GRAMMATECH

PE32 Malware Detection Using Features from Static Disassembly

Akshay Sood

Katherine Leffel · Junghee Lim

HCSS 2026

Malware Detection

Scale.

~2M new files/day on VirusTotal · ~500K estimated malicious daily

ML is central to malware detection.

Widely-adopted public benchmarks saturate on parsing-based features.

Parsing does not capture deeper properties of the binary.

Can structural features from static disassembly provide complementary signals?

Research Questions

1

Can features extracted at scale from static disassembly meaningfully improve malware detection on PE32 Windows binaries?

2

What do the trained models reveal about the salient properties of malicious binaries?

Overview

Goals

Can we use static disassembly using DDisasm to improve malware detection for PE32 Windows binaries and learn new insights about malware?

Approach

Layer static-disassembly features (DDisasm) on top of parsing features and capability-rule features (CAPA). Train models, perform feature ablation, analyze feature importance.

Findings

Disassembly features add signal beyond parsing alone and contribute to robustness in cross-distribution evaluation. Disassembly quality itself is a malware tell. A small fraction of the full feature set captures significant predictive power.

DDisasm

Datalog Disassembly (2020)

Parse PE file format → decode a superset of possible instructions → generate an initial set of Datalog facts.

Declarative analysis identifies code locations, symbolization, and function boundaries.

Refined facts are translated to GTIRB — GrammaTech's intermediate representation for binaries.

GTIRB can be used for downstream binary analysis, transformation, and reassembly.

Flores-Montoya & Schulte. Datalog Disassembly. USENIX Security 2020.

Disassembly as Weighted Interval Scheduling with Learned Weights (2025)

Conflict resolution among candidate instructions is reduced to a WIS optimization problem; heuristic weights are learned from ground-truth-annotated binaries.

Flores-Montoya, Lim, Seitz, Sood, Raff, Holt. IEEE S&P 2025.

Feature Groups

Parsing

PE header metadata · imports & exports · section attributes · byte and entropy histograms · string statistics.

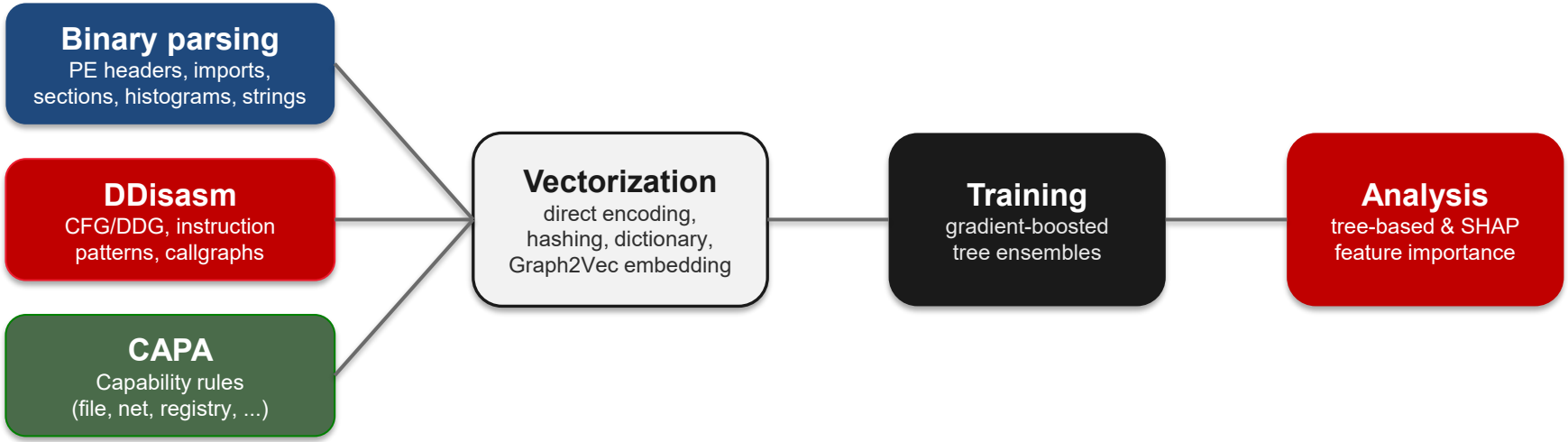
DDisasm

Control-flow and data-dependency graph properties · instruction-type ratios · symbolization and disassembly quality metrics · referenced strings · callgraph embeddings

