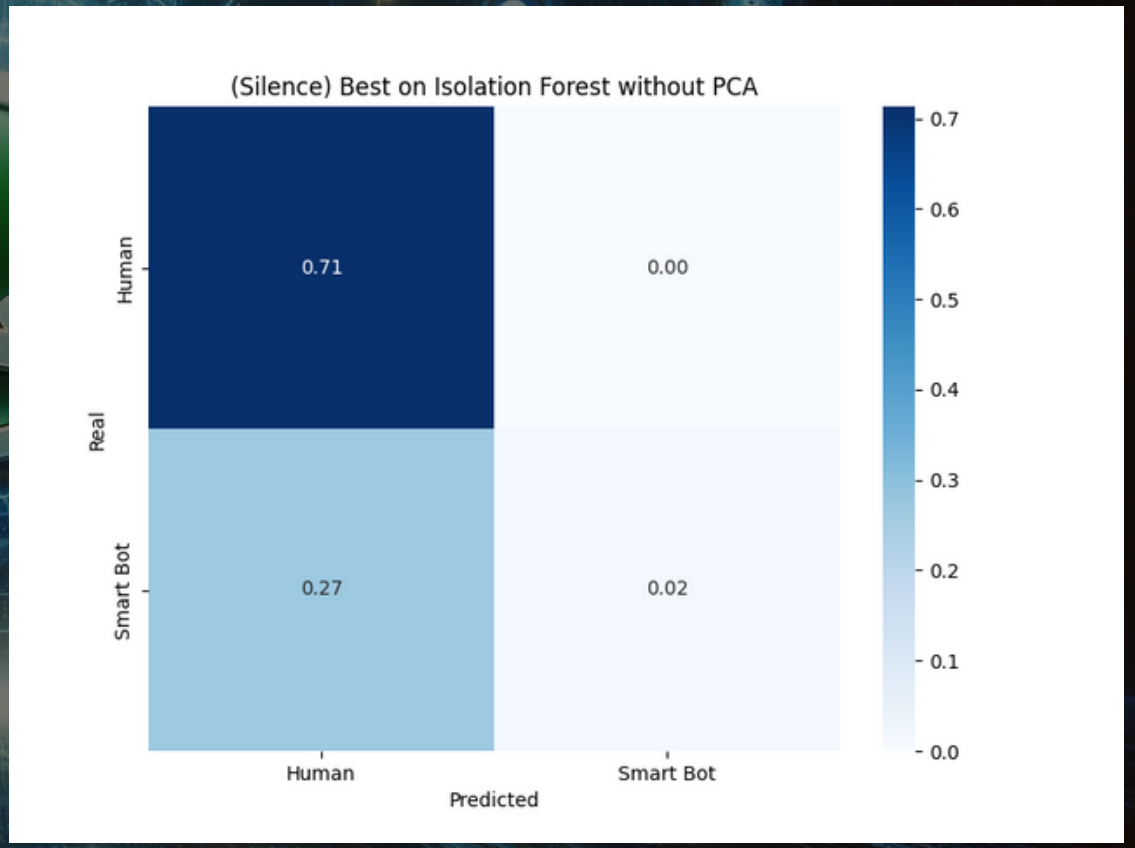
Basic vs Advanced Results

Isolation Forest

F1 Score: 86.15



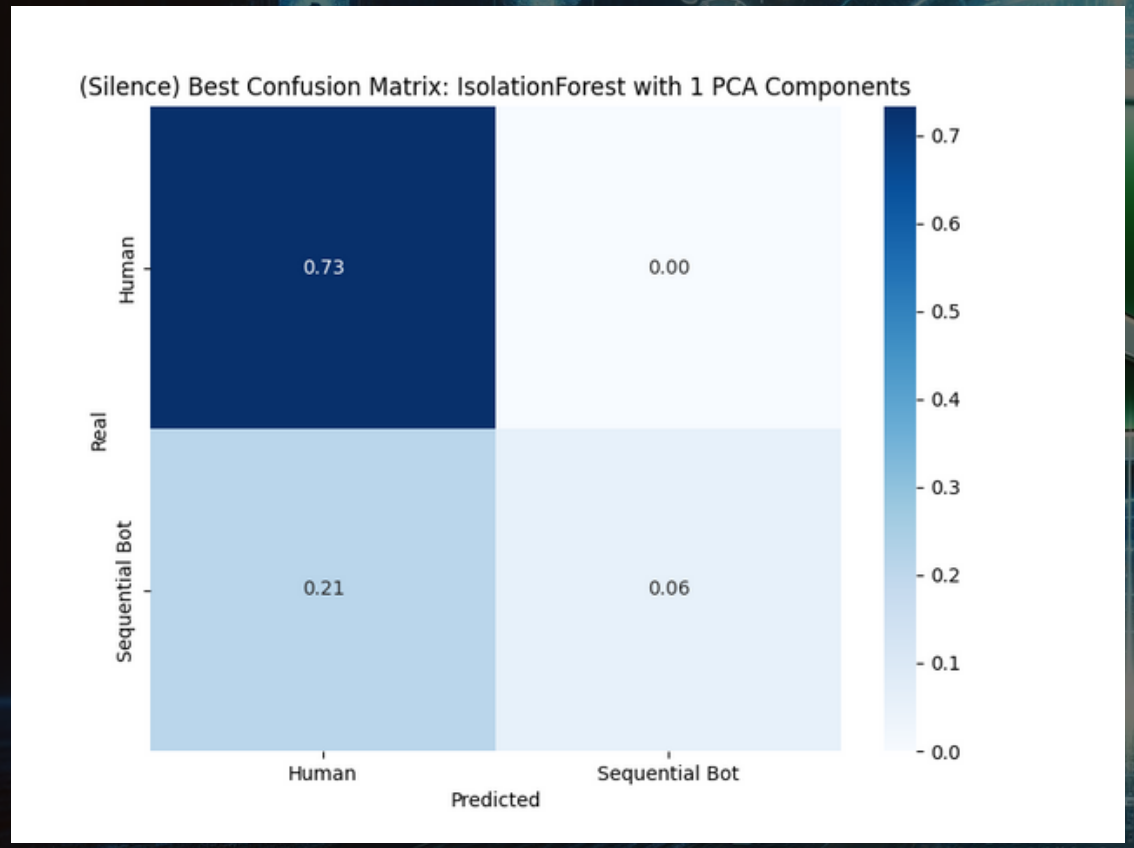
F1 Score: 84.11



