Leveraging Large Language Models for Fileless Malware Detection at the Edge

Author: Redon Kokaj
Supervisor: Prof. Elena Ferrari
Co-supervisor: Dr. Christian Rondanini

MSc in Computer Science
University of Insubria
Academic Year 2024/2025

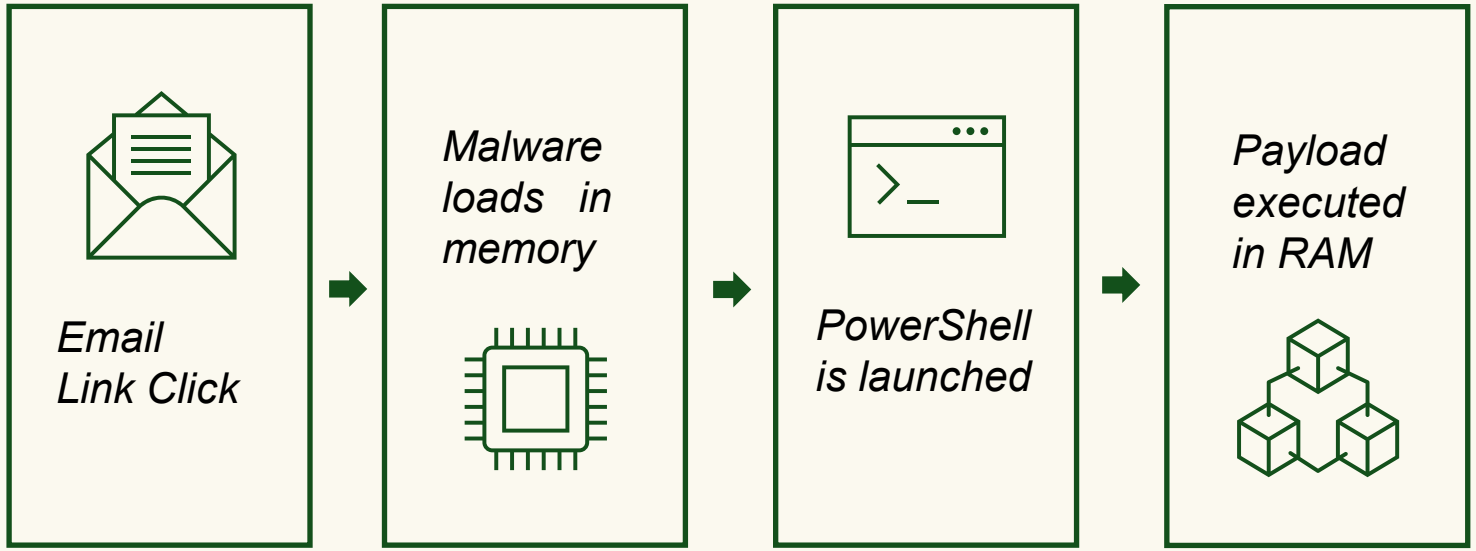
10/12/2025

Index

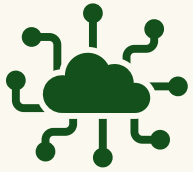
The New Battlefield: Invisible Threats in the Machine

Unlike traditional malware, fileless attacks:

- Operate **entirely in memory**.
- Use the system's own tools against it (*living off the land)*.
- 10 times more likely to succeed.
- Traditional antivirus software is often **blind to this threat**.



Fileless Malware Characteristics



In-Memory Execution

Malicious code is injected directly in RAM.

Registry Persistence

Payload is hidden in registry keys to be executed on startup.

Living-Off-The-Land

Legitimate system tools are exploited (e.g., PowerShell, WMI).

Macro Exploitation

Malicious code is injected in documents (e.g., Office) and executed when user activates macros.