CAPA

Capability labels from Mandiant's rule engine. Each matched rule contributes one feature (file I/O, network, registry, cryptography, packer/compiler detection, ...).

Pipeline



Datasets

EMBER2018 (Anderson & Roth, 2018)

Benchmark PE32 dataset.

500K PE32 binaries after filtering · 80/20 temporal split · balanced benign and malware

VTPipeline

Daily VirusTotal samples since Sept 2023.

50K PE32 binaries · ~10% benign / 90% malicious

EMBER2024 (Joyce et al., 2025)

Successor to EMBER2018.

720K train / 180K test

EMBER2018 Feature Ablation

Parsing	DDisasm	CAPA	EMBER2018
✓	✓	✓	0.994
✓	✓	–	0.993
✓	–	✓	0.992
–	✓	✓	0.986
✓	–	–	0.990
–	✓	–	0.980
–	–	✓	0.918

Cross-Dataset Generalization

Parsing	DDisasm	CAPA	EMBER2018	VTPipeline
✓	✓	✓	0.994	0.978
✓	✓	–	0.993	0.972
✓	–	✓	0.992	0.929
–	✓	✓	0.986	0.942
✓	–	–	0.990	0.937
–	✓	–	0.980	0.928
–	–	✓	0.918	0.897

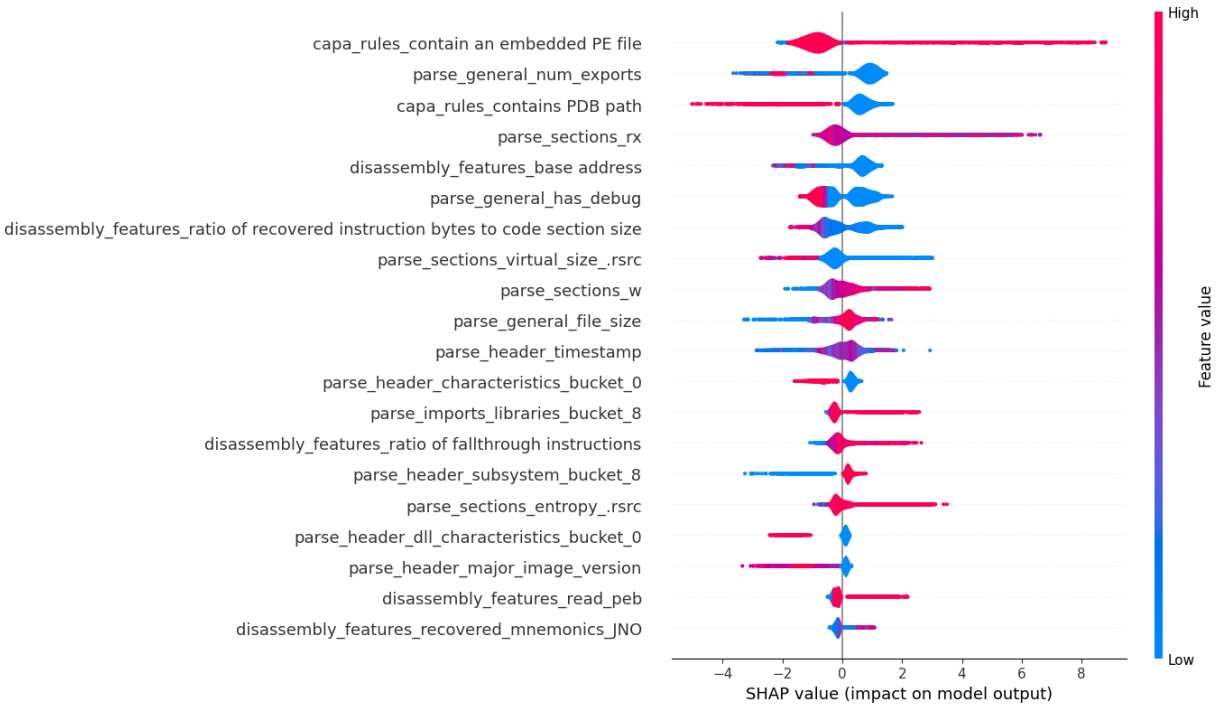
Feature decomposition pattern holds across independent dataset

