Effective IoT Device Identification at the Edge

Coseners 2020
32nd Multi-Service Networks workshop (MSN 2020)

Roman Kolcun, Anna Maria Mandalari, Yiming Xie,
Hamed Haddadi (Imperial College London)
Diana Andreea Popescu, Vadim Safronov,
Richard Mortier (Cambridge University)
Poonam Yadav (University of York)

July 9, 2020

Motivation

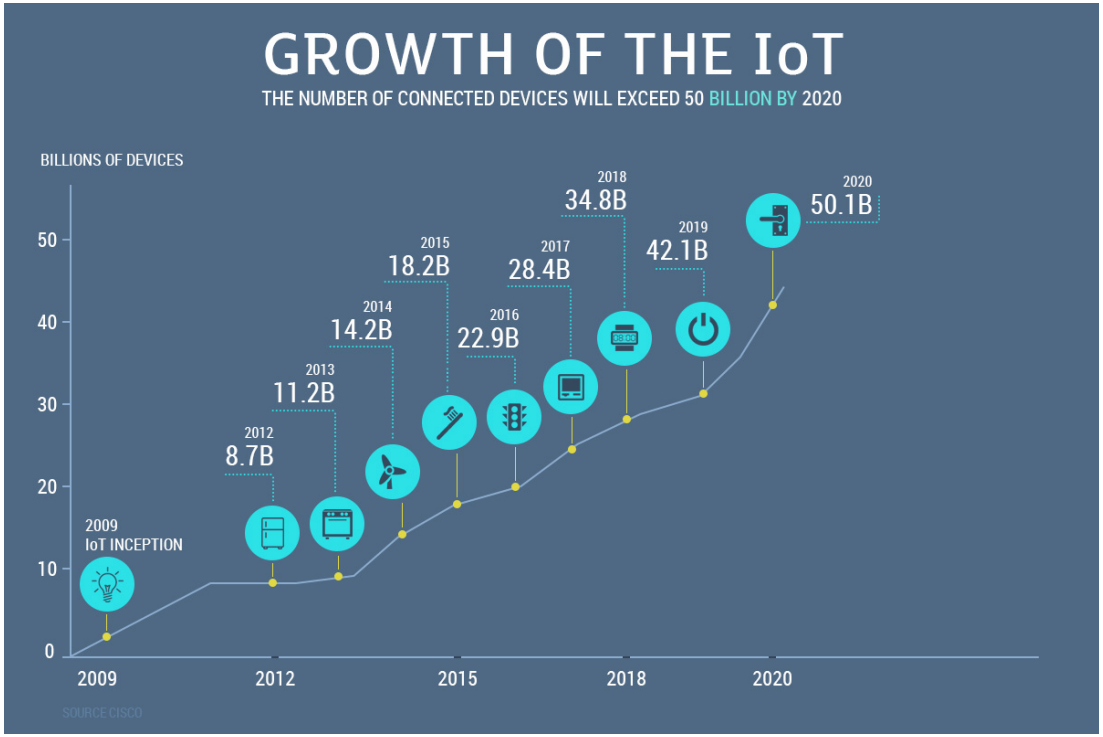


Figure: Number of IoT devices is rising

Motivation

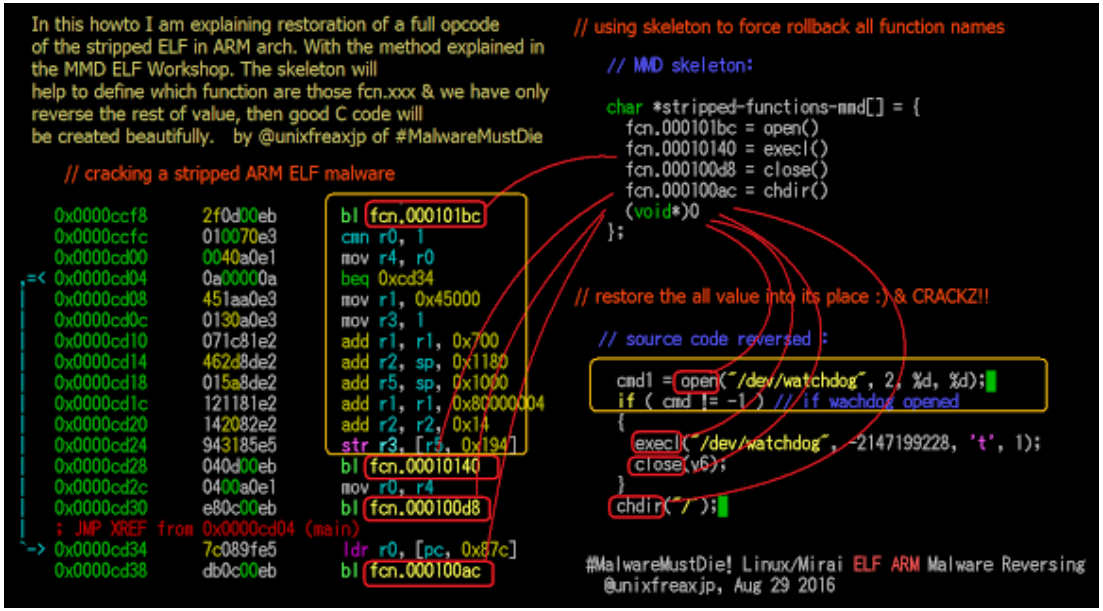


Figure: Apparently a part of Mirai source code

Motivation

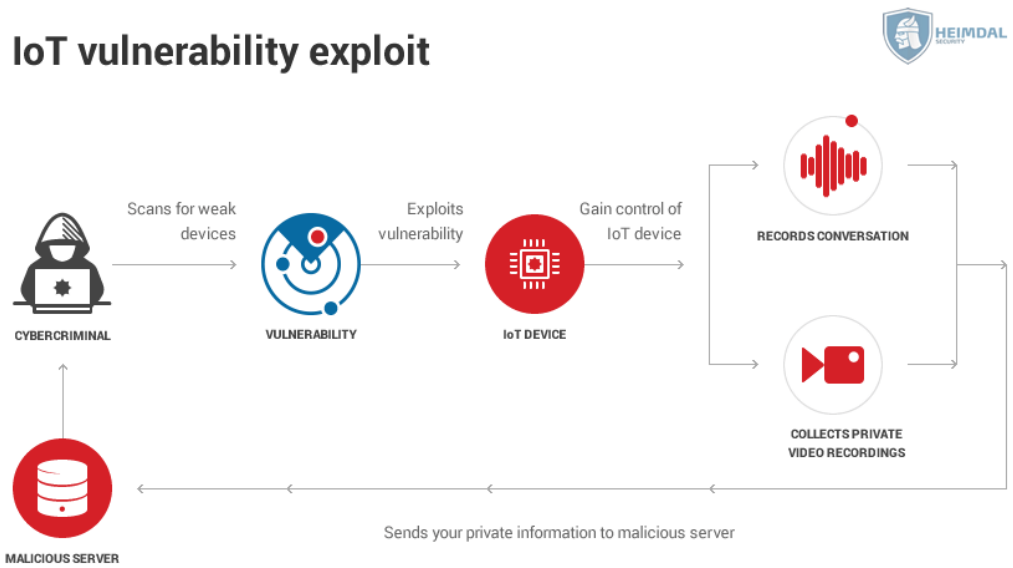


Figure: How vulnerabilities are exploited

Motivation

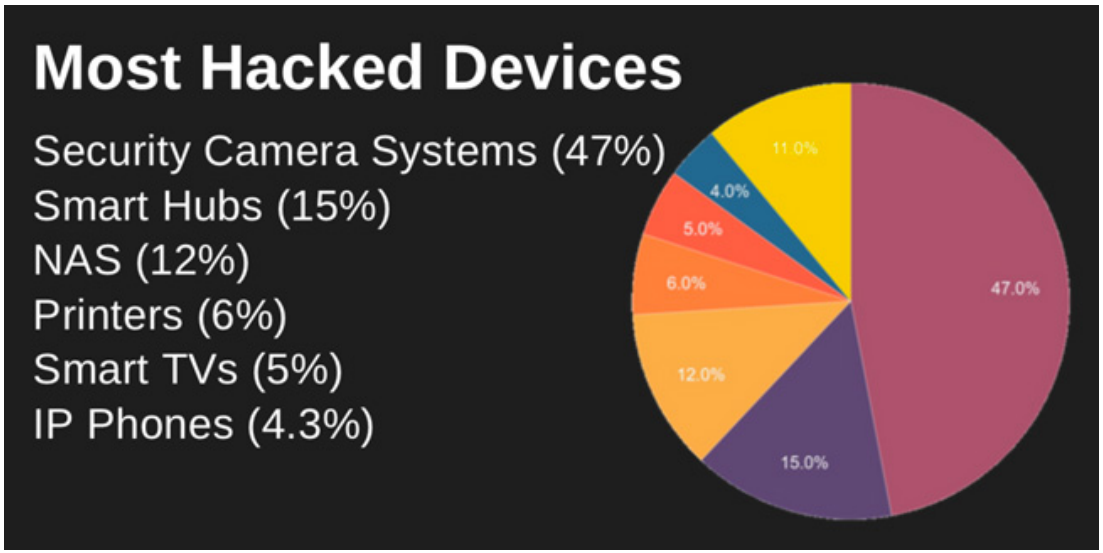


Figure: A pie chart

Motivation

