

Deep in the Dark - Deep Learning-based Malware Traffic Detection without Expert Knowledge

Gonzalo Marín^{1,2} Germán Capdehourat² Pedro Casas³

(¹) Tryolabs (²) IIE-FING, UDELAR, Uruguay (³) AIT Austrian Institute of Technology

2nd Deep Learning and Security Workshop

IEEE S&P – May 23, 2019 – San Francisco, CA

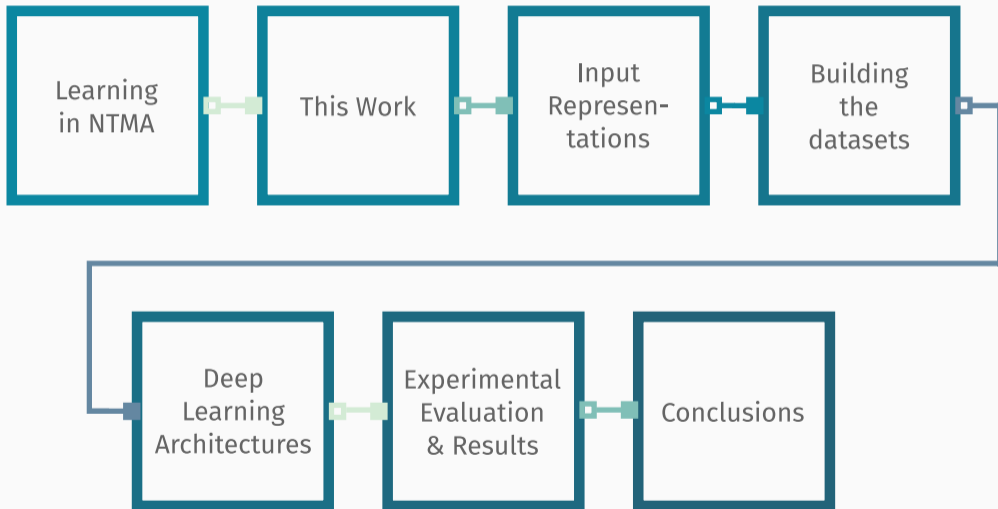
 tryo·labs



UNIVERSIDAD
DE LA REPÚBLICA
URUGUAY



W|W|T|F



Learning in NTMA

- ❖ The analysis of network traffic measurements is an **active research field**.
- ❖ **Machine learning** models are appealing since we have **tons of data and several problems to solve**.
- ❖ Some examples:
 - ❖ Traffic prediction and classification
 - ❖ Congestion control
 - ❖ Anomaly detection
 - ❖ Cybersecurity (e.g., malware detection, impersonation attacks)
 - ❖ QoE estimation
 - ❖ ...

- ❖ Traditional –shallow– machine learning models are commonly used.

- ❖ Traditional –shallow– machine learning models are commonly used.
- ❖ Decision trees and random forest, SVM, k-NN, DBSCAN... the list is **as vast as the associated literature.**

- ❖ Traditional –shallow– machine learning models are commonly used.
- ❖ Decision trees and random forest, SVM, k-NN, DBSCAN... the list is as vast as the associated literature.
- ❖ Feature engineering needed!

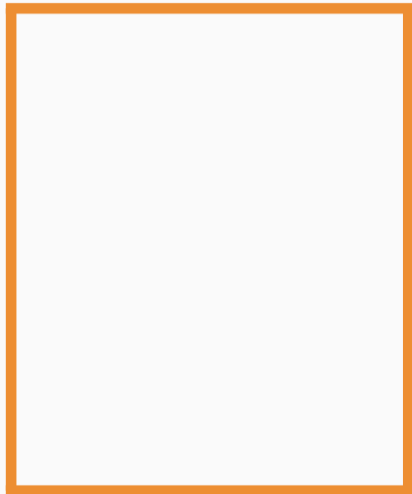


- ❖ Traditional –shallow– machine learning models are commonly used.
- ❖ Decision trees and random forest, SVM, k-NN, DBSCAN... the list is as vast as the associated literature.
- ❖ Feature engineering needed!
- ❖ Handcrafted-expert domain features are **critical** to the success of the applied models.

