MALWARE DETECTION

Use for IoT Devices

PROBLEM IDENTIFICATION

STATEMENT

IoT has proven to have a significant impact on human life by the integration of devices in a myriad of industries. There will be around **125 IoT devices connected to the internet by 2030.**

Developing **deployable technology in the form of algorithms, frameworks or even complete SIEM Systems, could be extremely useful to get a better understanding of the behavior of malware infections** where IoT devices are the main target.

STAKEHOLDERS TO PROVIDE KEY INSIGHTS

- Cyber Security Teams
- Antivirus Companies
- IoT Companies

SCOPE OF SOLUTION SPACE

Dataset: **20 malware captures executed in IoT devices, and 3 captures for benign IoT devices traffic.** Published in January 2020. These were captured in the [Stratosphere Laboratory, AIC group, FEL, CTU University, Czech Republic](#).
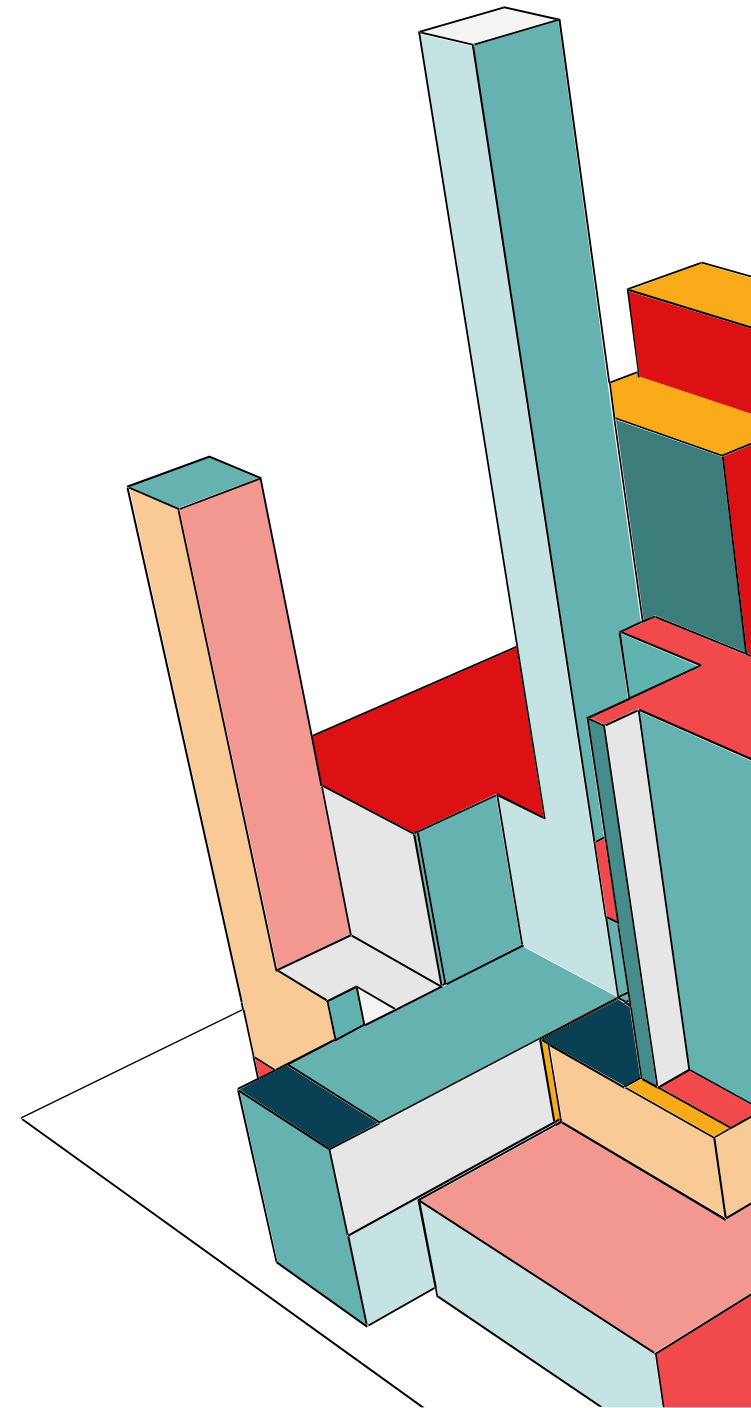
KEY DATA SOURCES