Cross-Dataset Generalization

Parsing	DDisasm	CAPA	EMBER2018	VTPipeline	Δ
✓	✓	✓	0.994	0.978	0.016
✓	✓	–	0.993	0.972	0.021
✓	–	✓	0.992	0.929	**0.063**
–	✓	✓	0.986	0.942	0.044
✓	–	–	0.990	0.937	0.053
–	✓	–	0.980	0.928	0.052
–	–	✓	0.918	0.897	0.023

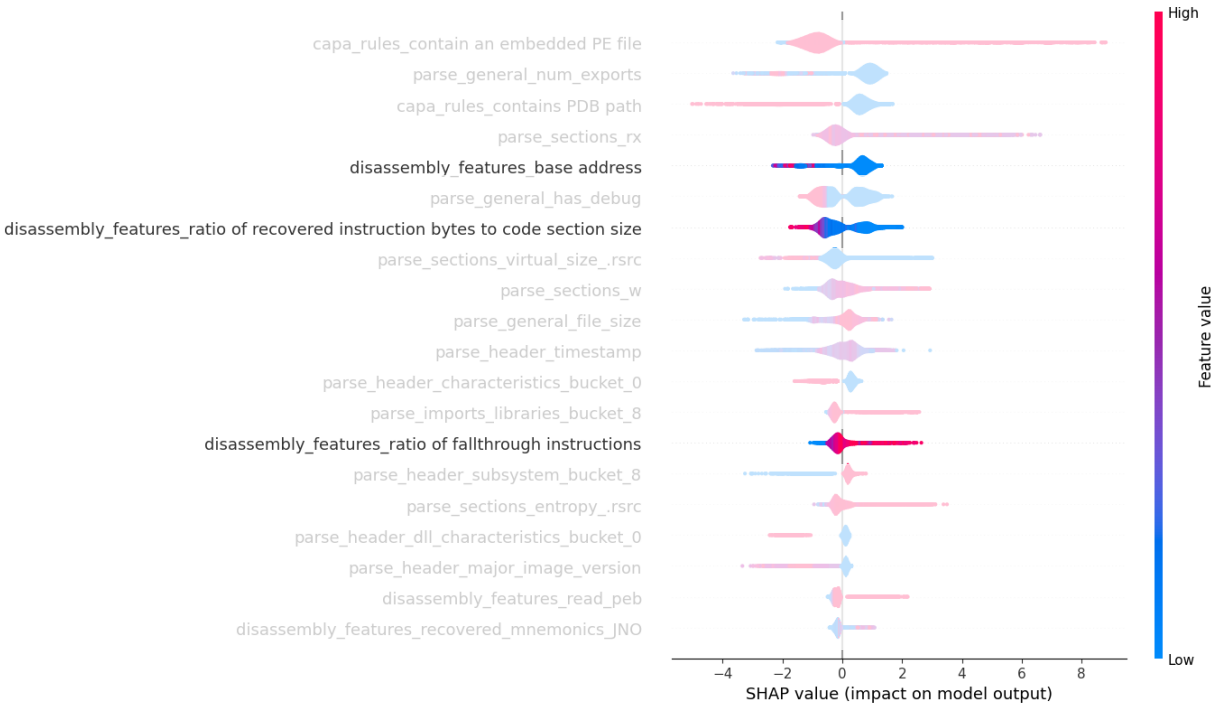
DDisasm-less configurations lose the most under cross-dataset evaluation

Top-Ranked Features



Top-Ranked Features

Disassembly quality is predictive of malware



Several features measure how cleanly DDisasm recovered the binary's code.