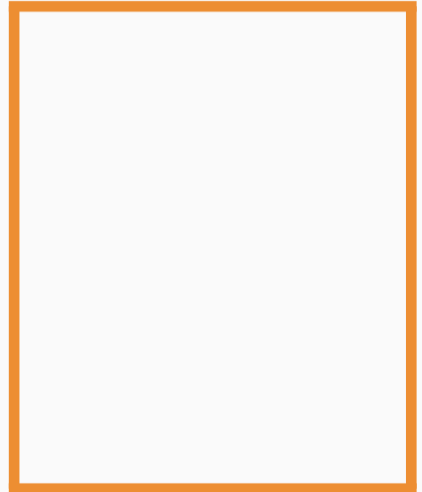


- ❖ Features are heavily dependant on the expert background and the specific problem.

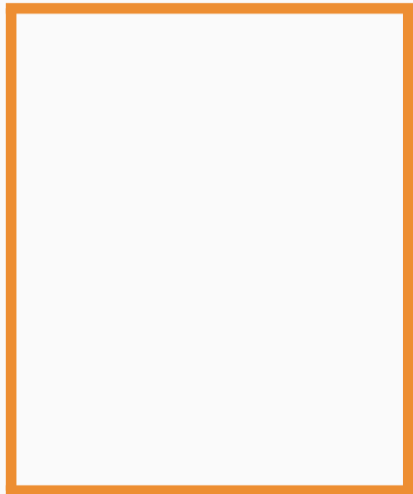
- ❖ Features are heavily dependant on the expert background and the specific problem.
- ❖ Each paper in the literature defines its own set of input features for the considered problem, hindering generalization and benchmarking of different approaches.



- ❖ Features are **heavily dependant** on the expert background and the specific problem.
- ❖ Each paper in the literature defines its own set of input features for the considered problem, hindering generalization and benchmarking of different approaches.
- ❖ Feature engineering is **costly**.



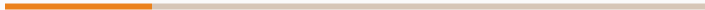
- ❖ Features are **heavily dependant** on the expert background and the specific problem.
- ❖ Each paper in the literature defines its **own set of input features for the considered problem**, hindering generalization and benchmarking of different approaches.
- ❖ Feature engineering is **costly**.
- ❖ All in all, **good results can be achieved**.




; *D* *a*





This Work





- ❖ **Main goal:** to explore the end-to-end application of deep learning models to **complement traditional approaches** for NTMA, using different representations of the input data.

- ❖ **Main goal:** to explore the end-to-end application of deep learning models to **complement traditional approaches** for NTMA, using different representations of the input data.
- ❖ To do this, **malware traffic detection and classification** problem is addressed, using  , bytestream-based data as input.

- ❖ **Main goal:** to explore the end-to-end application of deep learning models to **complement traditional approaches** for NTMA, using different representations of the input data.
- ❖ To do this, **malware traffic detection and classification** problem is addressed, using , bytestream-based data as input.
- ❖ **Research questions**

- ❖ **Main goal:** to explore the end-to-end application of deep learning models to **complement traditional approaches** for NTMA, using different representations of the input data.
- ❖ To do this, **malware traffic detection and classification** problem is addressed, using , bytestream-based data as input.
- ❖ **Research questions**
 1. Is it possible to achieve **high detection accuracy** with **low false alarm rates** using the **raw-input, deep learning-based models**?

- ❖ **Main goal:** to explore the end-to-end application of deep learning models to **complement traditional approaches** for NTMA, using different representations of the input data.
- ❖ To do this, **malware traffic detection and classification** problem is addressed, using , bytestream-based data as input.
- ❖ **Research questions**
 1. Is it possible to achieve **high detection accuracy** with **low false alarm rates** using the **raw-input, deep learning-based models**?
 2. Are the proposed models  than the commonly used shallow models, when **feeding them all with raw inputs**?

