Spy Your Mate

Riccardo Pezzoni

August 31, 2022

1 Introduction

Even if the traffic of a video conference call is encrypted, a network capture is able to reveal many metrics about the kinds of packets transmitted, their sizes and their rate. It is reasonable to assume that a machine learning algorithm, provided with those metric and a test set, can predict if at the certain moment the webcam was framing a human or not.

2 Work Description

2.1 Software

The call was performed with using the Zoom app running on macOs and the traffic was recorded with Wireshark running on the same machine. The capture produced was than exported as a CSV file and imported in Google Colab for ease of use and readability.

2.2 Data Processing

The CSV exported from Wireshark was the result of the application of the filter "ip.src == 192.168.2.107 && ip.dst == 149.137.12.171" to isolate the uplink traffic from the pc to the Zoom server. The downlink traffic was not considered because a change in some metric could have been the consequence of an action performed by the person the call was made with, which was not of interest. In Google Colab the DataFrame derived from the capture was split into one DataFrame for each protocol found: TCP, UDP, TLSv1.2 and WireGuard. Each one of those DataFrames was filtered to keep only the relevant time span of the experiment and than re-sampled in 500ms intervals extracting the number of packets and the median size of each period. The CSV representing the grand truth, realised with only the transition time, was than re-sampled as well in the same time intervals. A merge of those metrics was than provided to three classifiers:

- Logistic Regression

- Random Forest
- K-Nearest Neighbors

All of the classifiers, including the code needed to create the train and test sets and the various metrics further discussed, have been imported from Scikit-learn.

3 Results

3.1 Logistic Regression

The results of the logistic Regression algorithm are very good even with default setting. None of the changes performed lead to an increase of some metric. What is immediately visible is that false positive are much more present than false negatives.

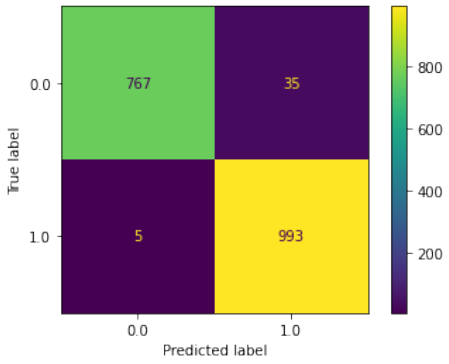

Figure 1: Logistic Regression confusion matrix

	precision	recall	f1-score	support
Not Present	0.99	0.96	0.97	802
Present	0.97	0.99	0.98	998
accuracy			0.98	1800
macro avg	0.98	0.98	0.98	1800
weighted avg	0.98	0.98	0.98	1800

Figure 2: Logistic Regression report

3.2 Random Forest

The Random Forest algorithm shows an improvement compared to Logistic Regression in both false positives and false negatives even at default settings,

reaching its best performance changing the number of trees to 200. As before, false positives are more present than false negatives.

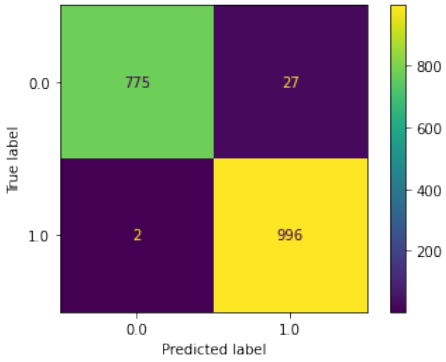

Figure 3: Random Forest confusion matrix

	precision	recall	f1-score	support
Not Present	1.00	0.97	0.98	802
Present	0.97	1.00	0.99	998
accuracy			0.98	1800
macro avg	0.99	0.98	0.98	1800
weighted avg	0.98	0.98	0.98	1800

Figure 4: Random Forest report

3.3 K-Nearest Neighbors

The K-Nearest Neighbors performance at default settings is comparable with the one obtained with Random Forest, and no changes lead to an improve of the performance.