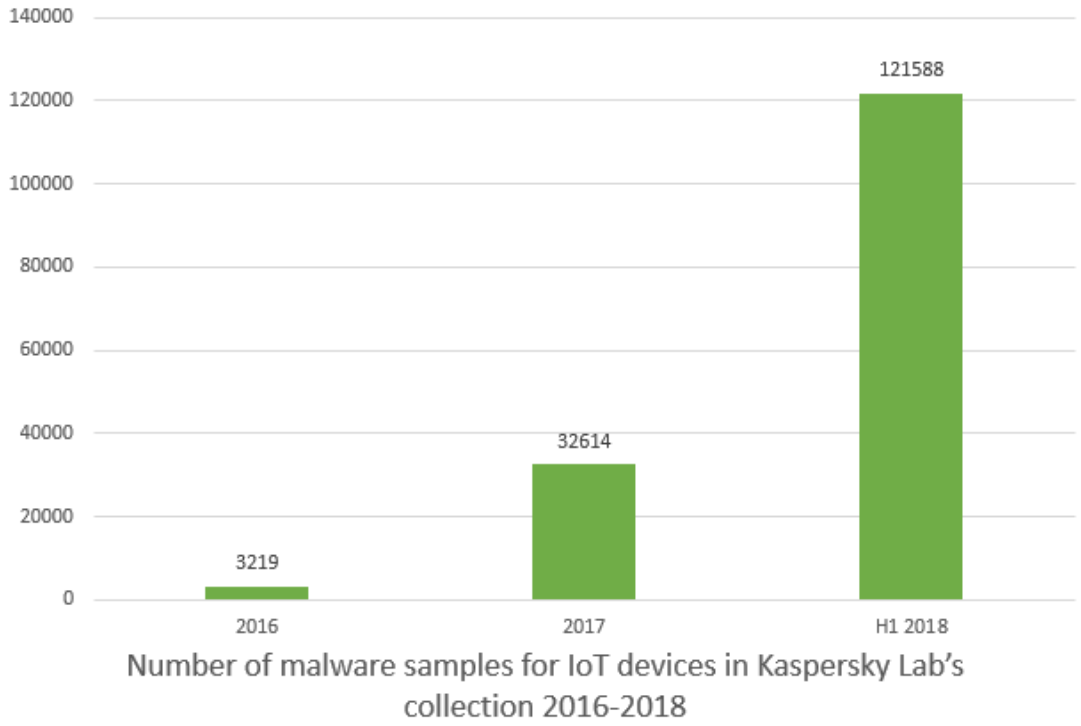


Figure: A bar chart

Motivation

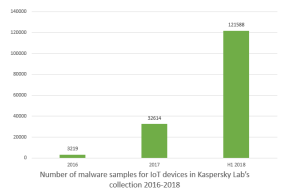


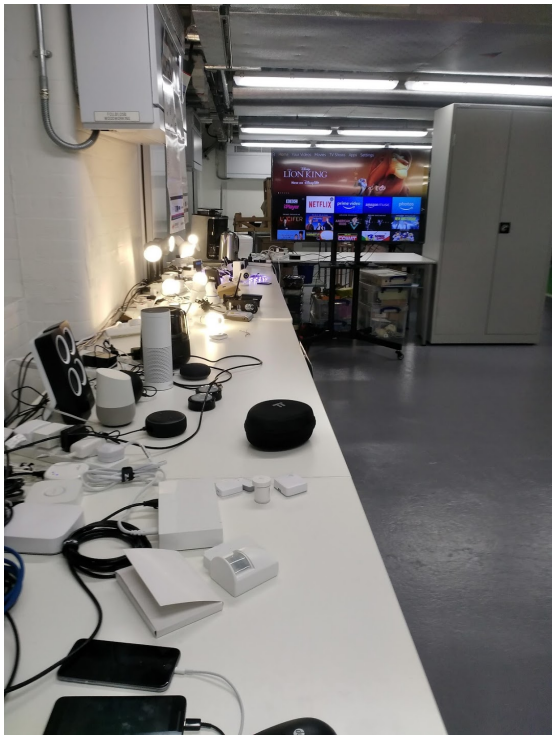
Figure: A bar chart

- IoT are penetrating our households and the number is ever rising
- IoT are source of large number of security threats
- They would benefit from automated management
- This requires identification of devices
- The most natural way is to identify them by their network traffic at the home router

Research Question

- Which ML models are best suited for this task?
- Once trained, do these models stay accurate?
- Is it feasible to run inference of the models at the edge?
- If needed, can these models be trained at the edge?

Test-beds



- Large Test-bed - 43 devices
- Small Test-bed - 9 devices (a subset)

All devices were split into 6 categories: Surveillance, Media, Audio, Hub, Appliance, Home Automation

Datasets

Two types of data were collected from both test-beds:

- Idle, i.e. no interaction with test-beds (3 weeks)
- Active, i.e. automated interaction with test-beds (1 week)

Evaluation

- Five different types of models:
  - Fully Connected Neural Network
  - LSTM Network
  - 1D Convolutional Network
  - Random Forest Classifier
  - Decision Tree Classifier
- Four different groups of models:
  - One model for all devices
  - One model per device
  - One model for all categories
  - One model per category