- Log files provided by the laboratory

- Classification methods spreadsheets
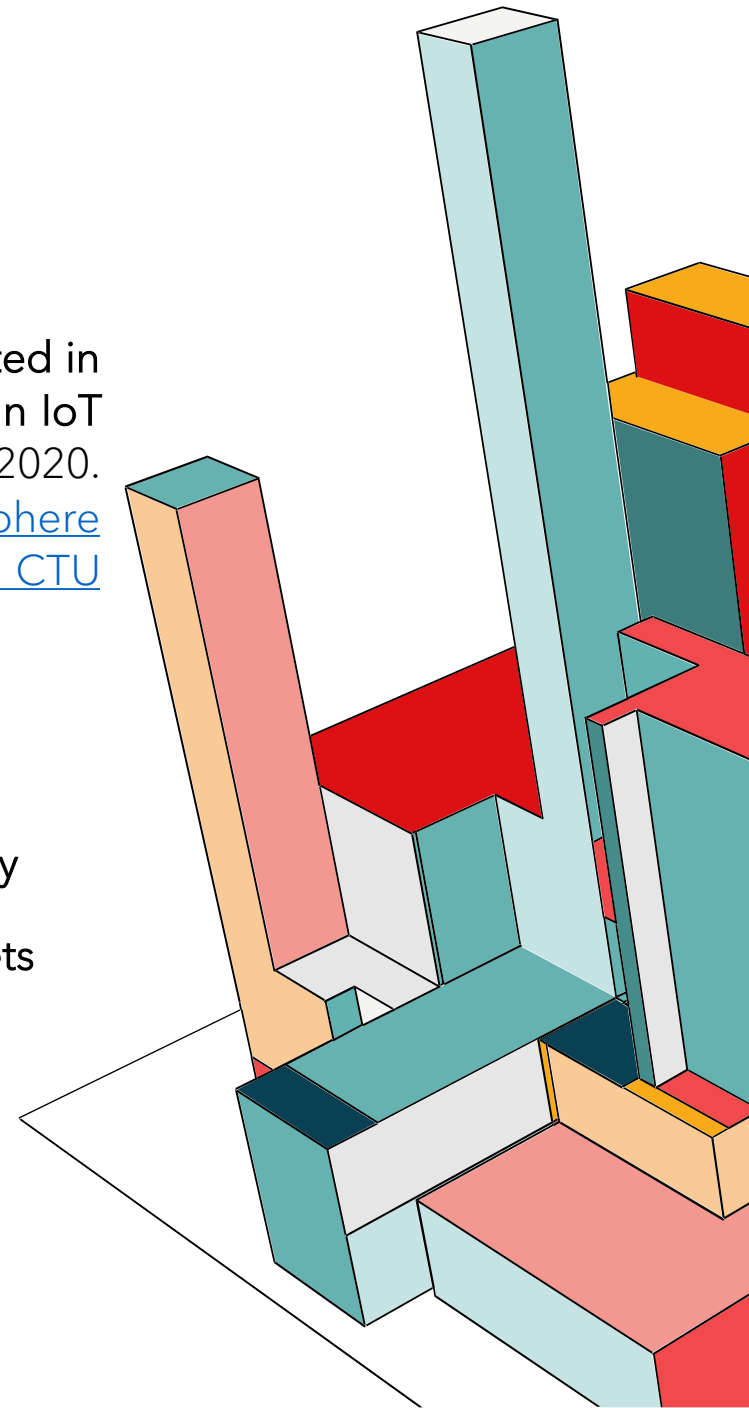
CRITERIA FOR SUCCESS

- Implement a Machine Learning algorithm able to detect at least 80% of malicious network flows

- Implementing a Malware Type Detector

- Having a deployable pipeline for malware detection in real-time setting

CONSTRAINS WITHIN SOLUTION SPACE

- Not getting enough **computational power** to analyze all log files.

