TECHNICAL REPORT:

Implementation and Performance Evaluation of Opcode-Based Malware Classification Using SVM, KNN, and Decision Tree Algorithms

1. Introduction

Malware detection continues to be one of the biggest challenges in cybersecurity, especially with the rise of Advanced Persistent Threat (APT) groups that use custom-built and constantly evolving malicious software. Traditional antivirus systems rely on signatures or predefined rules, which often fail to recognize new or obfuscated malware variants.

To address this limitation, researchers are increasingly turning to machine learning (ML) for malware detection. ML models can learn the underlying behavior or structural patterns of malicious code and generalize to unseen samples.

Three machine learning algorithms were evaluated:

- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN) with k = 3.5
- Decision Tree (DT)

Each classifier was trained and tested using both 1-Gram (single opcode) and 2-Gram (pair of consecutive opcodes) feature sets. The goal is to understand how these models perform when the instruction sequence context becomes more complex.

2. Dataset and Feature Extraction

2.1 Dataset Overview

The dataset used in this project consists of opcode sequences extracted from both malicious samples (collected from several known APT campaigns) and benign programs (trusted executables from clean sources).

Each binary file was disassembled using static analysis tools such as IDA Pro or Ghidra to retrieve its opcode stream — a series of instructions like MOV, PUSH, JMP, or CALL.

2.2 Opcode N-Gram Representation

To convert raw opcode text into numerical data suitable for ML models, an n-gram feature extraction method was applied:

- 1-Gram (Unigram): Counts how often each unique opcode appears in a file.
- 2-Gram (Bigram): Captures sequential relationships between pairs of opcodes, which helps preserve the execution flow context.

The opcode frequencies were then normalized using TF-IDF weighting, reducing the influence of common but less informative instructions. This helped emphasize opcodes that are more discriminative for malware behavior.

3. Preprocessing Steps

Before training the models, several preprocessing steps were performed:

1. Opcode Extraction: Each binary was disassembled to obtain the raw opcode sequence.
2. Cleaning and Formatting: Operands, addresses, and comments were removed to keep only the instruction mnemonics.
3. Vectorization: The cleaned opcode data was transformed into TF-IDF feature vectors.
4. Normalization: All feature values were scaled using Min–Max normalization to maintain consistency across classifiers.
5. Dataset Splitting: Data was divided into 80% for training and 20% for testing, ensuring each class was equally represented in both sets.

4. Analysis

Three machine learning classifiers — Support Vector Machine (SVM), K-Nearest Neighbors (KNN) with

k = 3.5, and Decision Tree — were trained and tested on 1-Gram and 2-Gram opcode datasets extracted from malware samples belonging to various APT groups.

4.1 1-Gram Results

When using 1-Gram features (individual opcodes), overall performance was moderate. The Decision Tree classifier achieved the best results with an accuracy of 60%, outperforming KNN (50%) and SVM (30%). Most classifiers struggled to identify smaller or less distinctive APT families such as APT32 and FIN4, while Poseidon Group, Carbanak, and Equation were detected more reliably. Decision Tree's superior performance suggests that it could capture key opcode frequency thresholds more effectively than the other models.

4.2 2-Gram Results

With 2-Gram features (sequential opcode pairs), performance remained similar or slightly lower. The Decision Tree and KNN classifiers both achieved around 40% accuracy, while SVM again lagged at 30%. Despite incorporating sequential information, 2-Gram features did not significantly enhance classifier accuracy due to the small dataset and class imbalance. Poseidon Group continued to be the most consistently identified family across all models.

4.3 Overall Findings

Across both feature sets, the Decision Tree classifier demonstrated the highest and most stable performance, indicating its suitability for small or heterogeneous opcode datasets. SVM performed the weakest, likely due to insufficient data diversity for constructing clear decision boundaries. The results also highlight that opcode-based features alone may not fully capture the complexity of APT malware behavior, and integrating additional static or dynamic analysis features could improve detection accuracy.