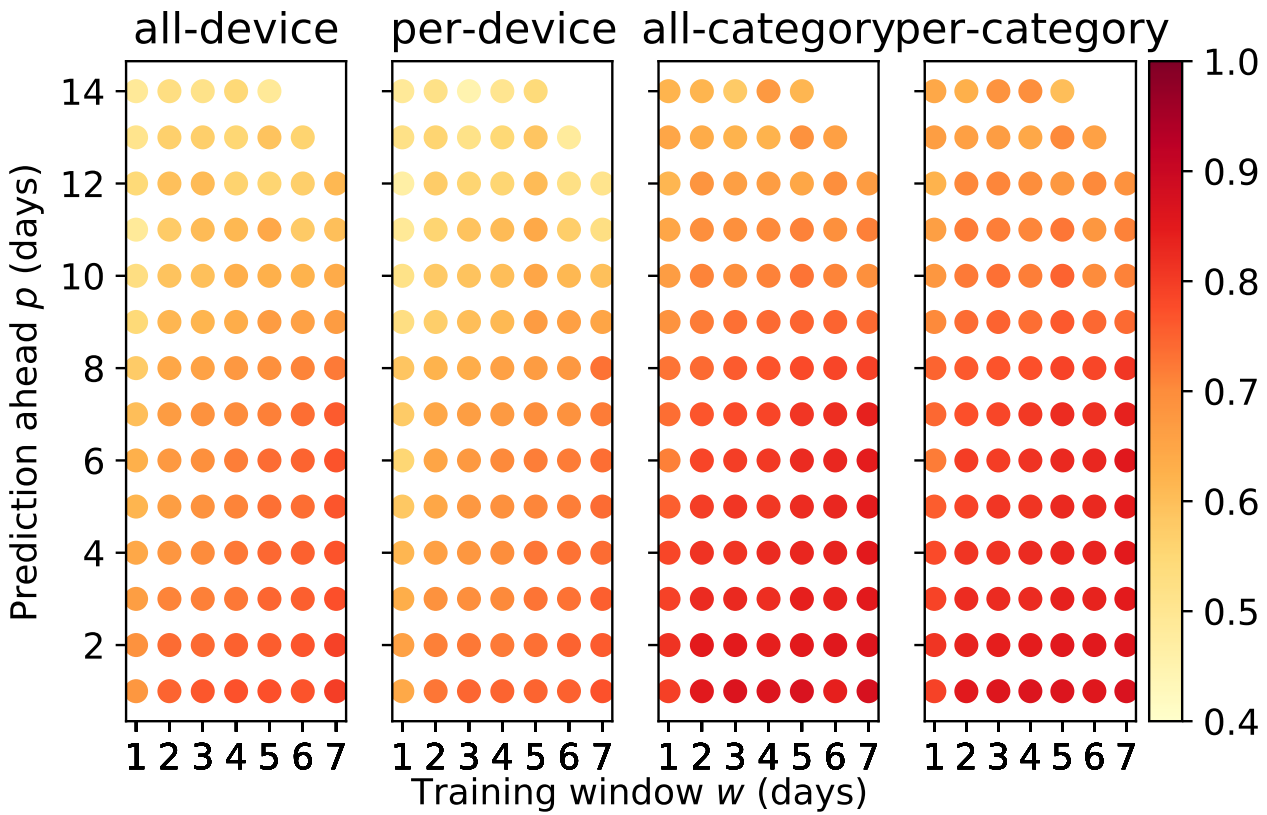
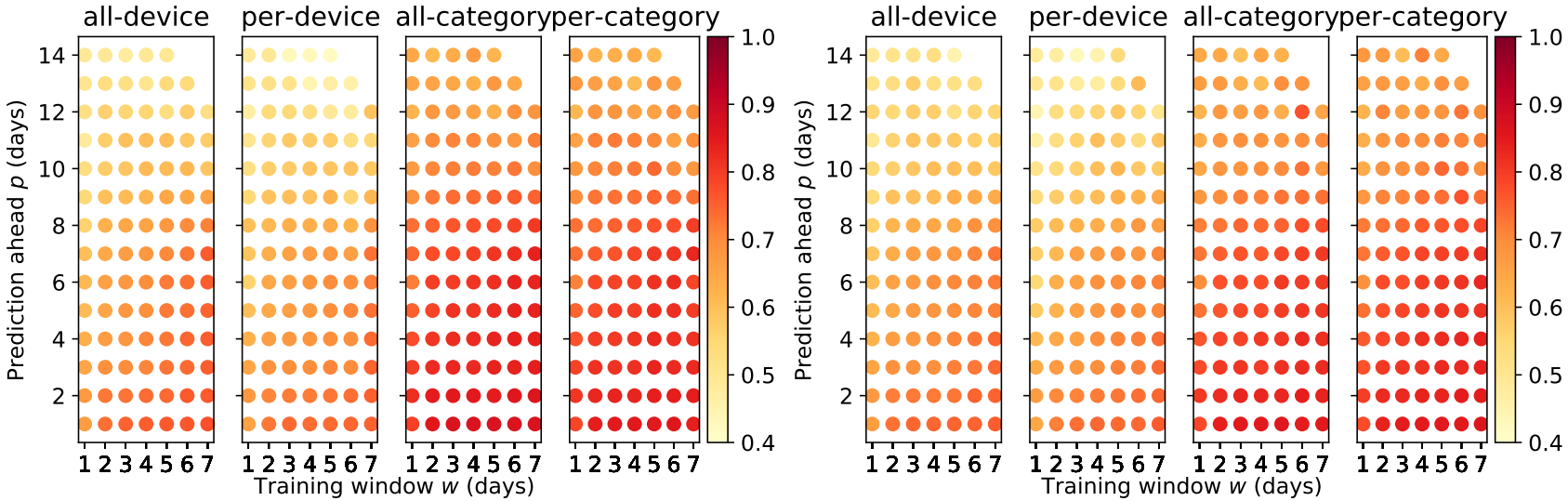