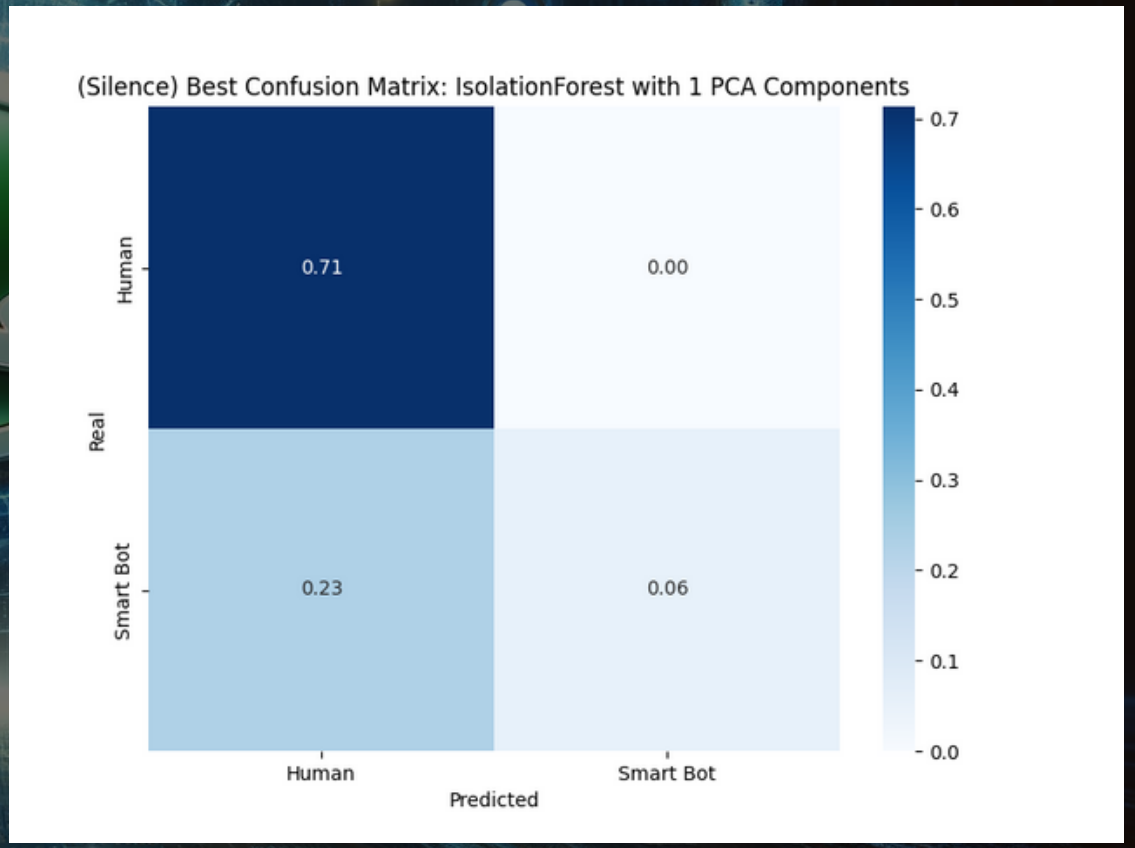
Basic vs Advanced Results

Isolation Forest PCA Components

F1 Score: 87.23



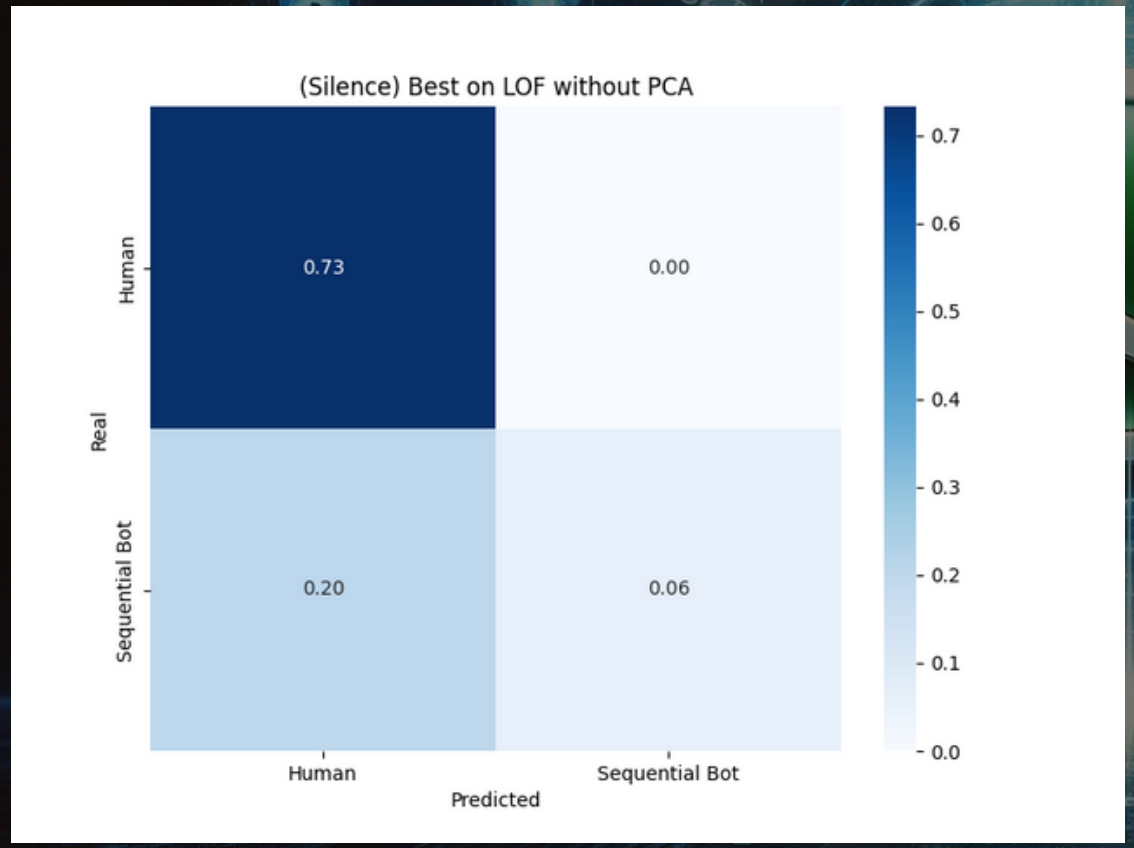
