What features do Convolutional Neural Networks use to classify packed malware?

Tristan Rietveldt t.rietveldt@student.tudelft.nl

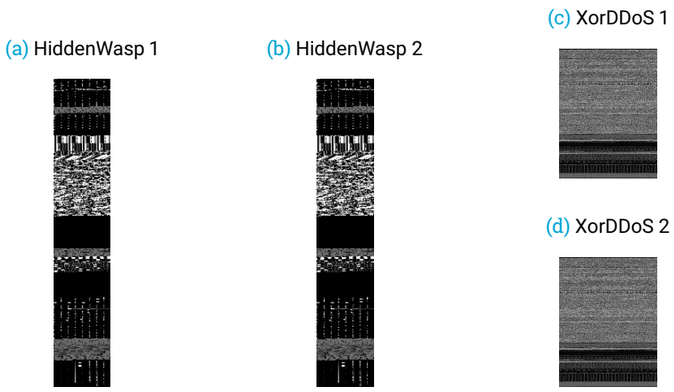
CSE3000 Research Project 2026/Q4 — Supervisors: Tom Viering, Akash Amalan

1 Background

- Traditional malware detection relies on slow, manual analysis
- No longer sustainable: too many new variants

Grayscale image visualisation (Figure 1) produces **distinct, family-consistent visual patterns**

Figure 1: Visual comparison of malware binaries converted to images. The first two images demonstrate texture similarity between two distinct samples of HiddenWasp, which are distinct from two similar samples of XorDDoS displayed on the right.



However, most malware is **packed**: the original binary is **compressed, encrypted, or obfuscated** in a wrapper executable to hide its signature from antivirus software

CNN-based classification methods reached **very high accuracies**

- over 99% accuracy on unpacked datasets
- only slightly lower accuracy (ranging from 95-99%) on packed datasets

2 Research Question

(1) What is the impact of packing on CNN malware family classification, and (2) what do xAI methods reveal about the underlying features the models rely on?

3 Methodology

- **Performance Baseline (SQ1):** Train CNN on unpacked malware and packed malware, and compare
- **Feature Identification (SQ2):** Use ScoreCAM to generate visual heatmaps of important regions
- **Feature Validation (SQ3):** Occlude regions from SQ2 to validate their causal effects

TU Delft Delft University of Technology

4 Results

(SQ1) Packed classification performance

Classification performance **remains robust** (Figure 2):

- Almost no degradation for encryption and obfuscation (98%+ accuracy)
- Some degradation for compression (80-90% accuracy); however: only on large ELF files!

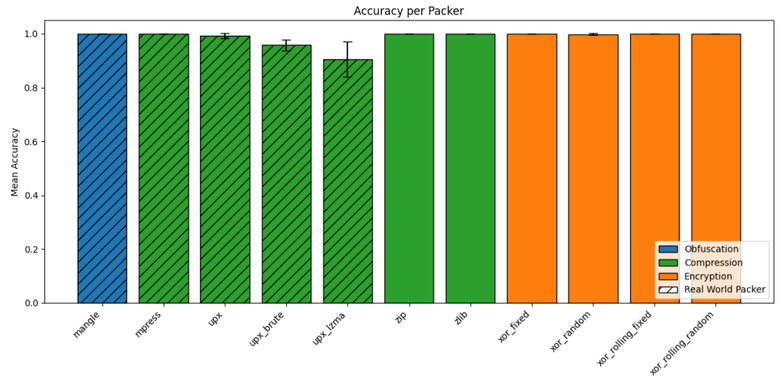


Figure 2: Mean accuracy of models trained on binary images packed with various packers

Cross-packer classification performance shows models are **somewhat packer-reliant**, cross-packer evaluation shows:

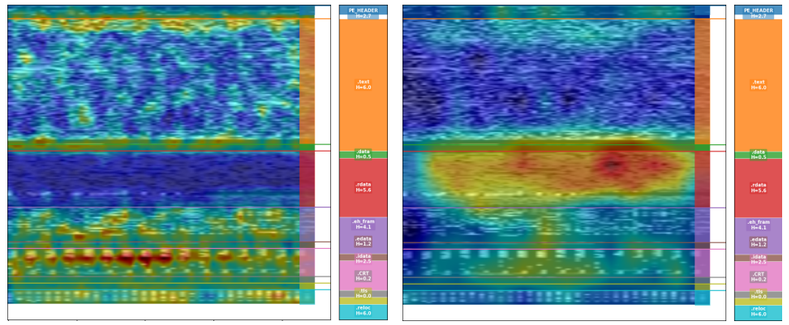
- Accuracy is high between UPX packers, XOR packers, and zip/zlib packers
- **Insight**: **Cross-packer generalisation only possible with similar structure**
- XOR packers and zip/zlib packers perform well on each other (70-90% accuracy)
- **Insight**: XOR and zip/zlib only obfuscate the Code section

(SQ2) ScoreCAM analysis

- ScoreCAM is a **gradient-free** (no backpropagation) visual explanation method for CNNs.
- ScoreCAM generates heatmaps that roughly show **what areas of an image the CNN pays attention to**

Different layers pay attention to **different file sections** (Figure 3):

- Attention heatmaps **relatively consistent** across unpacked, XOR, and zip/zlib packers. Attention from layer 1 to layer 4:
  - *PE_HEADER* decreases
  - *Code* stays the same
  - *Data* increases
  - *Linking & Memory* increases
  - *Runtime & Exceptions* increases



(a) Layer 1 CNN attention (b) Layer 3 CNN attention

Figure 3: Layer 1 and Layer 3 pay attention to different sections on an unpacked Adware sample

(SQ3) Occlusion analysis

We replaced areas of the image with **random noise** to see if they are **necessary**.

- *Code* section occlusion degrades performance less than random section occlusion
- **Insight**: CNNs **rely little on the *Code* section**
- XOR and zip/zlib have increased reliance on the *Data* section
- **Insight**: CNNs **shift their reliance** from the *Code* section to the *Data* section when *Code* is obfuscated
- Replacing *Code* with random noise only has impact for unpacked and fixed XOR.
- **Insight**: CNNs **learn encryption keys** with fixed-key encryption

Unexpected finding: ScoreCAM is **not fully accurate**

- ScoreCAM highlights the Code section **even when it is not necessary for prediction**

5 Limitations

- Dataset is synthetic
- Random horizontal flips were performed during training
- Resizing to 224 × 224 for classification and back for attention statistics (can warp large files)
- Random noise occlusion can be biased towards zip/zlib

6 Conclusions and Future Work

Conclusions:

- Models perform poorly on large files packed with UPX
- Models only generalise well across structure-preserving packers
- Models shift their reliance to unobfuscated sections when possible

Future Work:

- Investigate the discrepancy between ScoreCAM and occlusion
- Investigate effects of resizing on classification