- ❖ **Main goal:** to explore the end-to-end application of deep learning models to **complement traditional approaches** for NTMA, using different representations of the input data.
- ❖ To do this, **malware traffic detection and classification** problem is addressed, using **raw-input**, bytestream-based data as input.
- ❖ **Research questions**
 1. Is it possible to achieve **high detection accuracy** with **low false alarm rates** using the **raw-input, deep learning-based models**?
 2. Are the proposed models **more accurate** than the commonly used shallow models, when **feeding them all with raw inputs**?
 3. How good are these models **as compared to traditional approaches**, where domain expert knowledge is used to build the set of features?

Input Representations

- ❖ Decimal normalized representation of each of each packet is considered as a different .

- ❖ Decimal normalized representation of each `packet` of each packet is considered as a different `packet`.
- ❖ Each `packet` is considered as a different `packet`.


- ❖ Decimal normalized representation of each `packet` of each packet is considered as a different `packet`.
- ❖ Each `packet` is considered as a different `packet`.
- ❖ It is necessary to choose the **number of bytes** from the packet to be considered (`len`).


❏ A is considered as a different .



- ❖ A `Flow` is considered as a different `Flow`.
- ❖ Each `Flow` (group of packets) is considered as a different `Flow`.



- ❖ A is considered as a different .
- ❖ Each (group of packets) is considered as a different .
- ❖ It is necessary to choose the **number of bytes** from the packet to consider () and the **number of packets per flow** to consider ().

Building the Datasets

- ❖ Malware and normal captures performed by the  of the CTU University of Prague in Czech Republic were considered.

- ❖ Malware and normal captures performed by the  of the CTU University of Prague in Czech Republic were considered.
- ❖ Captures are gathered under **controlled conditions**: fixed scenario (IPs, ports, etc.)

- ❖ **Malware and normal captures** performed by the  of the CTU University of Prague in Czech Republic were considered.
- ❖ Captures are gathered under **controlled conditions**: fixed scenario (IPs, ports, etc.)
- ❖ **Not**  network traffic.

- ❖ **Malware and normal captures** performed by the  of the CTU University of Prague in Czech Republic were considered.
- ❖ Captures are gathered under **controlled conditions**: fixed scenario (IPs, ports, etc.)
- ❖ **Not**  network traffic.
- ❖ Let's consider the **payload**, as the key information to analyze and to build the datasets.

Representation	Dataset size	n (bytes)	m (packets)
\hat{u} \hat{o}	248;850	1024	N/A
\hat{u} o	67;494	100	2

Table 1: Parameters selection for building the input representation for training the deep learning models.

Deep Learning Architectures

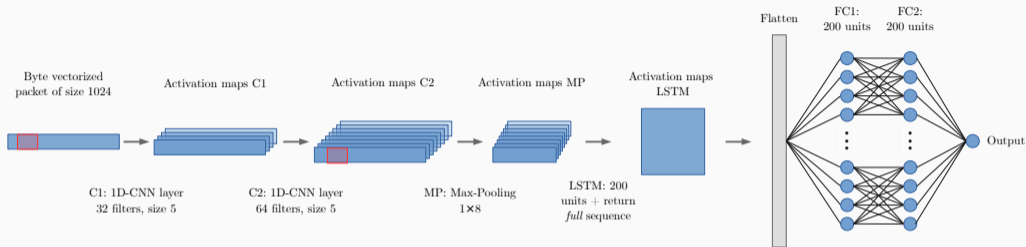
o



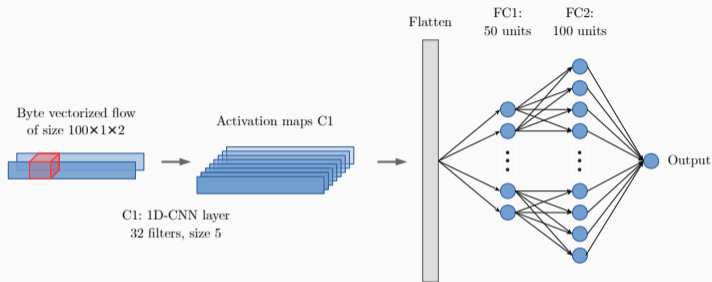
- ❖ The core layers used for both models are basically two: convolutional and recurrent.
- ❖ **Convolutional**, to **build the feature representation** of the spatial data inside the packets and flows.
- ❖ The **recurrent** layers will be used together with the convolutional to improve the performance of the architecture, allowing the model to keep track of temporal information.
- ❖ **Fully-connected** layers to deal with the different combinations of the features in order to arrive to the final decisions (i.e., classify).

