Characterizing Intrusion Detection Systems On Heterogeneous Embedded Platforms

Camélia Slimani, Louis Morge-Rollet, Laurent Lemarchand, Frédéric Le Roy, David Espes, Jalil Boukhobza

September 8, 2023

26th Euromicro Conference on Digital System Design

Table of contents

Context and Presentation of DISPEED Project

Context

Swarms of drones advent

- Swarms of drones are supposed to gain more autonomy and efficiency during their mission (ex. coastal and port inspection) [8].
- Security threats and low energy can disrupt mission progression.

Context

Swarms of drones advent

- Swarms of drones are supposed to gain more autonomy and efficiency during their mission (ex. coastal and port inspection) [8].
- Security threats and low energy can disrupt mission progression.

Intrusion Detection Systems

- Software or hardware system that identifies suspicious actions on a computer system to maintain security [4].
- Modern IDS rely on Machine Learning techniques that are resource hungry [3].

Context

Swarms of drones advent

- Swarms of drones are supposed to gain more autonomy and efficiency during their mission (ex. coastal and port inspection) [8].
- Security threats and low energy can disrupt mission progression.

Intrusion Detection Systems

- Software or hardware system that identifies suspicious actions on a computer system to maintain security [4].
- Modern IDS rely on Machine Learning techniques that are resource hungry [3].

Drones can be equipped with heterogeneous computing (GPU, CPU, DLA, FPGA) [3] and memory capabilities (DRAM, NVMs).

Problem Statement

How to leverage heterogeneous drone resources to achieve a trade-off between energy, efficiency and security ?

DISPEED Project

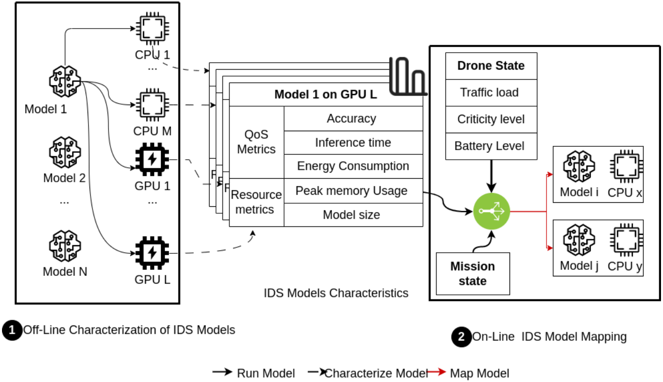
Problem Statement

How to leverage heterogeneous drone resources to achieve a trade-off between energy, efficiency and security ?

Challenges

- How to characterize IDS models on a drone ?
- How to map IDS on a single drone given mission and drone states ?
- How to distribute IDS given a swarm of drones ?

DISPEED Project milestones

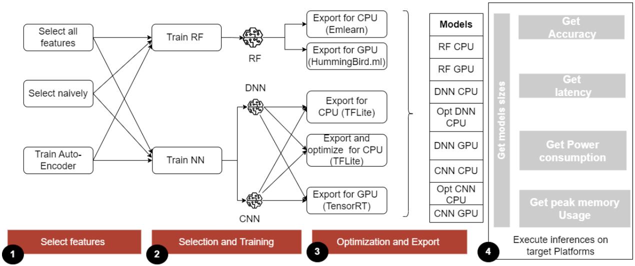


IDS Models Characterization

IDS Models Characterization Methodology

Data-set: UNSW-NB15 [5] (normal and malicious traffic).

1. **Features selection/extraction:** Several feature selection techniques were tried.
2. **Selection and Training:** Random Forest IDS models, Neural Networks IDS models.
3. **Optimization and export:** IDS models are exported and optimized for the target platforms.
4. **Execute Inferences on target platforms:** measure the inference latency, accuracy, power consumption and memory peak.



Feature Selection

1. No feature Selection (NoFS) : select all features.

Feature Selection

1. No feature Selection (NoFS) : select all features.
2. Expert Selection (ES) : select a subset of features based on a state-of-the art work [9].