Base address (#5)

Non-standard image bases correlate with malware.

Recovered instruction byte ratio (#7)

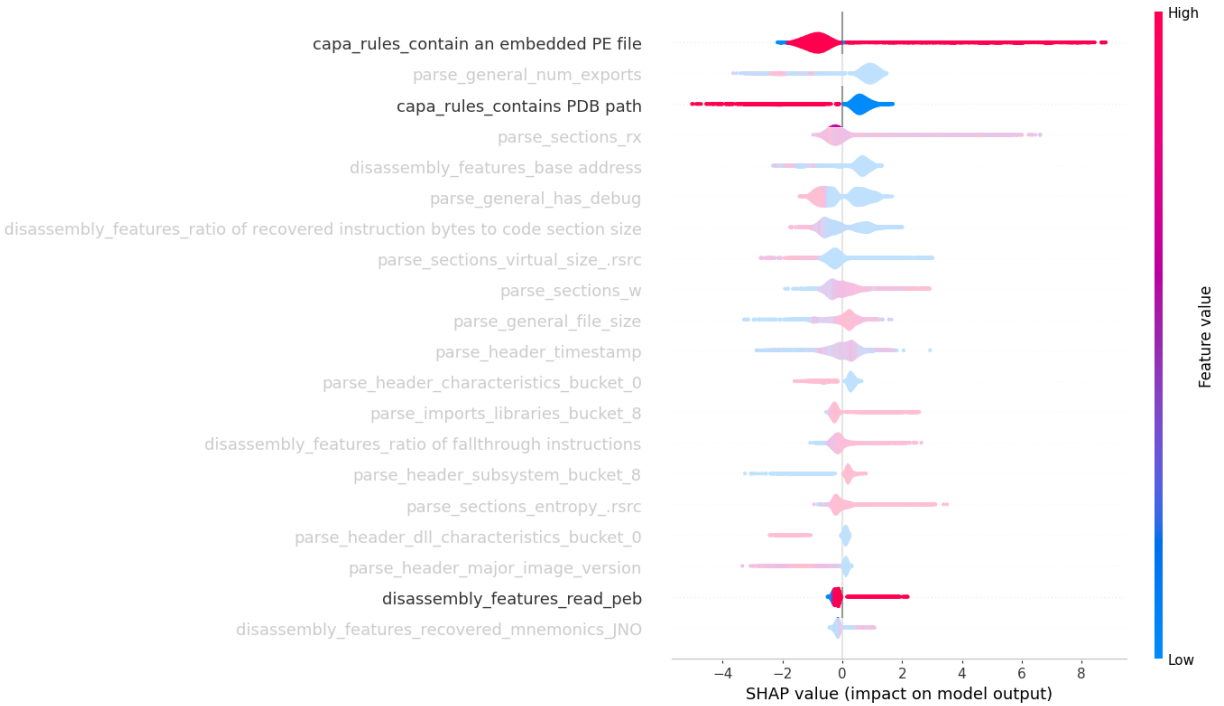
Fraction of code-section bytes DDisasm recovered as valid instructions.

Fallthrough instruction ratio (#14)

Fraction of instructions that proceed sequentially vs. branch.

Top-Ranked Features

Capability rules capture explicit malicious behaviors



CAPA's capability rule engine flags concrete malicious behaviors. Three are in the top 20.