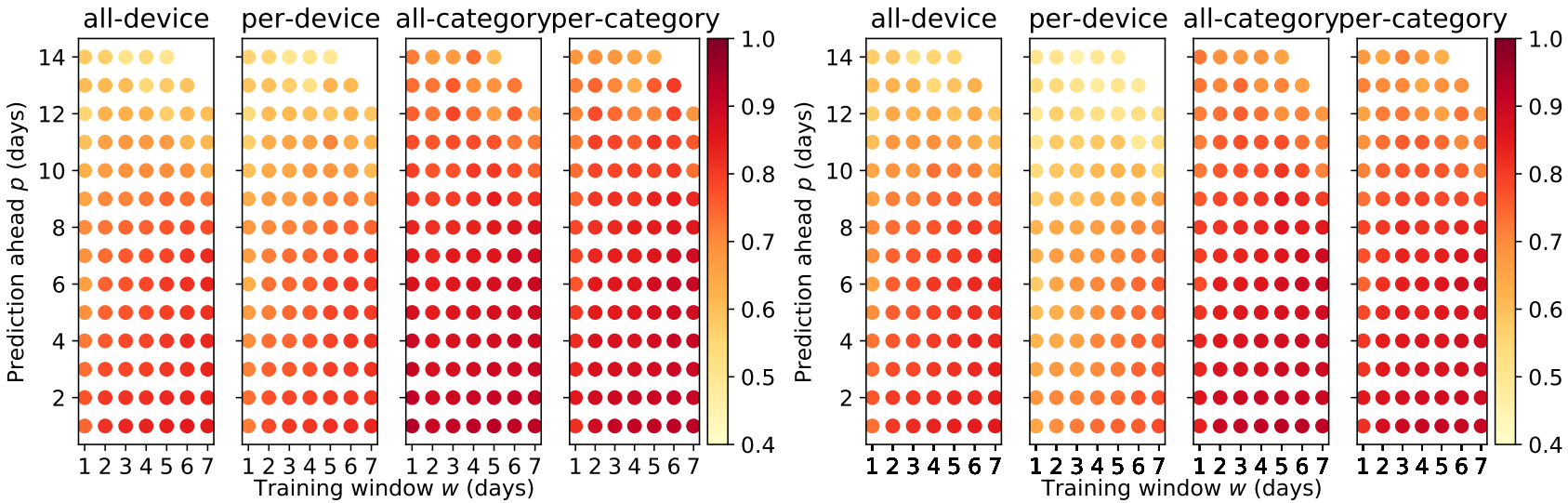