Feature Selection

1. No feature Selection (NoFS) : select all features.
2. Expert Selection (ES) : select a subset of features based on a state-of-the art work [9].
3. Auto-Encoder : use an auto-encoder to extract a new representation of traffic data.

Layer	Parameters		
	Neurones	Kernel Initializer	Activation
Dense	153	Glorot Uniforme	ReLu
Dense	121	Glorot Uniforme	ReLu
Dense	89	Glorot Uniforme	ReLu
Dense	57	Glorot Uniforme	ReLu
Dense	25	Glorot Uniforme	Sigmoid
Dense	57	Glorot Uniforme	ReLu
Dense	89	Glorot Uniforme	ReLu
Dense	121	Glorot Uniforme	ReLu
Dense	153	Glorot Uniforme	ReLu

IDS models

- Random Forests : 50 trees with a maximum depth of number of features.
- Dense Neural Networks : Two DNN architectures

	Neurones	Kernel Initializer	Activation
Dense	128	Glorot Uniforme	ReLu
Dropout	0.5		
Dense	64	Glorot Uniforme	ReLu
Dense	32	Glorot Uniforme	ReLu
Dense	10	Glorot Uniforme	Softmax

	Neurones	Kernel Initializer	Activation
Dense	1024	Glorot Uniforme	ReLu
Dense	704	Glorot Uniforme	ReLu
Dense	288	Glorot Uniforme	ReLu
Dense	64	Glorot Uniforme	ReLu
Dense	10	Glorot Uniforme	ReLu

- Convolution Neural Networks : a SOTA architecture

Layer	Parameters		
	Filters	Filter size	Activation
Conv2D	64	(3,1)	ReLu
Conv2D	64	(3,1)	ReLu
MaxPool	(2,1)		
Conv2D	256	(3,1)	ReLu
Conv2D	256	(3,1)	ReLu
Conv2D	256	(3,1)	ReLu
MaxPool	(2,1)		
Flatten			
	Neurones	Kernel Initializer	Activation
Dense	100	Normal	ReLu
Dropout	0.5		
Dense	20	Normal	Relu
Dense	10	Normal	Softmax

Optimize and Export

Used platforms

- CPU : RaspBerry Pi 4
- GPU : Nvidia Xavier AGX

Used Frameworks

- Random Forests: Emlearn [7] for CPU and HummingBird.ml [6] for GPU.
- Neural Networks: TFLite [2] for CPU and TensorRT [1] for GPU.

Execute Inferences on target platforms

QoS metrics

- Accuracy $\rightarrow$ Security
- Inference time $\rightarrow$ Performance
- Energy Consumption $\rightarrow$ Energy

Resource metrics

- Peak memory usage
- Model size

Results and Findings

Results and Findings

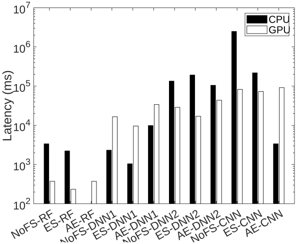
Accuracy and F1-Score

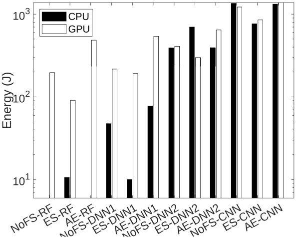
Feature Select.-Model	Accuracy (%)	Weighted-avg F1-Score (%)
NoFS-RF	81.59	82.83
ES-RF	77.28	79.5
AE-RF	75.26	77.73
NoFS-DNN 1	76.5	78.64
ES-DNN 1	70.34	74.26
AE-DNN 1	74.9	77.25
NoFS-DNN 2	80.94	80.21
ES-DNN 2	75.02	75.29
AE-DNN 2	79.03	81.96
NoFS-CNN	76.8	78.5
ES-CNN	74.09	78.03
AE-CNN	74.45	77.65