5. Evaluation Metrics

To evaluate the classifiers, the following standard performance metrics were used:

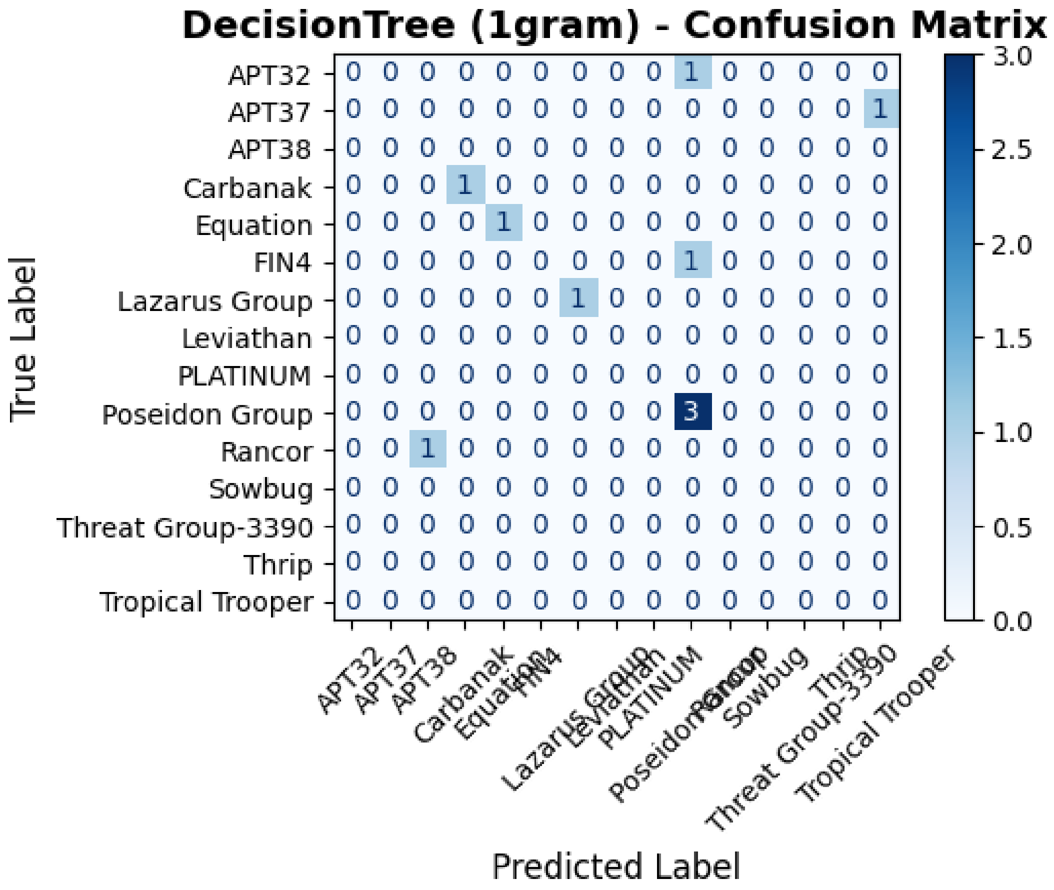
- Accuracy = $(TP + TN) / (TP + TN + FP + FN)$
- Precision = TP / (TP+FP)
- Recall = TP / (TP+FN)
- F1-Score = 2 × (Precision × Recall) / (Precision + Recall)

Where:

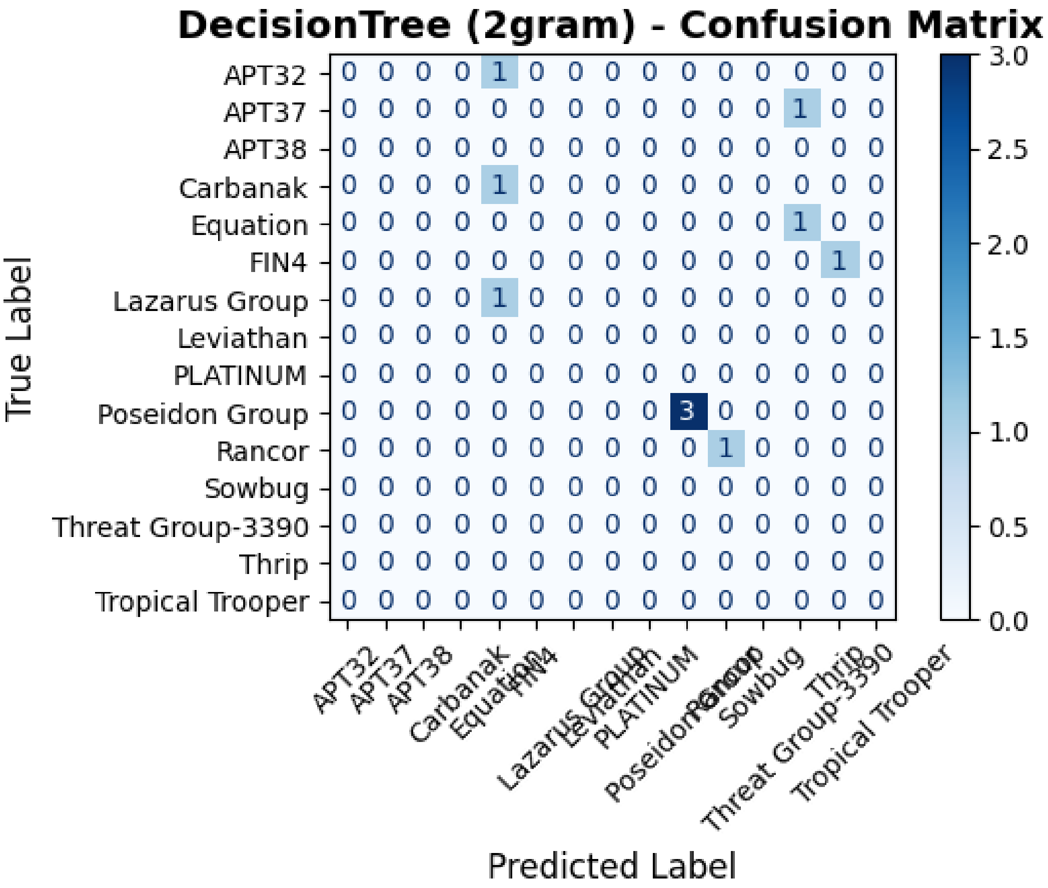
- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

The Confusion Matrix was also used to visualize how many samples were correctly or incorrectly classified by each model.