F1 Score: 86.1



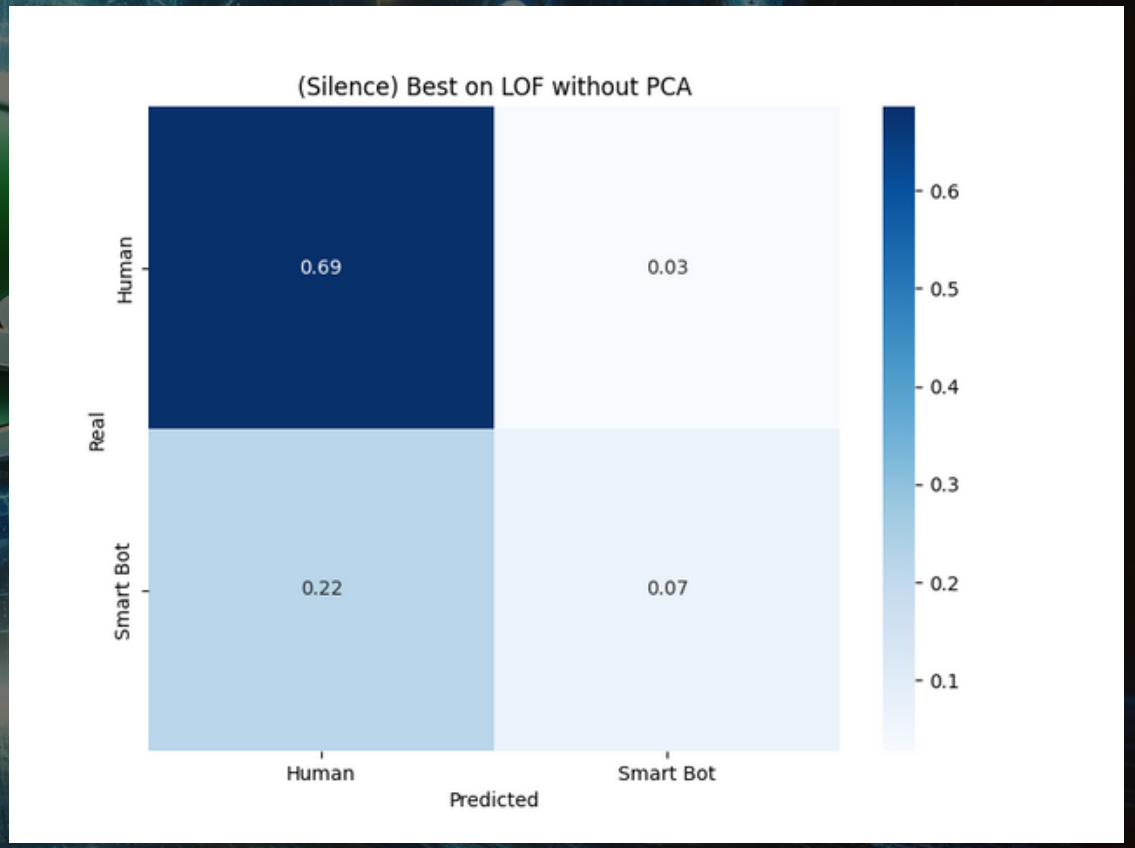
Basic vs Advanced Results

Local Outlier Factor

F1 Score: 87.77