4 Considerations

All the tested algorithms behaves extremely well in different executions and the performance degrades slowly when reducing the size of the training set. What is interesting to note is that all the algorithms tent to fail in the same time slots, so this is probably due to the data itself and not a fault of the algorithms. This can be proven by showing a side by side view of the graph of the points where at least one algorithm fails and the UDP data (here shown by vertical lines, while the underling square wave is the Grand Truth), which clearly in our case in the

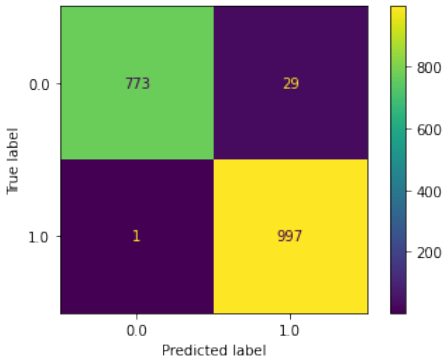

Figure 5: K-Nearest Neighbors confusion matrix

	precision	recall	f1-score	support
Not Present	1.00	0.96	0.98	802
Present	0.97	1.00	0.99	998
accuracy			0.98	1800
macro avg	0.99	0.98	0.98	1800
weighted avg	0.98	0.98	0.98	1800

Figure 6: K-Nearest Neighbors report

metric that leads the classification. With this view we can also note that the few false negatives are caused by a decrease in packet number, while the much more present false positives are due to a spike in median packet size.

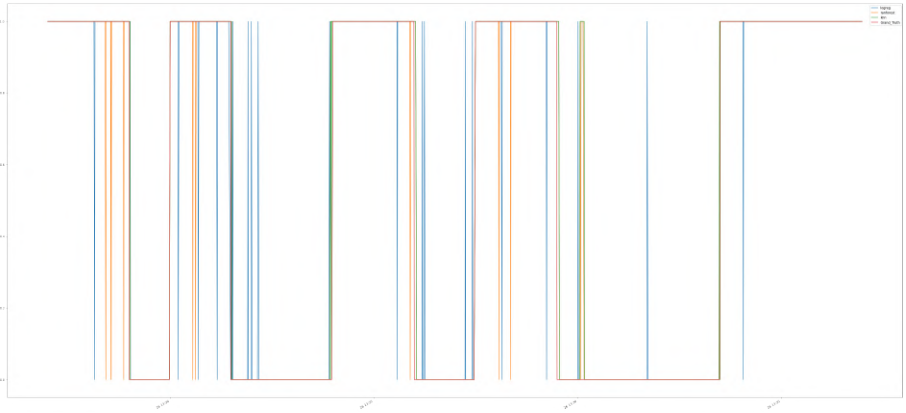

Figure 7: Mistakes made by at least one algorithm vs Grand Truth

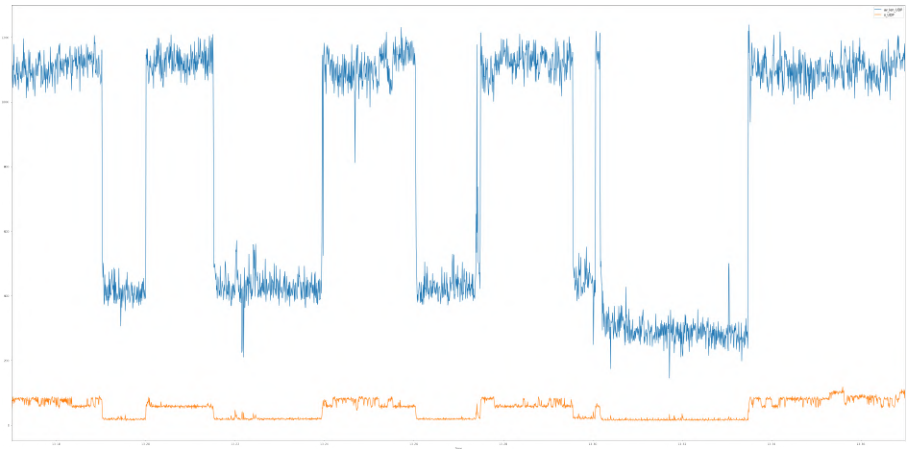

Figure 8: UDP metrics