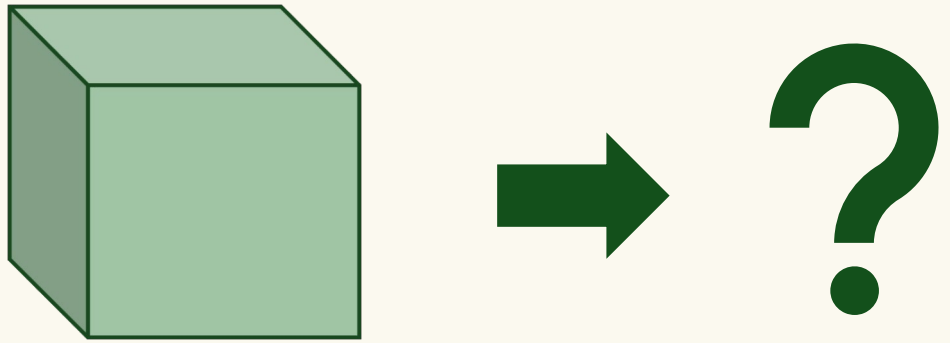
Large Language Models: Powerful but Heavy

The Features:

- LLMs excel at recognizing complex patterns in sequential data.
- We can treat memory as '*language*' to differentiate benign from malicious behavior.

The Problem:

- State-of-the-art models are massive.
- Too computationally expensive and slow for real-time detection on edge devices.



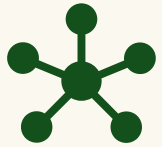
Full-Scale LLM (e.g., BERT)

High Power, High Latency

Edge-Ready Model

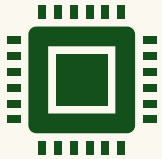
High efficiency, Low Latency

Core Challenges



Limited Data

How do we train a model without enough examples?

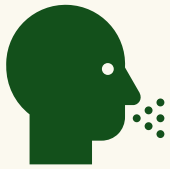


Resource Constraints

How do we ensure low-latency detection without sacrificing accuracy?

Dynamic Threats

How do we build a model that generalizes to unseen threats?



Overfitting & Bias

How do we prevent the model from learning false correlations while training?

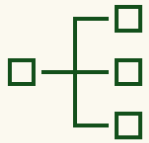
Our Objective: Intelligent Detection at the Edge

How can we detect these stealthy threats on edge where computational resources are limited?

Thesis Goal: design and validate a lightweight detection system using Large Language Models (LLMs) that can:

1. 

Identify faint patterns of malicious behavior in memory artifacts.

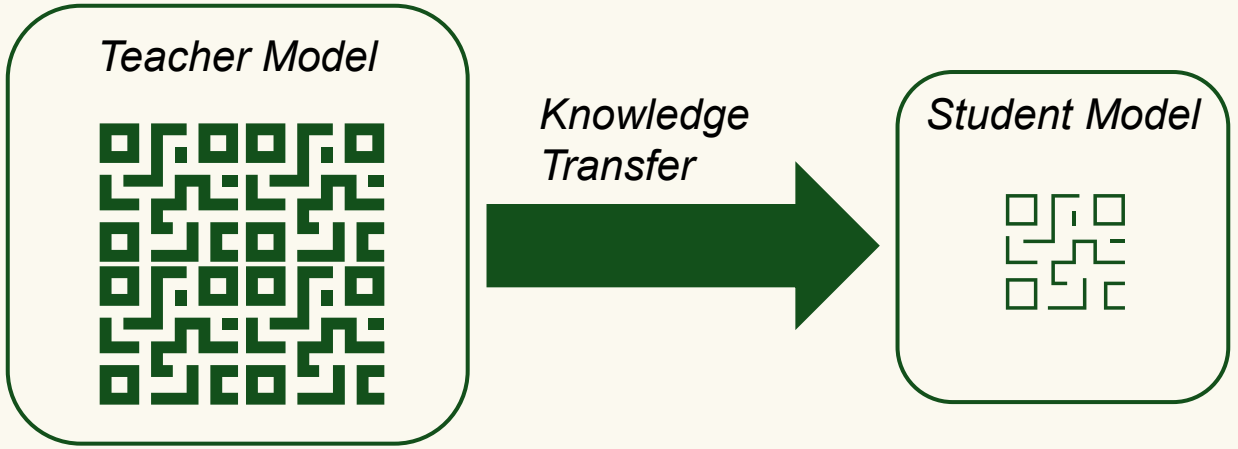
2. 

Generalize to new, unseen fileless attack patterns.

3. 

Operate efficiently on resource-constrained edge hardware.

The Distillation Process



Core Concept:

A large, pre-trained ‘Teacher’ model transfers its knowledge to a smaller, faster ‘Student’ model.