6. Results

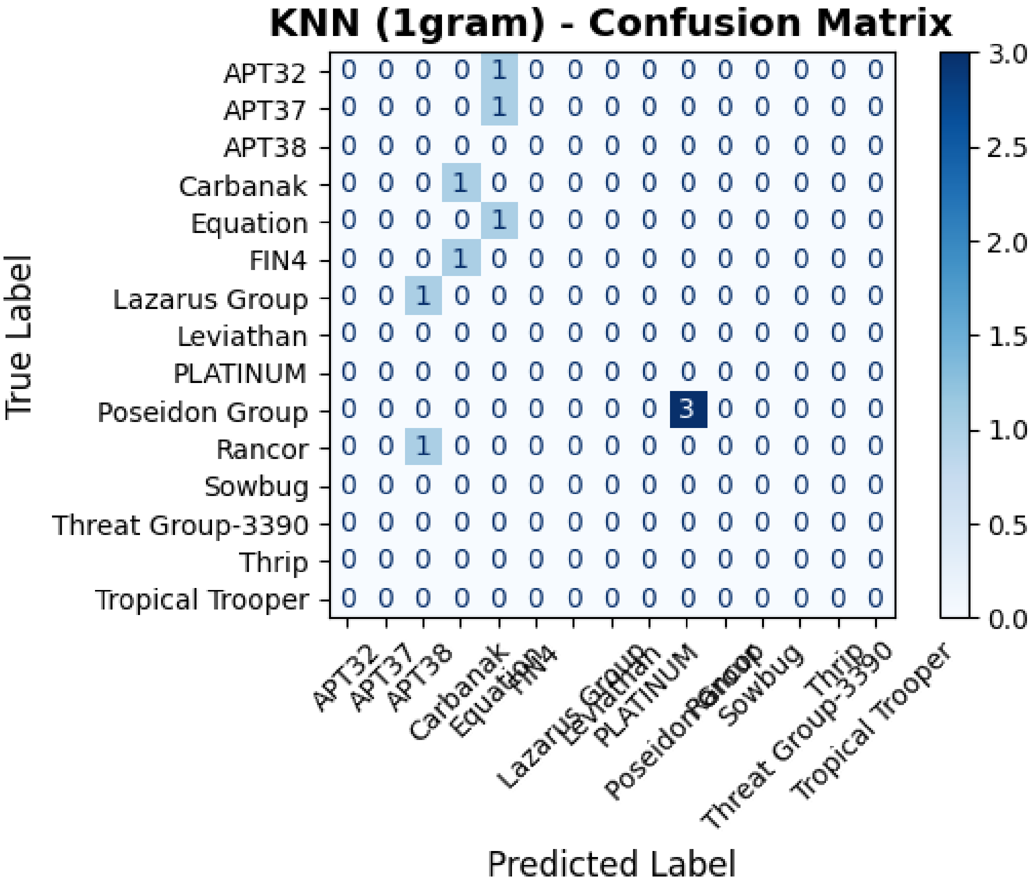


6.1 Decision Tree for 1-gram

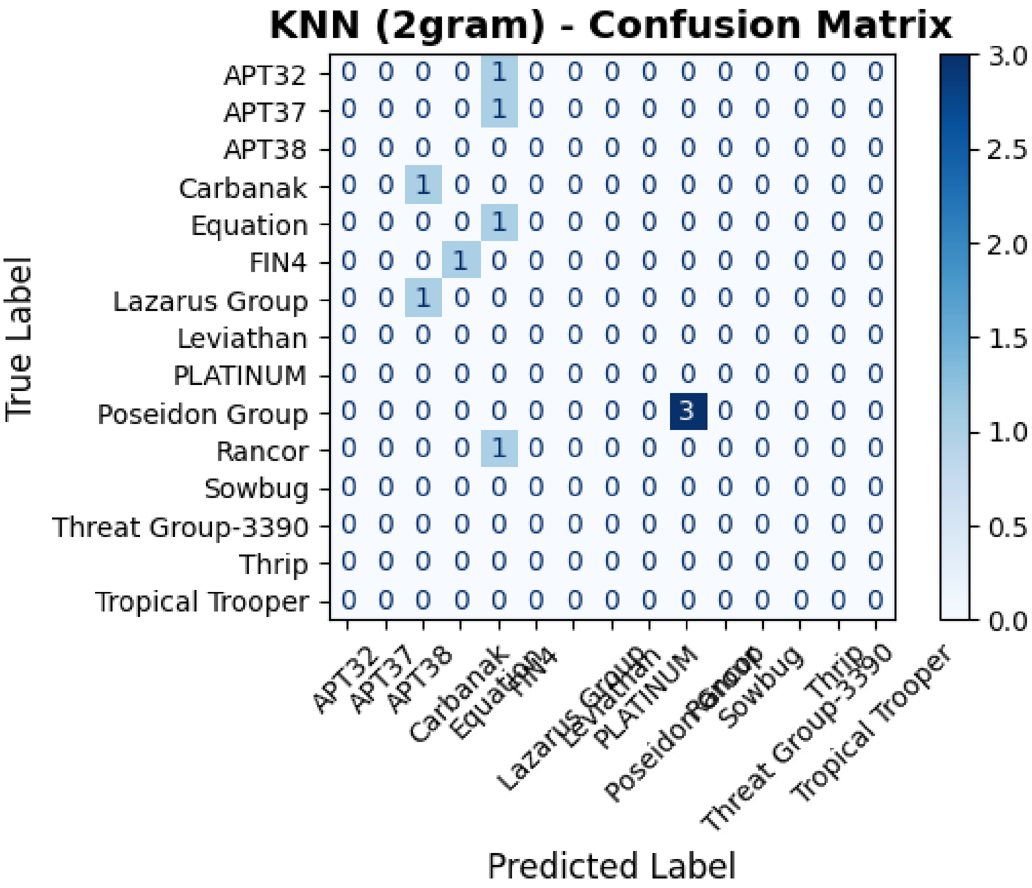


	APT32	APT37	APT38	Carbanak	Equation	FIN4	Lazarus Group	Leviathan	PLATINUM	Poseidon Group	Rancor	Sowbug	Threat Group-3390	Thrip	Tropical Trooper
APT32	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
APT37	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
APT38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Carbanak	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Equation	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
FIN4	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Lazarus Group	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Leviathan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLATINUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Poseidon Group	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
Rancor	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Sowbug	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Threat Group-3390	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Thrip	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tropical Trooper	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