- Malware information **being outdated** and not representative of how new malware works

- **Class imbalance** for the binary class or multi class project

KEY FINDINGS AND INSIGHTS

Binary classification project with a final dataset of:

380,000 observations

67 features

Key Features Enough for a One-Feature Model

Protocol
Origin Port
Port Used for Response
Network History
# of Bytes Sent to Host
# of Response Bytes
# of Response Packets

Constraints faced

Severe class imbalance for types of malware (8 classes)
Dataset only enough for **binary classification**
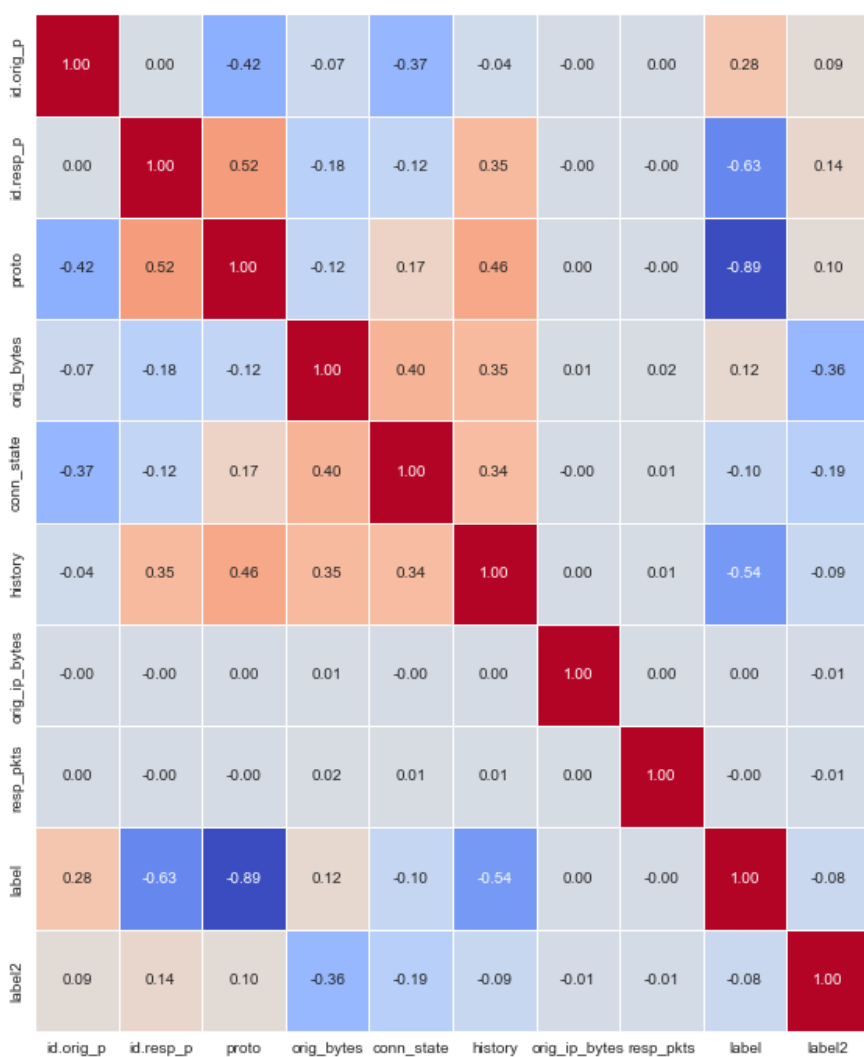
RAM power enough for analyzing 6 scenarios