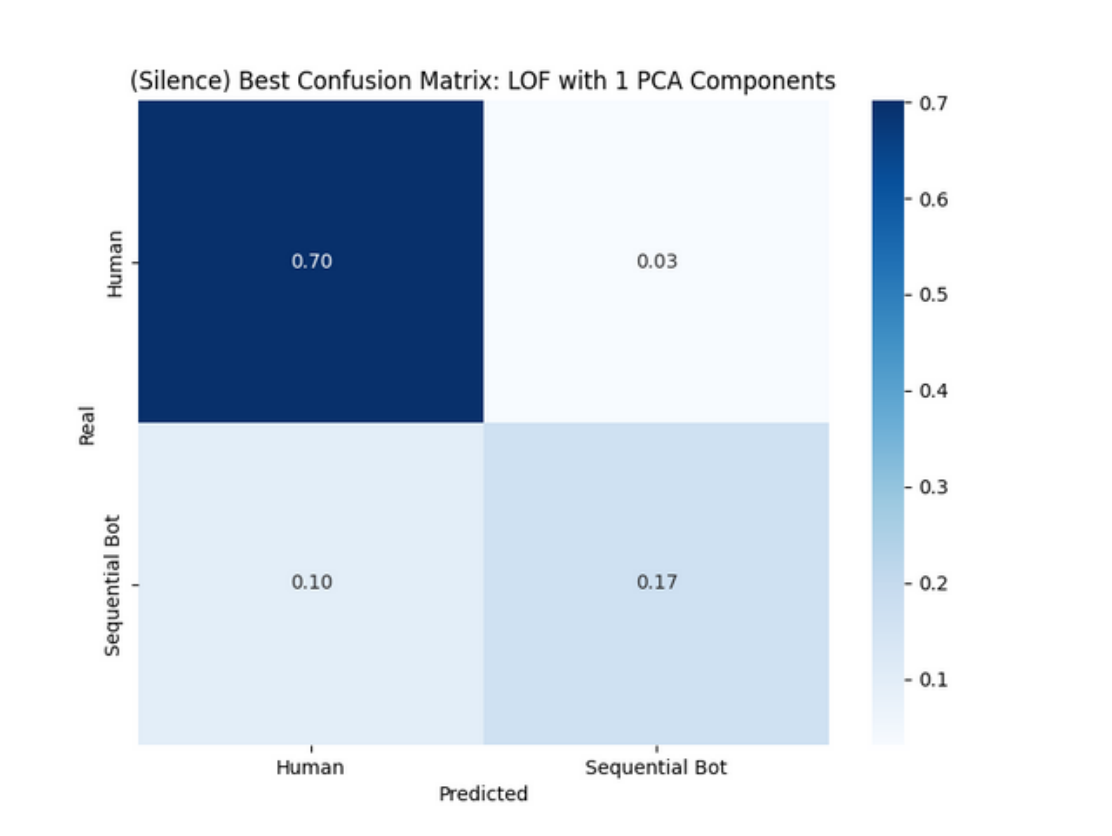
F1 Score: 84.72



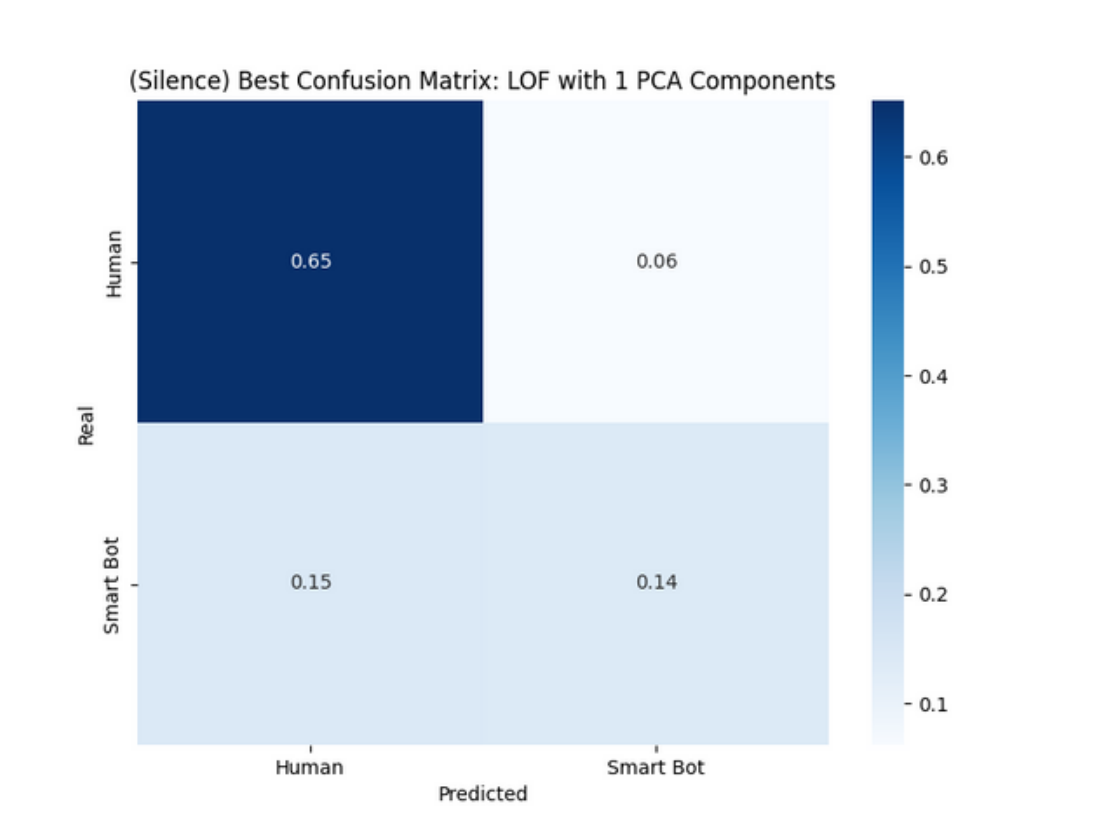
Basic vs Advanced Results

Local Outlier Factor PCA Components