Size of the models

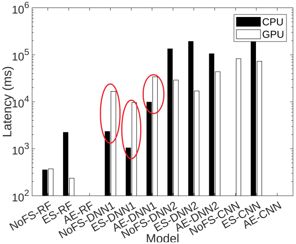
Model	Exported Model Size on CPU (MB)	Exported Model Size on GPU (MB)
NoFS-RF	28	15.4
ES-RF	9.1	5.5
AE-RF	/	32.9
NoFS + DNN 1	0.144	0.144
ES + DNN 1	0.053	0.053
AE + DNN 1	0.321	0.321
NoFS + DNN 2	3.33	3.33
ES + DNN 2	2.61	2.61
AE + DNN 2	2.96	2.96
NoFS-CNN	4.77	4.77
ES-CNN	2.6	2.6
AE-CNN	2.9	2.9

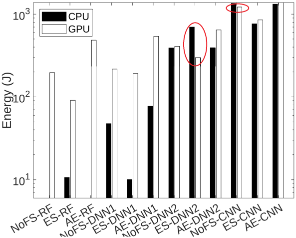
Results and Findings





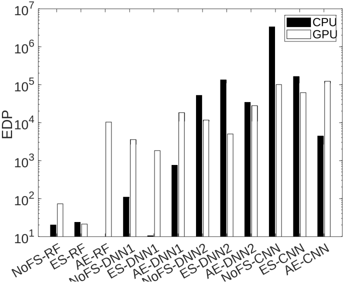
Results and Findings

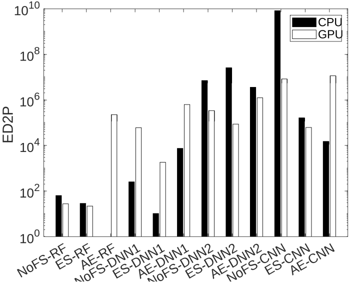




Findings

- For DNN1 (shallow model), inference on CPU is faster than GPU.
- When the GPU inference speed surpasses the CPU by several orders of magnitude, it is more energy efficient than the CPU for certain models.
- Random Forests show good performance on CPU and GPU on several metrics.





Findings

- For deep neuronal networks, GPU is more energy efficient than CPU.

Conclusion and Future Works

Conclusion and Future Works

Conclusion

- This first step of the project consisted in forming space search of IDS models;
- We defined the metrics that will allow us to choose the model to use one-line.

Future Works

- Other IDS models and embedded platforms will be considered in the characterization work ;
- We will consider the heterogeneity of the memory component in the characterization ;
- We will work on a multi-objective IDS mapping strategy.

References i

Nvidia tensorrt.
https://docs.nvidia.com/deeplearning/tensorrt/.

Tensorflow lite.
https://www.tensorflow.org/lite/guide.

M. Aissi, Y. Moumen, J. Berrich, T. Bouchentouf, M. Bourhaleb, and M. Rahmoun.
Autonomous solar usv with an automated launch and recovery system for uav: State of the art and design.
In *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, pages 1–6, 2020.

References ii

A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman.
Survey of intrusion detection systems: techniques, datasets and challenges.
Cybersecurity, 2(1):1–22, 2019.

N. Moustafa and al.
Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set).
In *2015 military communications and information systems conference*.

S. Nakandala, K. Saur, G.-I. Yu, K. Karanasos, C. Curino, M. Weimer, and M. Interlandi.
A tensor compiler for unified machine learning prediction serving.
In *14th USENIX Symposium on Operating Systems Design and Implementation (OSDI 20)*, pages 899–917, 2020.

J. Nordby.
emlearn: Machine Learning inference engine for Microcontrollers and Embedded Devices, Mar. 2019.

P. Svec and S. K. Gupta.
Automated synthesis of action selection policies for unmanned vehicles operating in adverse environments.
Autonomous Robots, 32(2):149–164, 2012.

A. Valero León.
Insides: A new machine learning-based intrusion detection system.
2017.