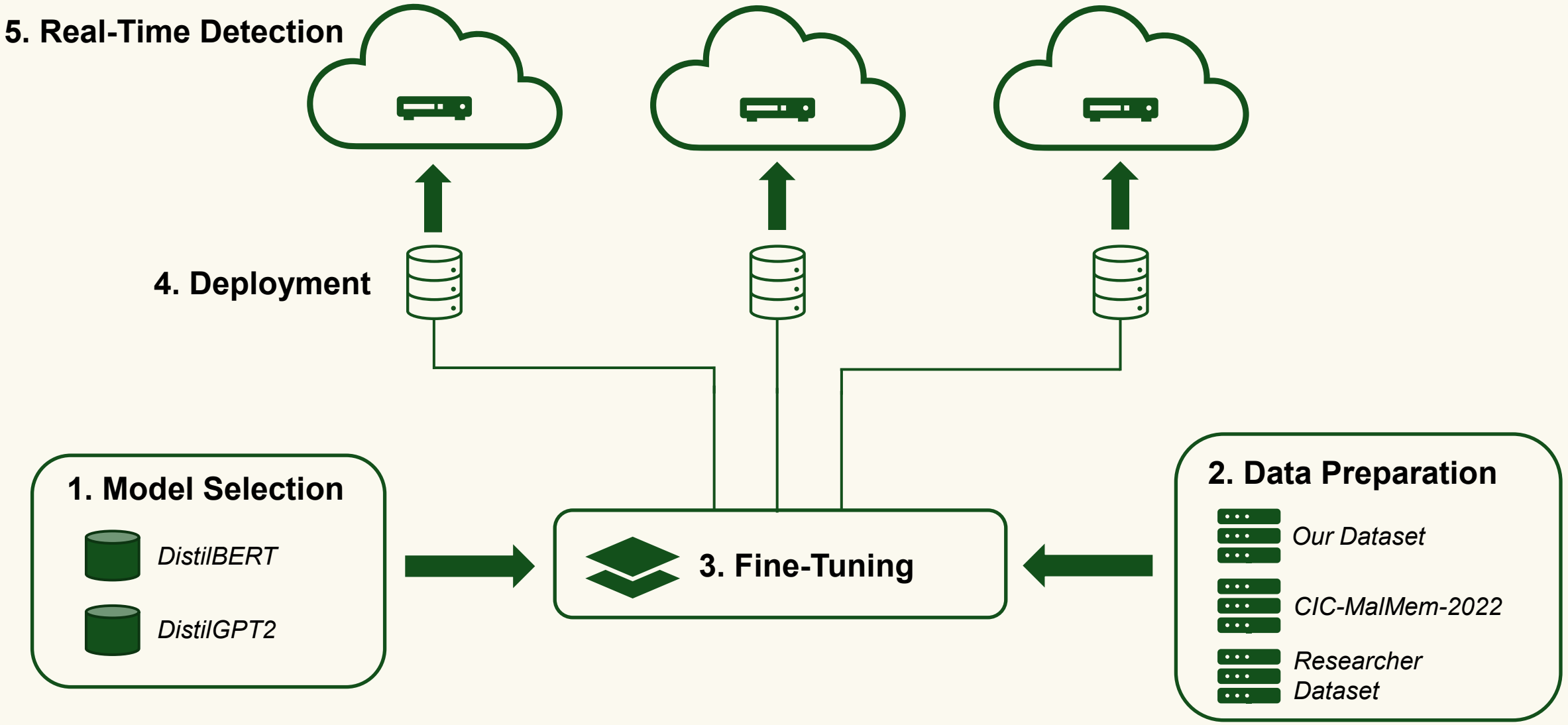
The Process:

The student learns to mimic the teacher’s decision-making process.

Outcome:

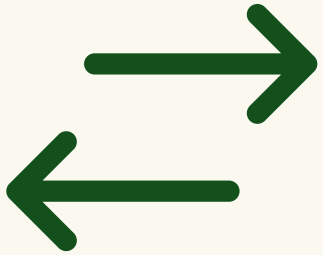
We get a compact model that retains much of the accuracy of its larger counterpart.

A Modular Architecture for Edge-Based Detection



Chosen Models

Two architectures, One goal: We selected two powerful, distilled models to compare their effectiveness.



Model: DistilBERT
Type: Encoder-only (Bidirectional)
Strength: Excellent for classification tasks requiring deep contextual understanding of the entire data sequence.
Stats: 66M parameters, 40% smaller and 60% faster than BERT.