F1 Score: 91.47



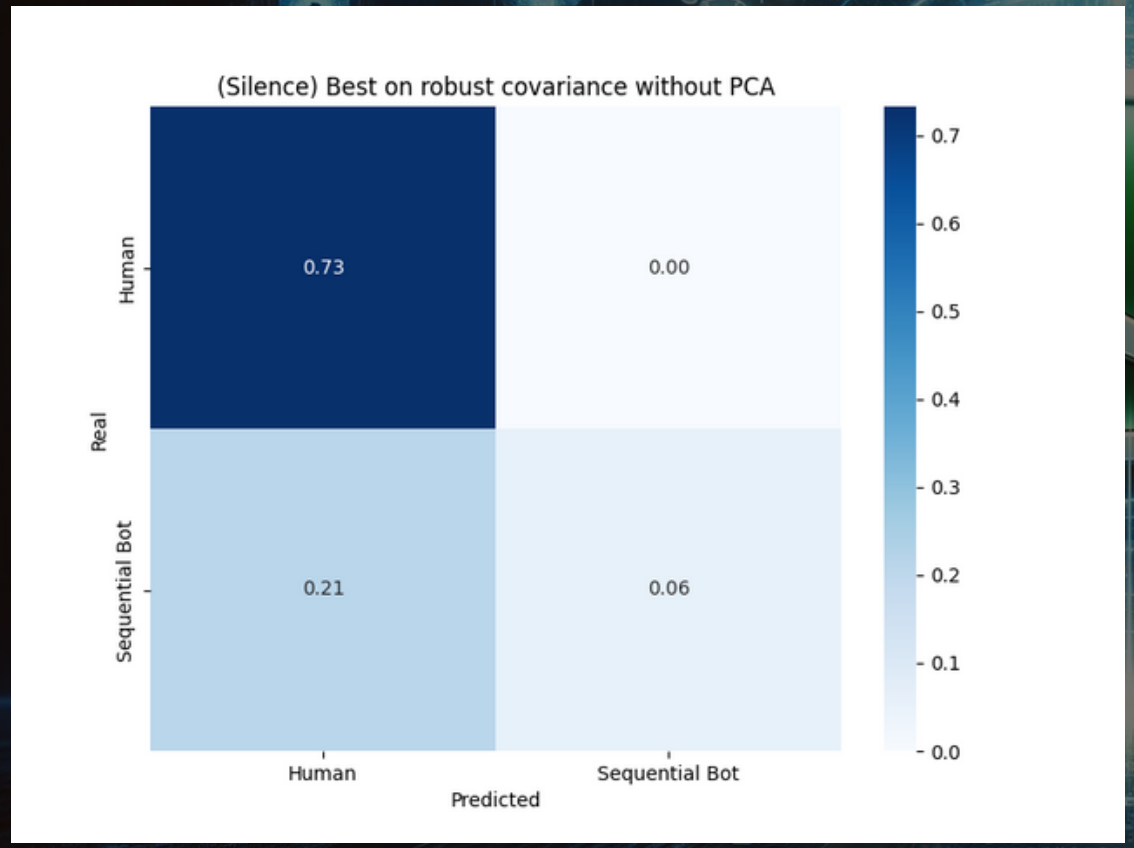
F1 Score: 86.25



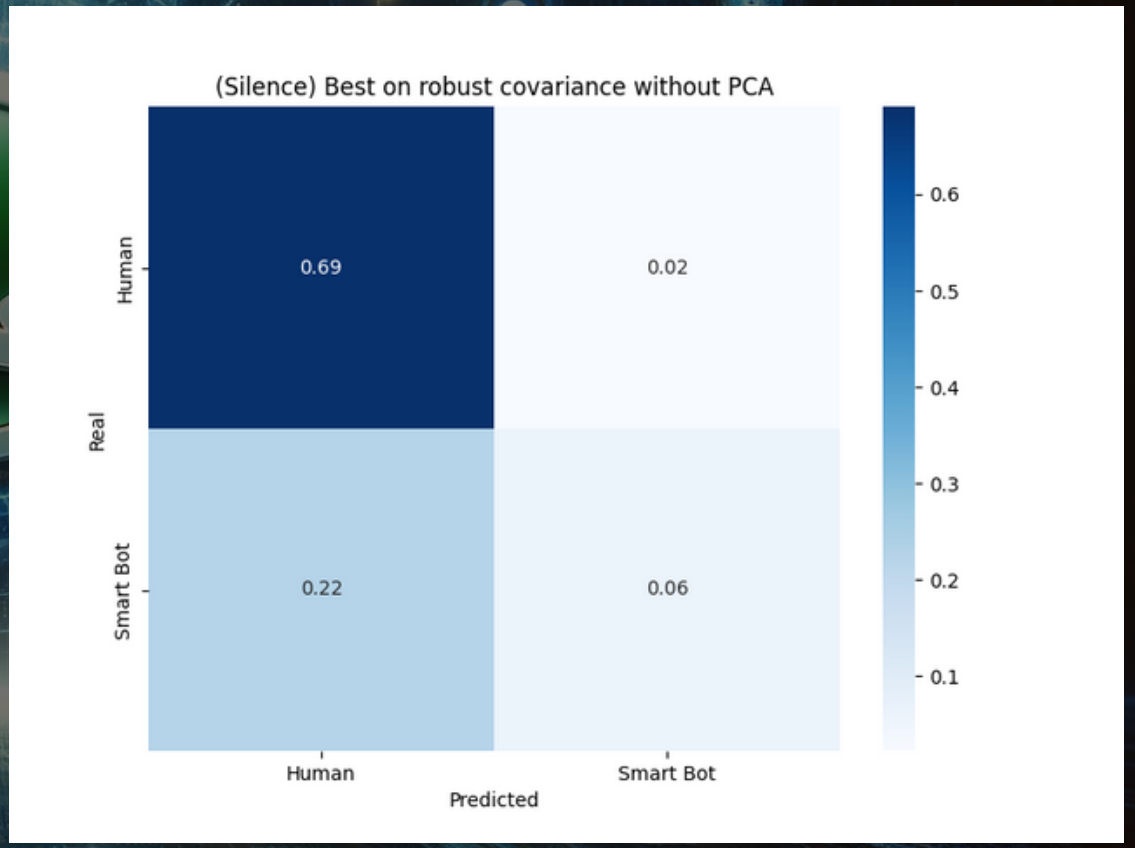
Basic vs Advanced Results

Robust Covariance