Figure: Fully Connected Network



(a) Long short-term memory model

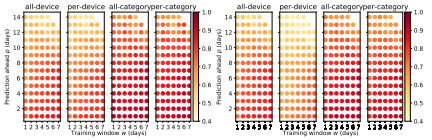
(b) 1D convolutional model



(a) Random forest classifier

(b) Decision tree classifier

Effective IoT Device Identification at the Edge



(a) Random forest classifier (b) Decision tree classifier

- Models performance is rather similar
- The accuracy of the models decrease over time
- The longer the training time, the longer the model remains accurate
- None of the models can reliably classify devices more than 2 weeks ahead
- Retraining of models is necessary

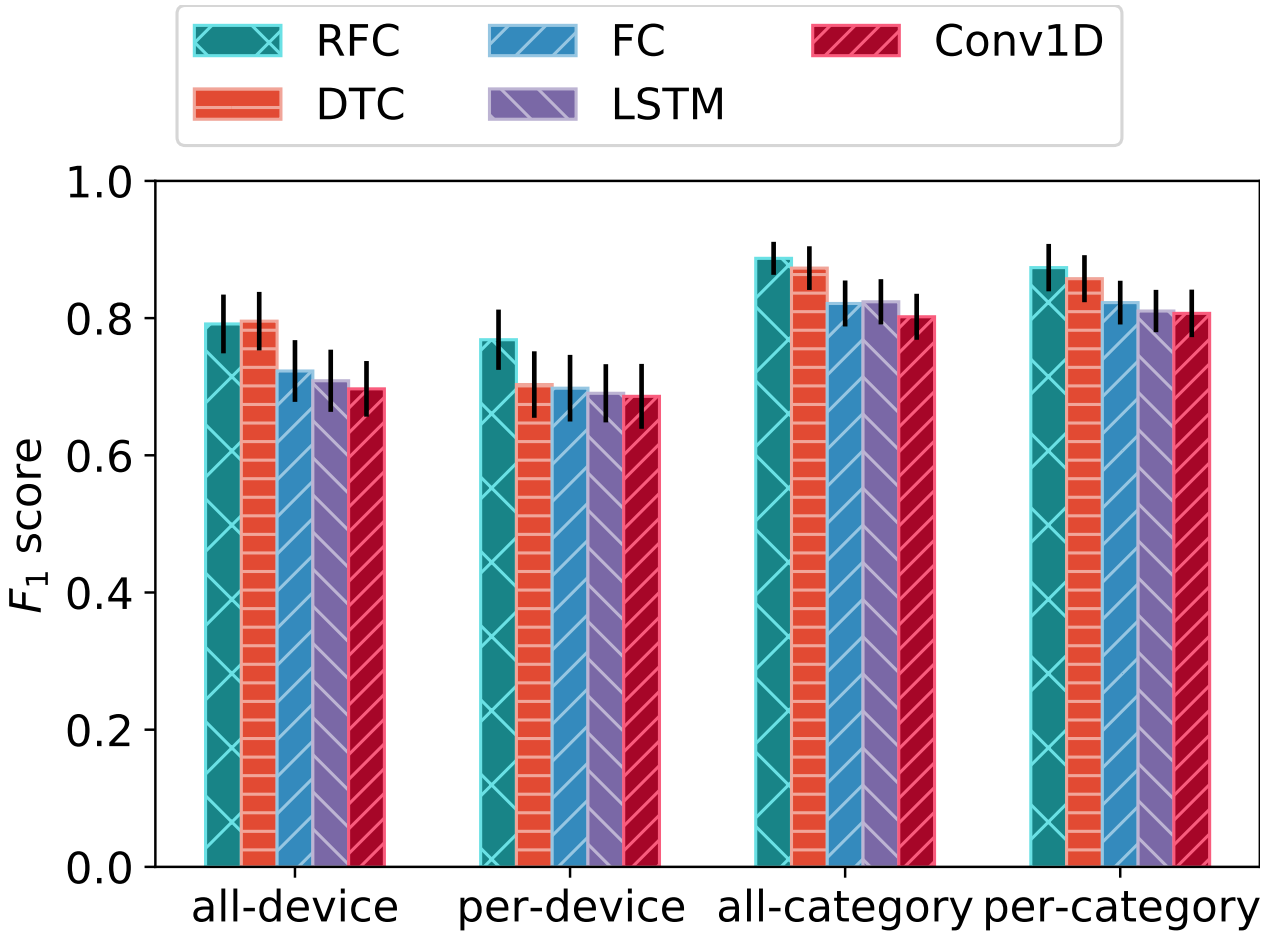


Figure: Average F_1 score of various models using a training window of various sizes (1-7) over the prediction of up to 7 days ahead.

Effective IoT Device Identification at the Edge

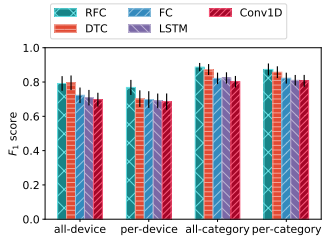
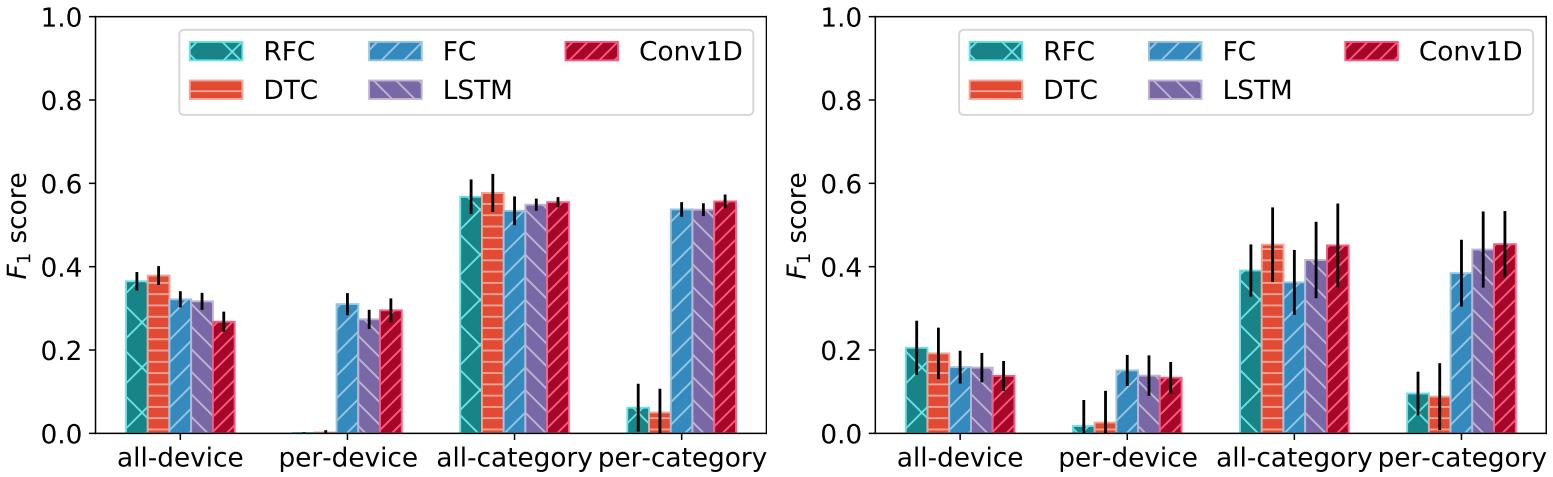


Figure: Average F_1 score of various models using a training window of various sizes (1-7) over the prediction of up to 7 days ahead.

- RFC & DTC models slightly outperform neural network based models
- Performance of neural network based models is virtually the same
- Device classification is less accurate than category classification
- Single multi-classification model outperforms multiple binary classification models

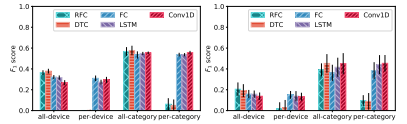


(a) Large Test-bed

(b) Small Test-bed

Figure: Average F_1 score of models trained on 7 day window of idle data and tested on active data of the large and the small test-bed.