Model: DistilGPT2
Type: Decoder-only (Autoregressive)
Strength: Excels at understanding sequential dependencies and patterns over time.
Stats: 82M parameters, 35% smaller and 50% faster than GPT-2.

Dataset Research and Selection

High-quality, public datasets for fileless malware are extremely rare due to privacy and security concerns.

1. Comprehensive Search

- Researched through academic literature.
- Found one dataset.
- Many other datasets were proprietary.

2. Direct Outreach

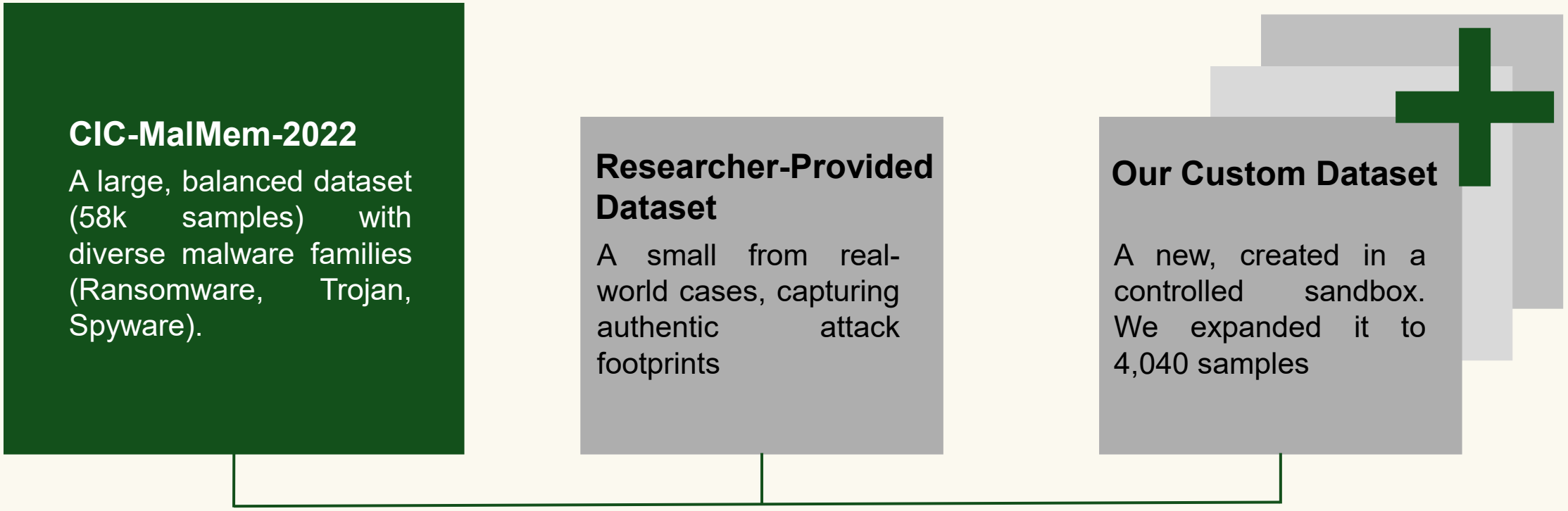
- Contacted multiple research groups.
- One responded, generously sharing their dataset.

3. In-house Creation

- We built a dataset from scratch.
- Created in a controlled sandbox.

This comprehensive effort was essential to build a credible foundation.

The Challenge of Realistic Data



All datasets were harmonized, aligning features extracted via the Volatility framework to create a consistent base for evaluation.