F1 Score: 87.5



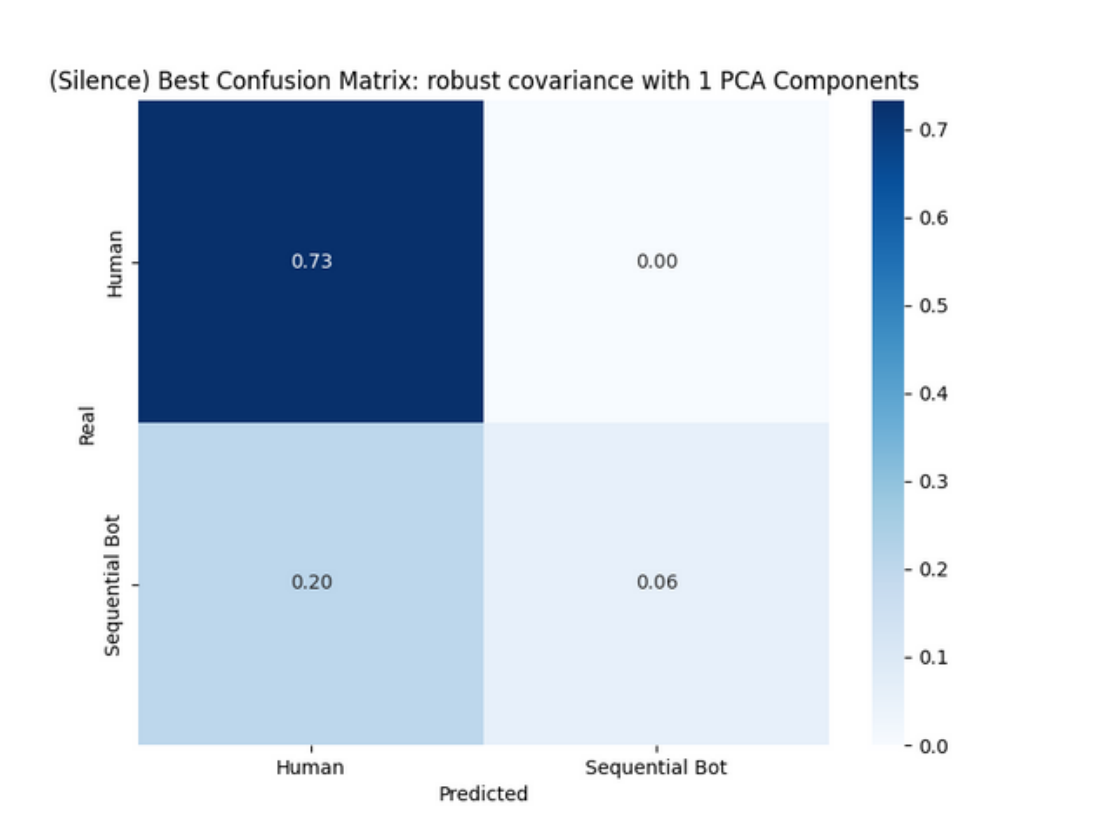
F1 Score: 84.83



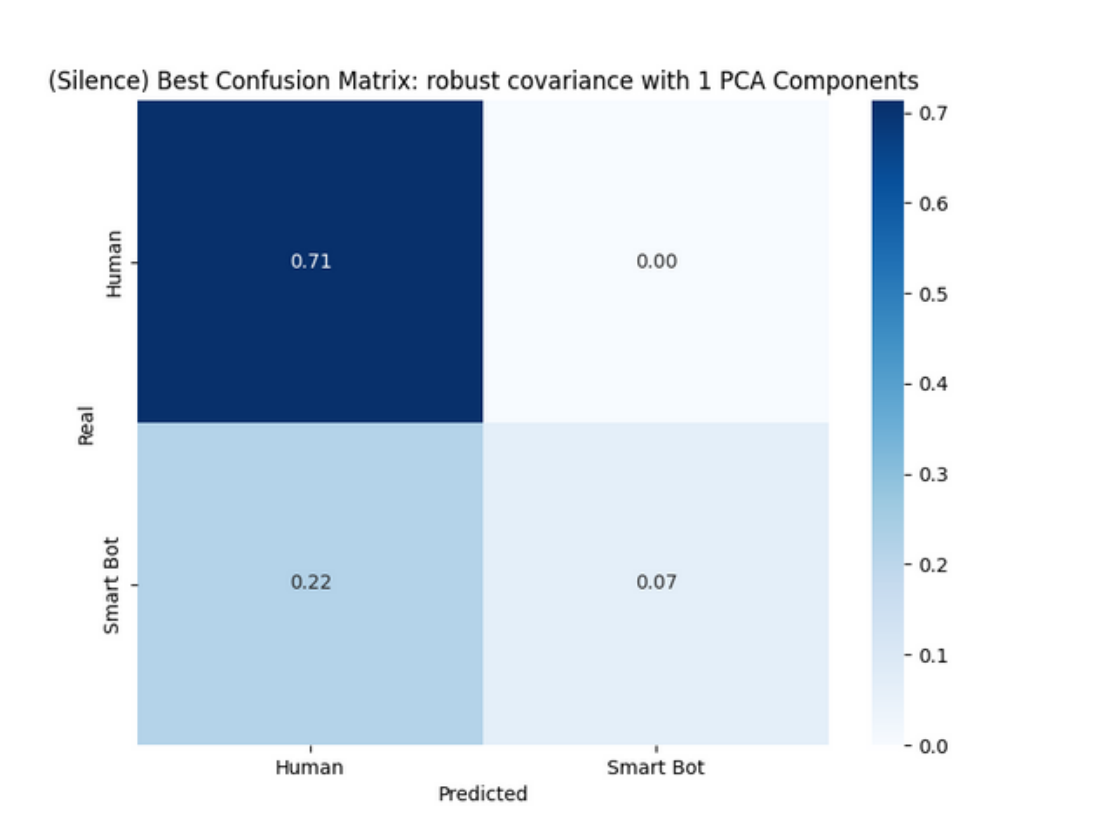
Basic vs Advanced Results