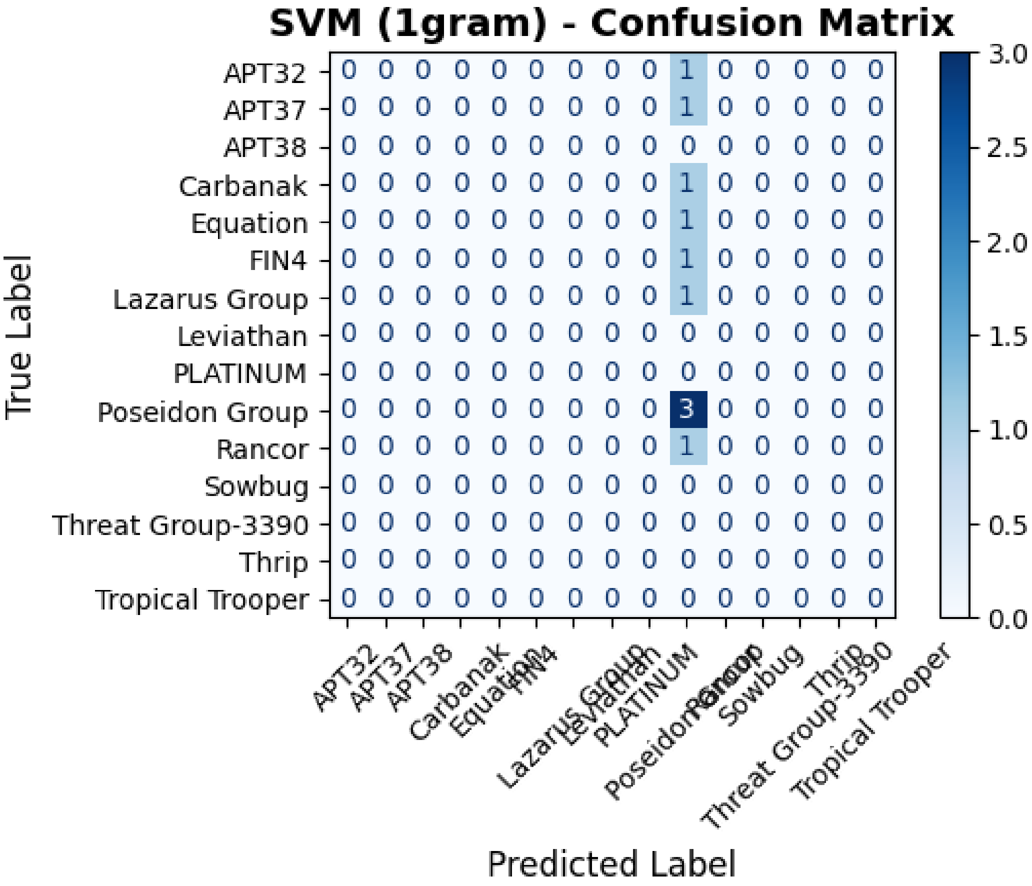
6.2 Decision Tree for 2-gram



6.3 KNN for 1-gram

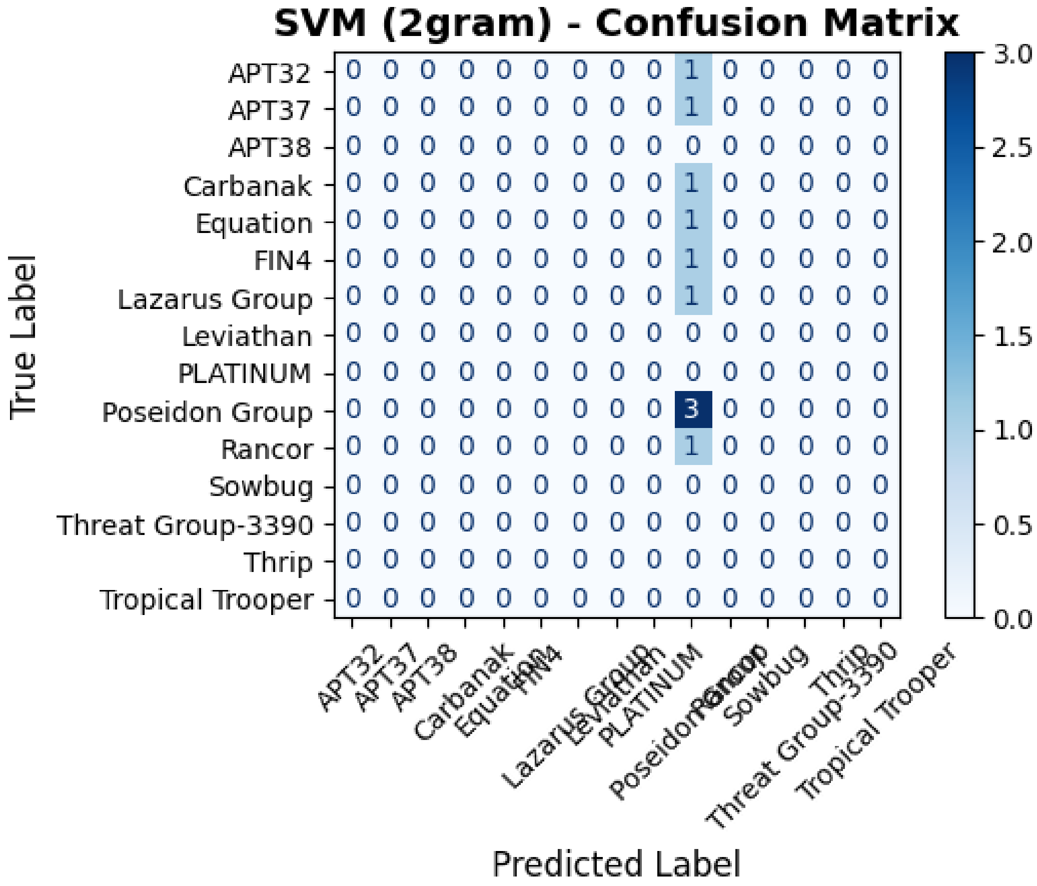


6.4 KNN for 2-gram



	APT32	APT37	APT38	Carbanak	Equation	FIN4	Lazarus Group	Leviathan	PLATINUM	Poseidon Group	Rancor	Sowbug	Threat Group-3390	Thrip	Tropical Trooper
APT32	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
APT37	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
APT38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Carbanak	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Equation	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
FIN4	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Lazarus Group	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Leviathan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLATINUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Poseidon Group	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0
Rancor	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Sowbug	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Threat Group-3390	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Thrip	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tropical Trooper	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

6.5 SVM for 1-gram



6.6 SVM for 2-gram

These are the image-based results

Core findings are as follows:

1. 1-gram data performs slightly better overall than 2-gram data.
2. 2-grams increase feature dimensionality dramatically, but without enough training samples, this leads to sparser data and poorer generalization.
3. This means our dataset is too small for the complexity that 2-grams introduce.
4. Therefore, for small datasets, 1-gram opcode frequencies are more stable.

7. Conclusion

This research applied SVM, KNN (k = 3.5), and Decision Tree classifiers on 1-Gram and 2-Gram opcode features extracted from malware samples of multiple APT groups. Among the tested models, the Decision Tree consistently performed best, achieving up to 60% accuracy with 1-Gram features, while KNN showed moderate results and SVM struggled to generalize.

Although 2-Gram features added opcode sequence context, they did not significantly improve classification performance due to limited and imbalanced data. Poseidon Group was the most consistently identified family, indicating distinctive opcode characteristics compared to other APT samples.

Overall, the results suggest that Decision Trees are more suitable for opcode-based malware detection in small datasets, but broader feature integration and larger sample sizes are needed for more accurate and reliable APT classification.