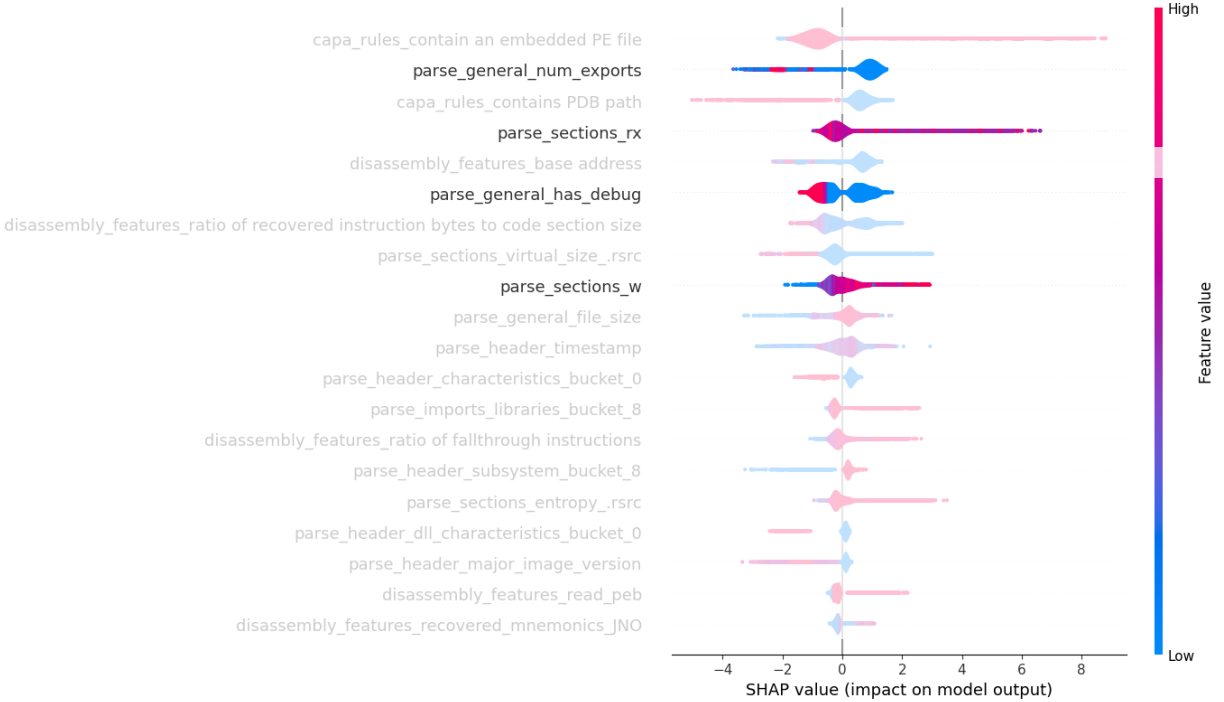
Embedded PE file (#1)
Droppers, packers, installers can carry their next stage as a PE inside the host.

Contains PDB path (#3)
Benign software more likely to retain PDB path for debugging and crash reporting.

read_peb (#19)
References to the Windows Process Environment Block.

Top-Ranked Features

Build provenance reveals benign vs. malware origins



Several top parsing-based features describe how the binary was built rather than what it does.