Effective IoT Device Identification at the Edge

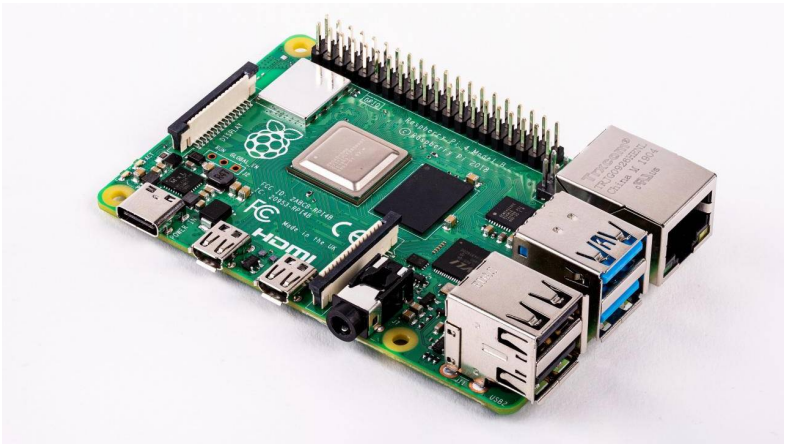


(a) Large Test-bed (b) Small Test-bed

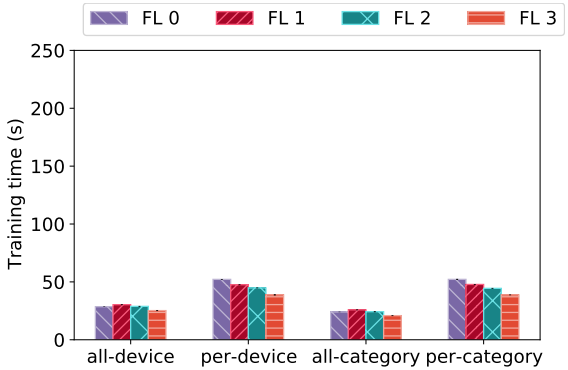
Figure: Average F_1 score of models trained on 7 day window of idle data and tested on active data of the large and the small test-bed.

- Models trained on an idle dataset are not accurate on active data
- Models trained on one test-bed are less accurate on the other test-bed
- Models updated with data from one test-bed increase accuracy on the same test-bed but have very small impact on the other test-bed
- Models need to be updated with local data

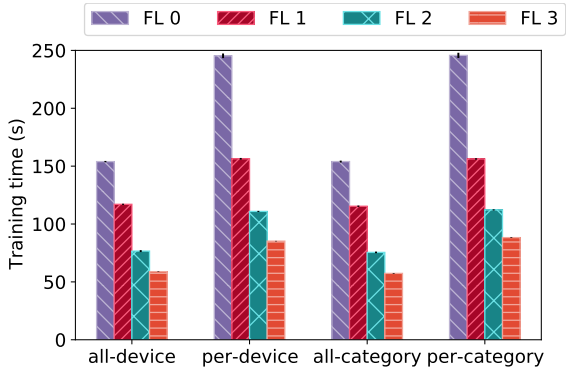
What Can be Done?



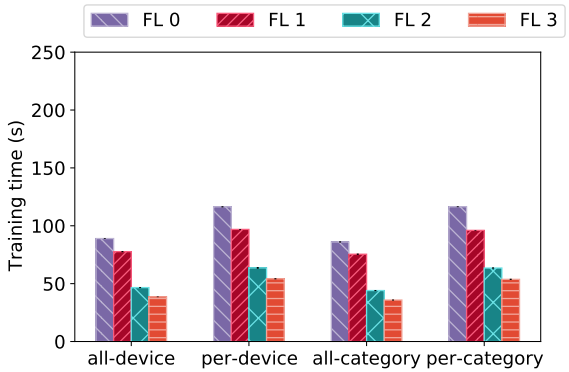
- Models need to updated frequently
- Models need to be updated with local data
- Model inference must be run at the edge
- Can a home router, aka Raspberry Pi 4, do it?



(a) Fully connected model



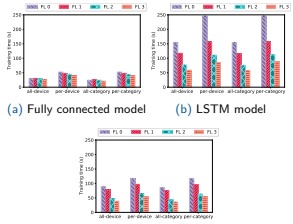
(b) LSTM model



(c) 1D conv. model

Figure: Average training time on RPi4 with different numbers of frozen layers (0, 1, 2 or 3 layers).

Effective IoT Device Identification at the Edge



(a) Fully connected model (b) LSTM model

(c) 1D conv. model

Figure: Average training time on RPi4 with different numbers of frozen layers (0, 1, 2 or 3 layers).