Robust Covariance PCA Components

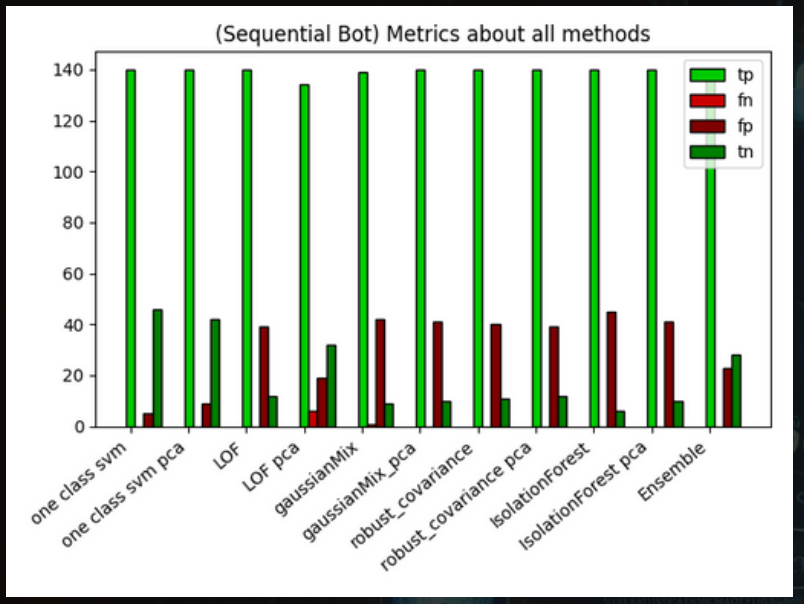
F1 Score: 87.77



F1 Score: 86.69



Ensemble - Basic vs Advanced



F1 - SCORE (Basic)