The Experiments

1. Classical Baselines
Does our approach outperform traditional methods? (*RF, KNN*)

2. Cross-Validation Strategy
How robust are the models against data leakage? (*KFold vs GroupKFold*)

3. Data Scarcity
Can the models learn with little data? (*Few Shot & Zero Shot*)

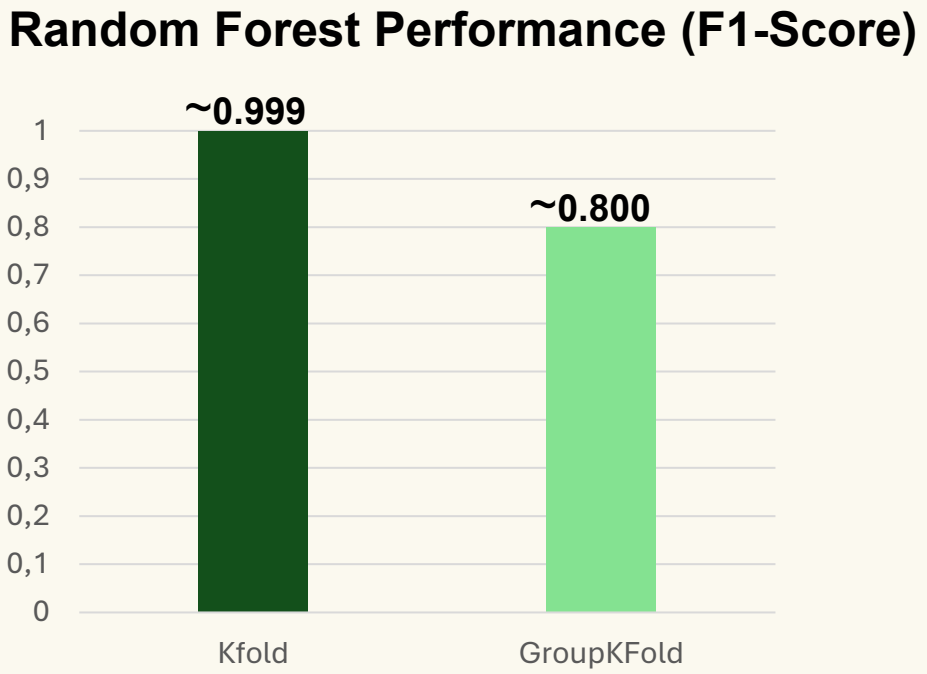
4. Imbalance Resilience
How do they perform when malicious samples are rare? (*Proportion Variations*)

5. Cross-Dataset Generalization
Can models trained in one environment detect threats in a completely different one?

The first Insight: Realistic vs. Inflated Scores

Key Finding: Both classical models (RF, KNN) and our Transformer models achieve near-perfect scores with standard KFold validation.

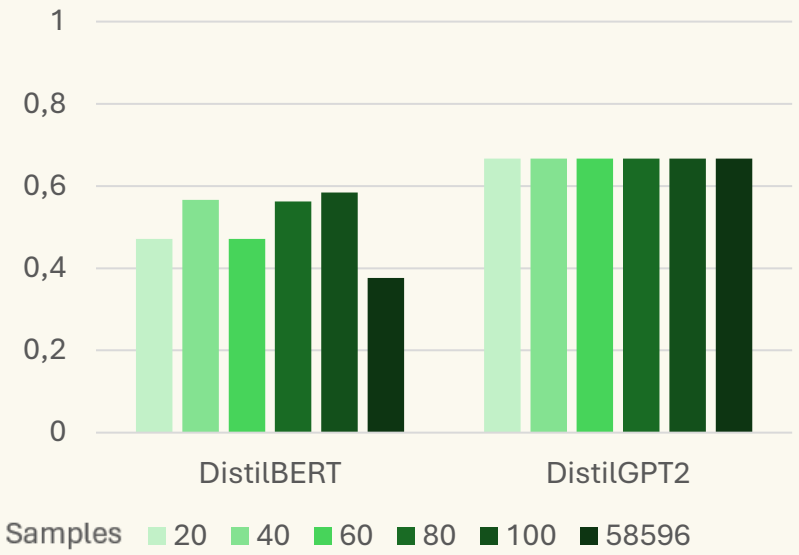
The Reality Check: When switching to the more rigorous GroupKFold, the performance of *all* models drop significantly.



True performance is measured by how well a model generalizes to truly unseen threats.