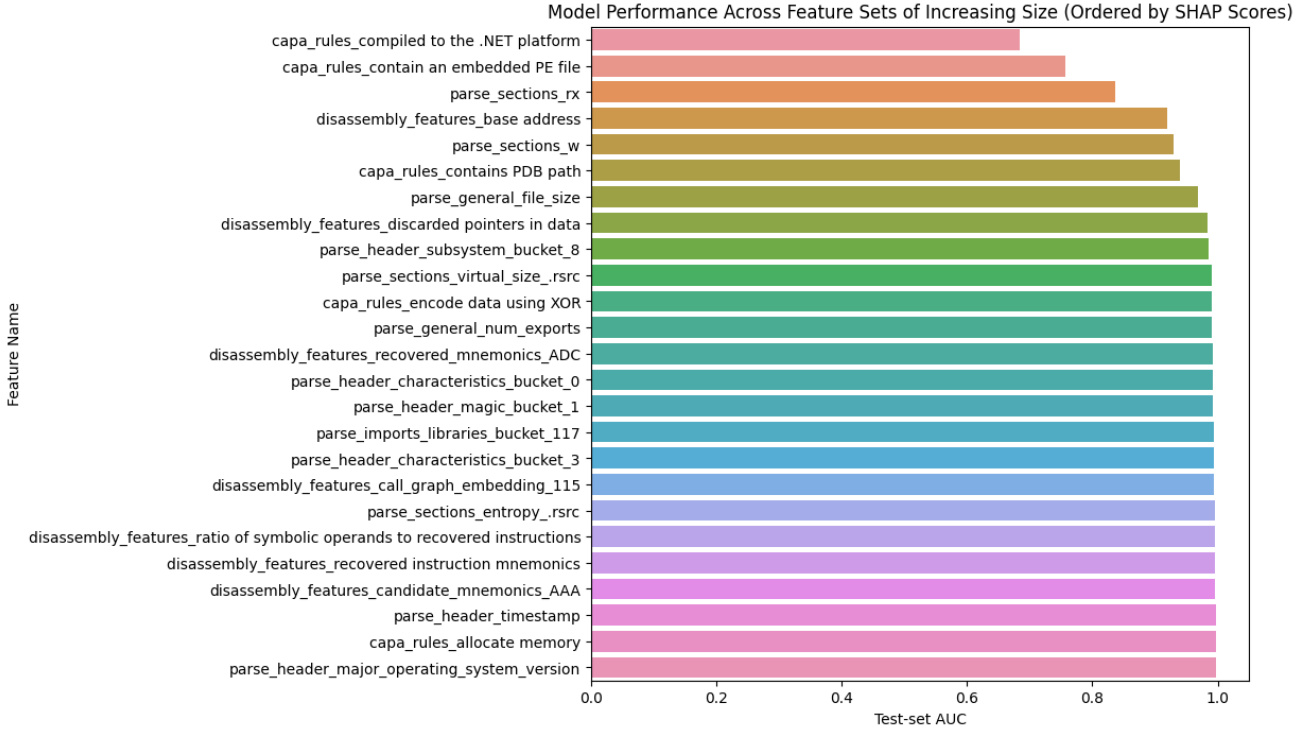
num_exports (#2)

sections_rx (#4)
Count of executable sections.

has_debug (#6)

sections_w (#9)
Count of writable sections.

A Small Feature Set Suffices



A small fraction of the full feature set captures much of the predictive signal.

Research Directions

Dynamic Analysis using TBDisasm

Challenge

Static disassembly recovers the unpacker stub but cannot reach the payload, which is only decompressed at runtime. This is a fundamental limitation of static analysis, applicable to any static disassembler.

Approach

Integrate TBDisasm, GrammaTech's trace-based disassembler. It captures execution traces of unpacked payloads at runtime and emits memory dumps in the GTIRB intermediate representation.

Pipeline: binary → TBDisasm (dynamic) → GTIRB → DDisasm (static) → features

Also developing anti-analysis countermeasures to identify and counter common evasion techniques — timing checks, anti-debug, anti-VM — that packers use to detect instrumentation.

Research Directions

Agent-driven Cost-aware Malware Triage Pipelines

Takeaways

1. Complementary signals.

Parsing, DDisasm and CAPA features contribute distinct information. Combining all three delivers better performance than any one feature group.

2. Disassembly quality is itself predictive.

Top SHAP-ranked features frequently reflect how cleanly a binary disassembles. Malware tends to be structurally harder to analyze even when not packed.

3. Small feature sets go far.

A small fraction of the full feature set captures most of the predictive signal. Useful for efficient deployment and interpretability.

4. DDisasm contributes to cross-dataset robustness.

When evaluating on independent collections, configurations without DDisasm lose the most AUC.