Goals: reduce the generalization error and improve the learning process.

- ❖ **Batch Normalization:** layer inputs are normalized for each mini-batch. As a result: higher learning rates can be used, model less sensitive to initialization and also adds regularization.
- ❖ **Dropout:** randomly drop units (along with their connections) from the neural network during training.
 - similar to train a huge number of different networks and average the results.



- ❖ 2 **1D-CNN** layers of 32 and 64 filters of size 5, respectively.
- ❖ A **max-pooling** layer of size 8.
- ❖ A **LSTM** layer of 200 units, returning the outputs of each cell.
- ❖ 2 **fully-connected** layers of 200 units each.
- ❖ Binary **cross-entropy** is used as the loss function.



- ❖ Smaller capacity than $\hat{u} \hat{o}$ (less number of features).
- ❖ 1 1D-CNN layer of 32 filters of size 5 and 2 fully-connected layers of 50 and 100 units each.
- ❖ Also, **binary cross-entropy** is used as the loss function.

Experimental Evaluation & Results

¶ D
 o
D a



Detect **malware at packet level**.

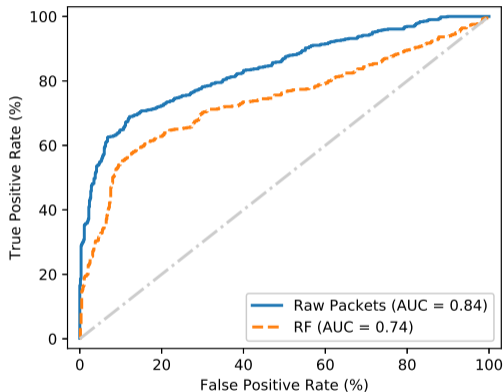
- ❖ \hat{u} \hat{o} deep learning architecture trained using the respective dataset version (250;000 samples).
- ❖ Split using a 80/10/10 schema: 80% for training, 10% for validation and 10% for testing.
- ❖ Training held over 100 epochs.
- ❖ Adam used as the optimizer function, annealing the learning rate over time.

Raw Packets learning process.



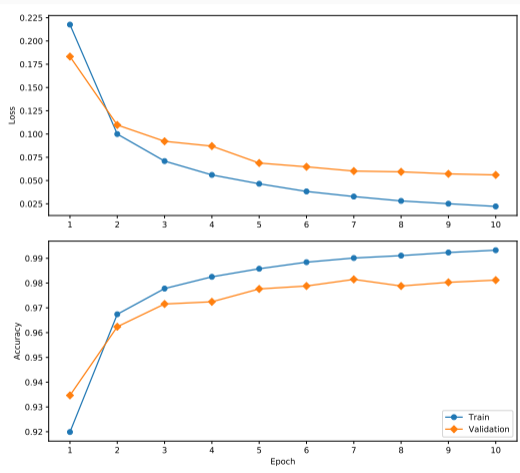
Raw Packets learning process.

- ❖ Accuracy: **77.6%** over the test.
- ❖ Comparison with a random forest model (100 trees), using **exactly the same input features**.
- ❖ **à** **ò** deep learning model **outperforms the random forest one.**



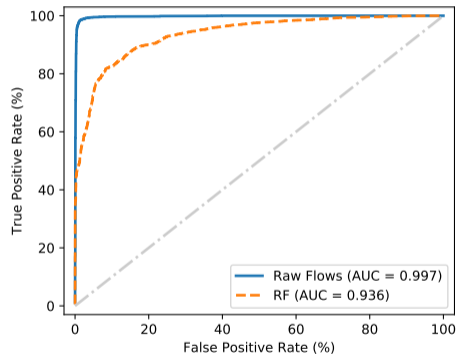
Detect **malware at flow level**.