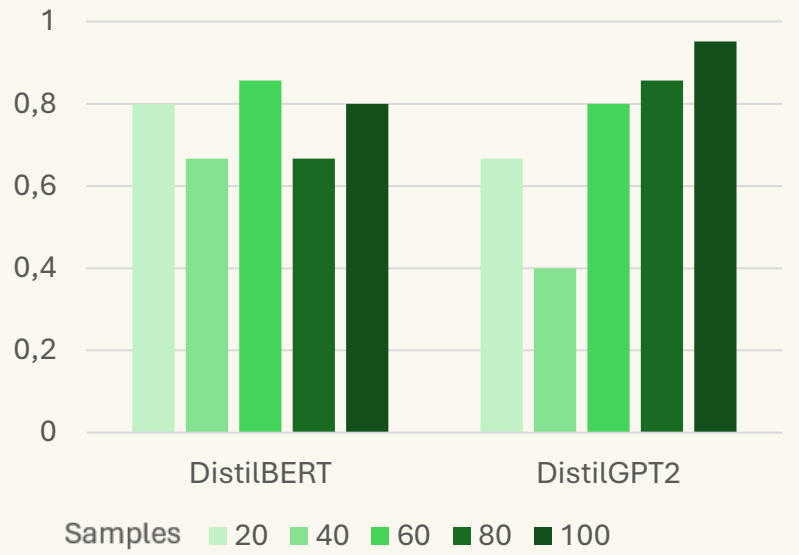
Adaptability Testing: Learning with Little Data

Zero Shot (F1-Score)



DistilBERT showcases modest results.
DistilGPT2 defaults to one output.

Few Shot (F1-Score)

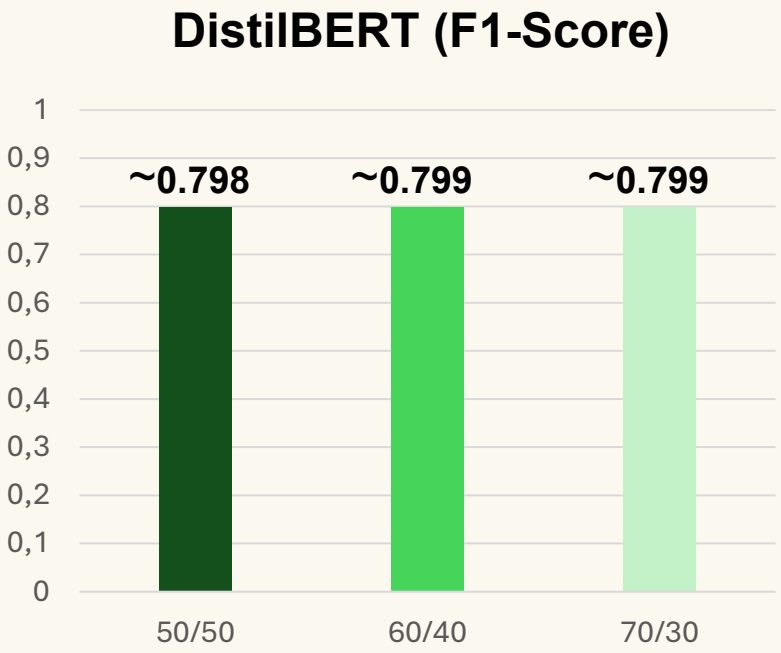


DistilBERT achieves good results overall.
DistilGPT2 learns rapidly.

DistilBERT demonstrates higher stability when data is scarce.

Imbalance Testing: Differences with Proportions

Does performance change when malicious samples get rarer?



The Test:
We tested on datasets with 60/40 and 70/30 splits.

The Results:
Performance remained stable under GroupKFold.

The models maintain high performance even as threats get less frequent.

Ultimate Test: Generalizing Across Unseen Environments

We trained models on one dataset and tested them on a completely different one.

DistilBERT (F1-Score)

Train \ Test	CIC-MalMem-2022	Our Dataset	Researcher Dataset
CIC-MalMem-2022		0.632	0.500
Our Dataset	0.710		0.528
Researcher Dataset	0.652	0.577	

DistilBERT provided higher recall and stability in most scenarios.

DistilGPT2 (F1-Score)

Train \ Test	CIC-MalMem-2022	Our Dataset	Researcher Dataset
CIC-MalMem-2022		0.565	0.690
Our Dataset	0.652		0.542
Researcher Dataset	0.836	0.573	

DistilGPT2 achieved higher peak performance in specific scenarios.

DistilBERT appears to be more reliable for generalization overall.

Final Considerations

1.

Validated Lightweight LLMs for Edge Detection.

Distilled transformers models are effective for fileless malware detection at the edge.

2.

Proved Superiority for Generalization.

Encoder-based models are superior for this specific task.

3.

Created a Reusable Research Asset.

Developed and harmonized a new, fileless malware specific dataset.

Future Directions

Continuous Learning & Adaptation: Move from static periodical retraining to dynamically learning in near real-time.
Advanced Robustness: Proactively research and defend against defense techniques.