- Model retraining is feasible at the edge
- The improvement in training time largely depends on the type and the architecture of the neural network
- Layer freezing more than halves the training time for LSTM and Conv1D models
- Fully connected layer freezing reduces the training time modestly

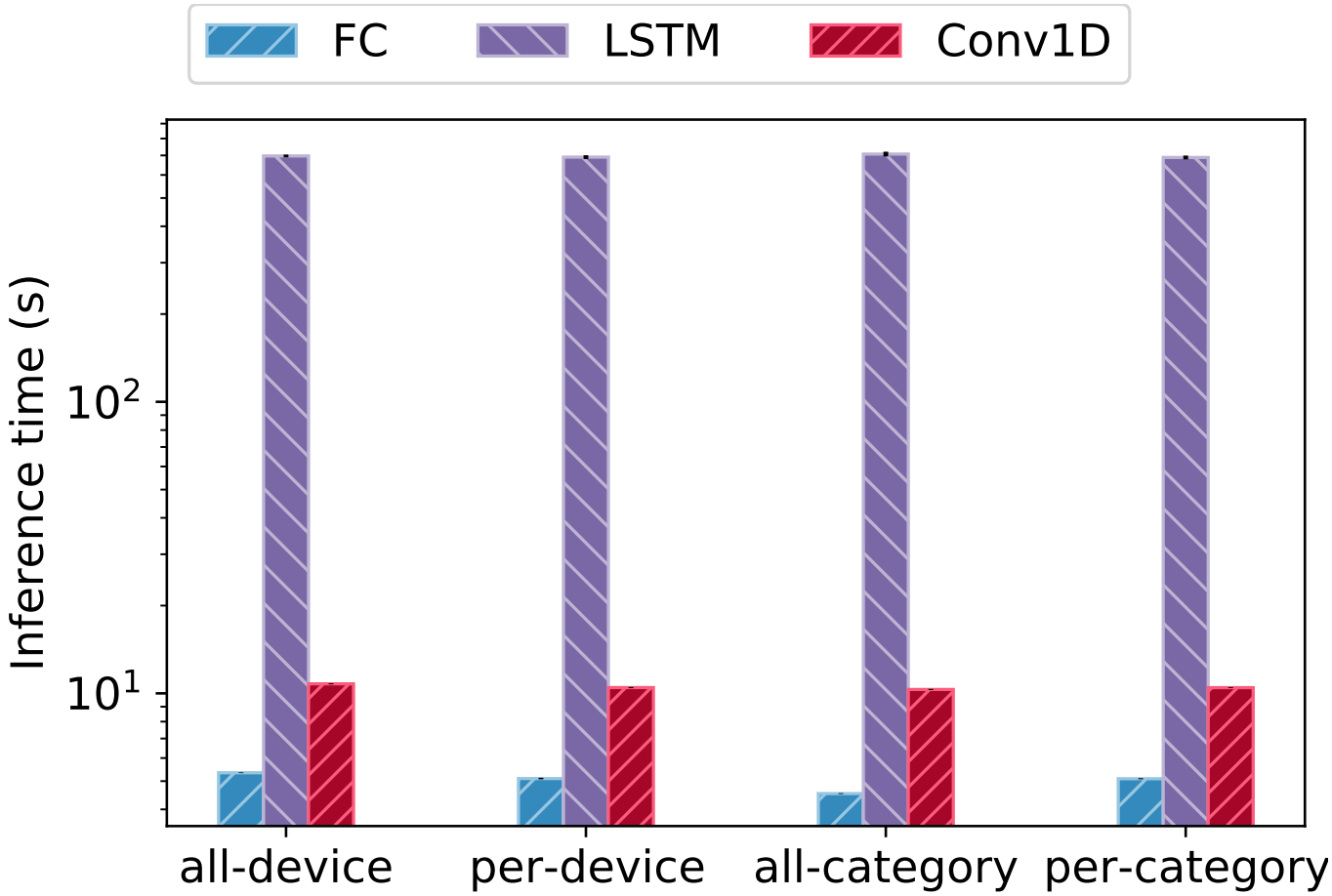


Figure: Average inference time on RPi4 of 100K samples using TensorFlow Lite.

Effective IoT Device Identification at the Edge

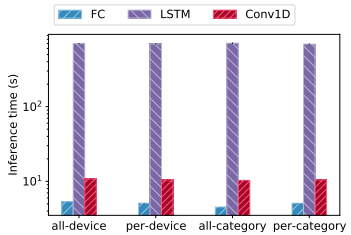


Figure: Average inference time on RPi4 of 100K samples using TensorFlow Lite.

- Results are similar across the all groups of models and model type
- The inference time for LSTM models is considerably larger
- Fully connected model is the fastest
- Inference time of 1D Convolutional model is double the time of FC model

Conclusion

- All models lose accuracy over time
- Models need to be updated with local data
- It is feasible to run model inference at the edge
- It is feasible to update the models at the edge

Is There A Way I Can Help?

Is There A Way I Can Help?

Would you like to know how chatty your IoT devices are?

Is There A Way I Can Help?

Would you like to know how chatty your IoT devices are?
Would you like to help researchers to collect data?

Is There A Way I Can Help?

Would you like to know how chatty your IoT devices are?
Would you like to help researchers to collect data? (or not)

Is There A Way I Can Help?

Would you like to know how chatty your IoT devices are?
Would you like to help researchers to collect data? (or not)
Good news: we will be running an experiment where you can install a Raspberry Pi 4 at your home which will act as a router for smart devices and send TCP/UDP headers (i.e. no payload) to our server.

Is There A Way I Can Help?

Would you like to know how chatty your IoT devices are?
Would you like to help researchers to collect data? (or not)
Good news: we will be running an experiment where you can install a Raspberry Pi 4 at your home which will act as a router for smart devices and send TCP/UDP headers (i.e. no payload) to our server.
Anyone interested should contact me at roman.kolcun@imperial.ac.uk or any of the aforementioned researchers.