Insights

Data Distribution between classes very marked in some features

Time-related Features nor dimensionality reduction

Key Features dropped to add complexity to the modeling
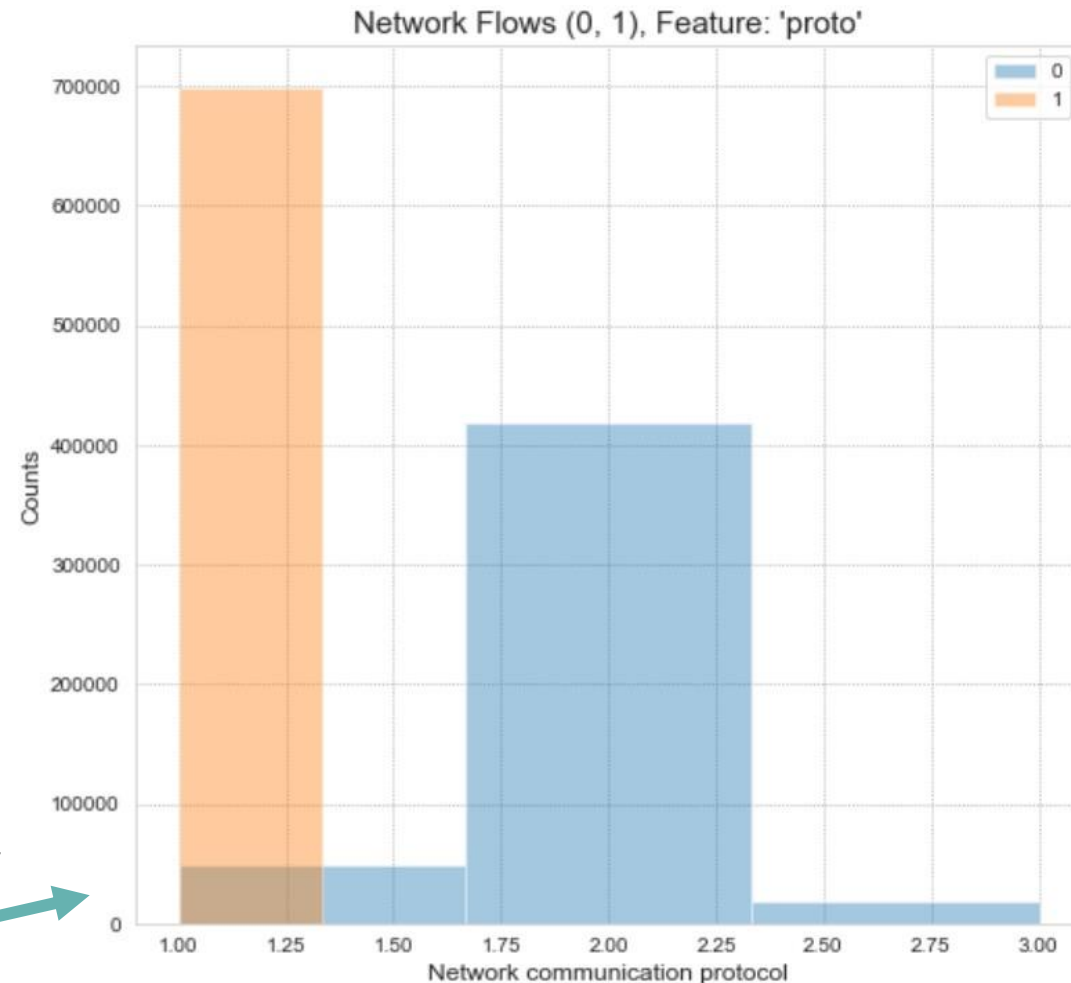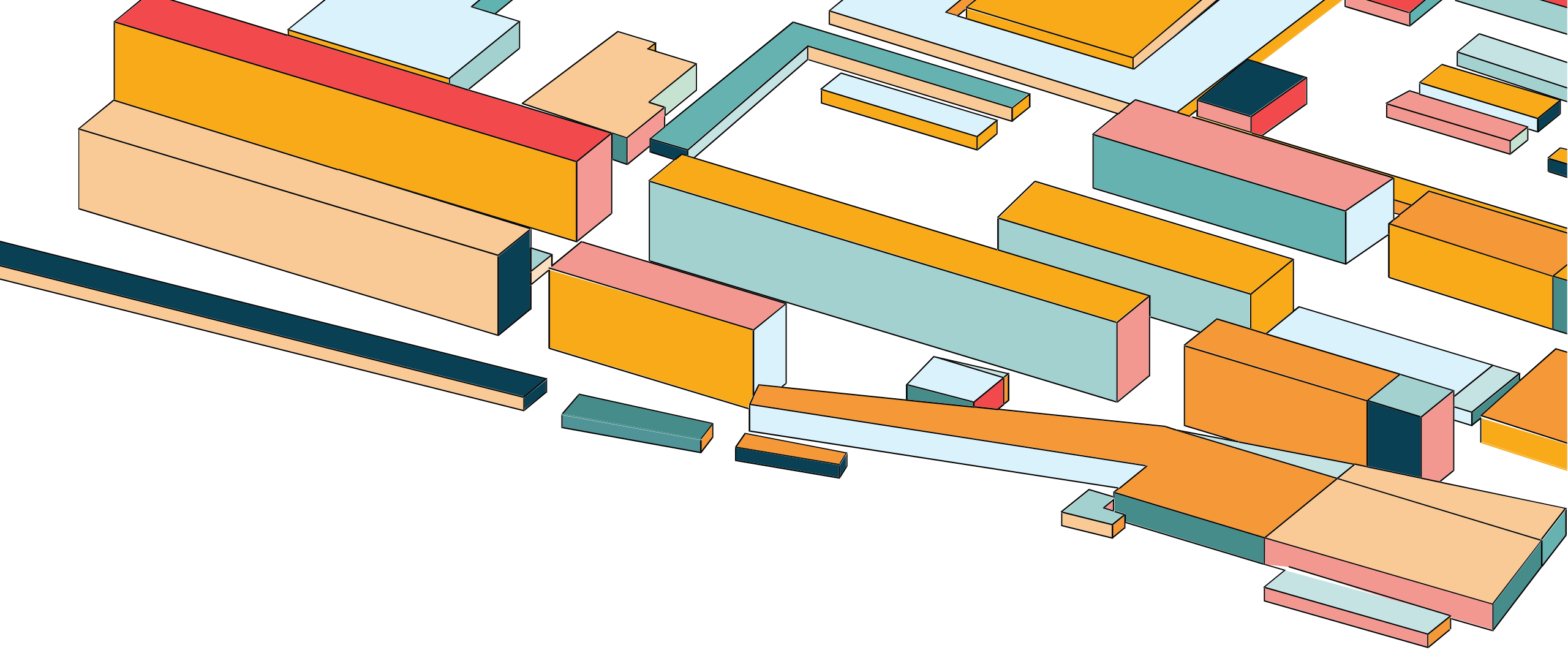
FEATURE HANDLING

Feature drop was based on:

- Collinearity

- Correlation with label 1

- Distribution differences between Classes (example of Protocol)

MODEL RESULTS AND ANALYSIS

"ONE-FEATURE" MODELS

Model	Feature	Precision (1)	Recall (1)
1	proto	0.89	1.0
2	id.orig_p	0.88	0.90
3	id.resp_p	0.90	0.92
4	History	0.90	0.93
5	orig_ip_bytes	0.88	1.0

Based on simple assumptions

Features showing high differences between classes were used

Example:

"All malicious network flows are using TCP protocol"

THANK YOU

Pablo Ruiz Lopez

+52 55-2729-6472

pabloweb8@gmail.com

https://github.com/pablo-git8

BMR Analysis