- ❖ \hat{u} o deep learning architecture trained using the respective dataset version (68;000 samples).
- ❖ Split using a 80/10/10 schema: 80% for training, 10% for validation and 10% for testing.
- ❖ Training held over 10 epochs.
- ❖ Adam used as the optimizer function.



Raw Flows learning process.

- ❖ Accuracy: **98.6%** over the test.
- ❖ Comparison with random forest: data was flattened in order to fit the input.
- ❖ ù o model can detect as much as **98% of all malware flows with a FPR as low as 0.2%**.
- ❖ This suggests that operating at flow level, ù o can actually provide highly accurate results, applicable in practice.



Domain knowledge vs. raw inputs

- ❖ How good is \hat{y} as compared to a random forest trained with a dataset made of expert-handcrafted features?

Domain knowledge vs. raw inputs

- ❖ How good is \hat{y} as compared to a random forest trained with a dataset made of expert-handcrafted features?
- ❖ **Flow-level features**, such as: traffic throughput, packet sizes, inter-arrival times, frequency of IP addresses and ports, transport protocols and share of specific flags (e.g., SYN packets).

Domain knowledge vs. raw inputs

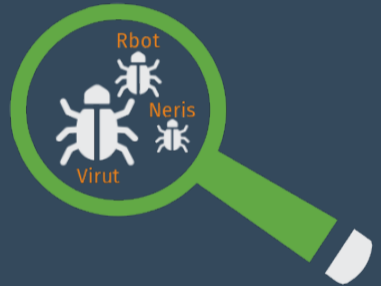
- ❖ How good is \hat{y} as compared to a random forest trained with a dataset made of expert-handcrafted features?
- ❖ **Flow-level features**, such as: traffic throughput, packet sizes, inter-arrival times, frequency of IP addresses and ports, transport protocols and share of specific flags (e.g., SYN packets).
- ❖ 200 of these features were built to feed a random forest model.

- ❖ The random forest model using expert domain features achieves highly accurate detection performance: 97% with FPR less than 1%.

- ❖ The random forest model using expert domain features achieves highly accurate detection performance: 97% with FPR less than 1%.
- ❖ The deep learning-based model using the \hat{u} o still outperforms this domain expert knowledge based detector.

- ❖ The random forest model using expert domain features achieves highly accurate detection performance: 97% with FPR less than 1%.
- ❖ The deep learning-based model using the \hat{u} o still outperforms this domain expert knowledge based detector.
- ❖ The deep learning model can perform as good as a more traditional shallow-model based detector for detection of malware flows, without requiring any sort of expert handcrafted inputs.

o ¶ D
¶ ;
ò



Conclusions

- ❖ This work explores the power of deep learning models to the analysis of network traffic measurements.

- ❖ This work explores the power of deep learning models to the analysis of network traffic measurements.
- ❖ The specific problem of malware network traffic detection and classification is addressed using representations of the input network data.

- ❖ This work explores the power of deep learning models to the analysis of network traffic measurements.
- ❖ The specific problem of malware network traffic detection and classification is addressed using representations of the input network data.
- ❖ Using \vec{v} as input for the deep learning models achieves better results than using \vec{v}^T .

- ❖ This work explores **the power of deep learning models to the analysis of network traffic measurements.**
- ❖ The specific problem of **malware network traffic detection and classification is addressed using** representations of the input network data.
- ❖ Using **as input for the deep learning models achieves better results** than using **.**
- ❖ **In all studied cases, the deep learning models outperform a strong random forest model,** using exactly the same input features.

- ❖ This work explores the power of deep learning models to the analysis of network traffic measurements.
- ❖ The specific problem of malware network traffic detection and classification is addressed using representations of the input network data.
- ❖ Using $\vec{u} \circ \vec{o}$ as input for the deep learning models achieves better results than using $\vec{u} \circ \vec{d}$.
- ❖ In all studied cases, the deep learning models outperform a strong random forest model, using exactly the same input features.
- ❖ The $\vec{u} \circ \vec{o}$ architecture slightly outperforms a random forest model trained using expert domain knowledge features.

THANKS for your attention!

Questions?

 @stillyawning

 gonzalo@tryolabs.com