OCSVM: 98.25
OCSVM PCA: 96.89

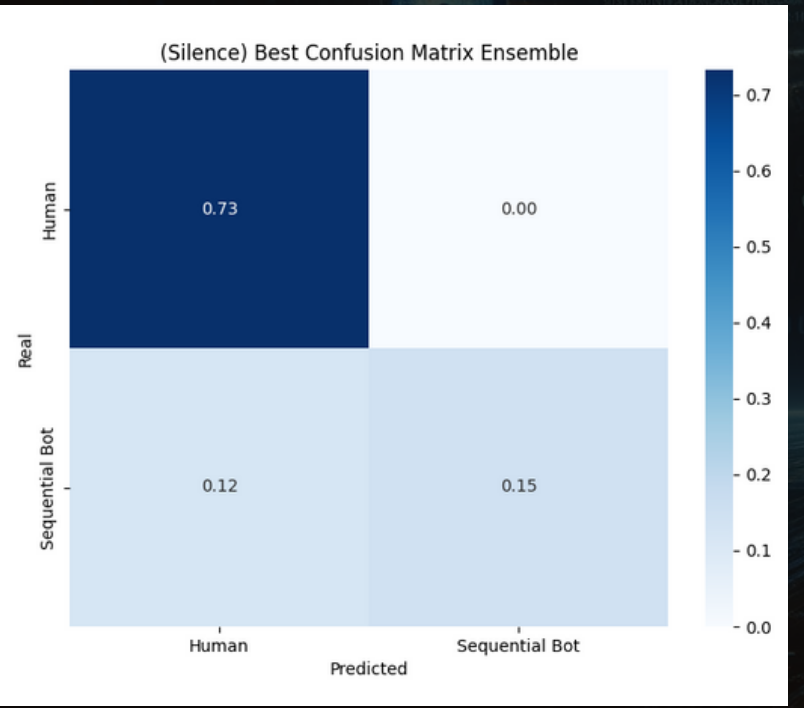
LOF: 87.77
LOF PCA: 91.47

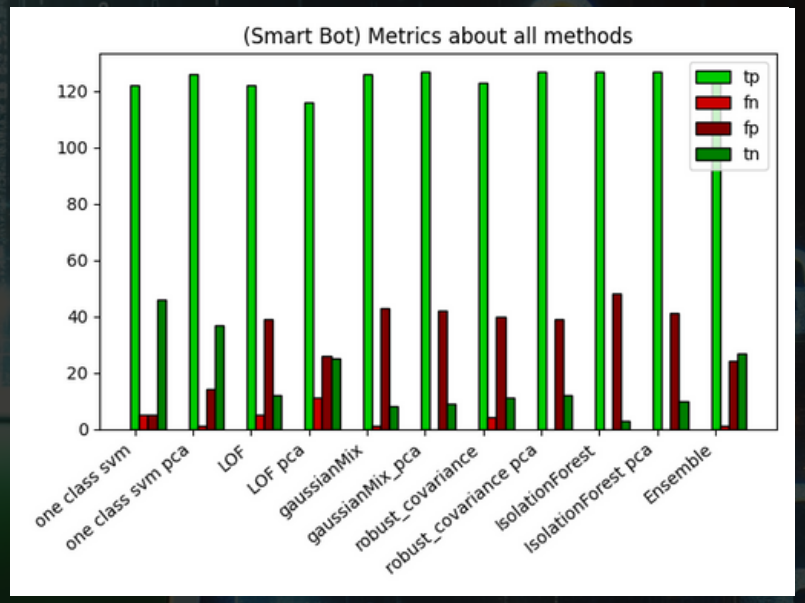
GMM: 86.6
GMM PCA: 87.23

RC: 87.5
RC PCA: 87.77

IF: 86.15
IF PCA: 87.23

Ensemble: 92.41





F1 - SCORE (Advanced)

OCSVM: 96.06
OCSVM PCA: 94.38

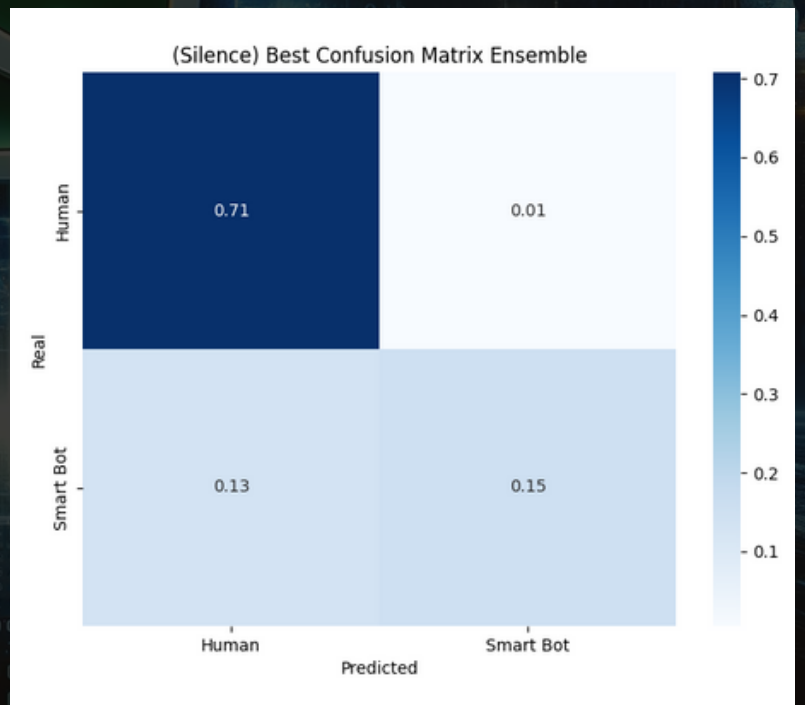
LOF: 84.72
LOF PCA: 86.25

GMM: 85.14
GMM PCA: 85.81

RC: 84.83
RC PCA: 86.69

IF: 84.11
IF PCA: 86.1

Ensemble: 90.97



Code

Github Repository:

- https://github.com/Raf4morim/TPR/


Any Questions?