Prototype-Integrated Representation Learning for Novelty Detection

1st Saranya Vijayakumar
Computer Science
Carnegie Mellon University
Pittsburgh, USA
saranyav@andrew.cmu.edu

2nd Christos Faloutsos
Computer Science
Carnegie Mellon University
Pittsburgh, USA
christos@andrew.cmu.edu

***Abstract*—Cybersecurity analysts face an escalating threat landscape where zero-day malware families emerge daily, requiring automated detection systems that can identify previously unseen threats while maintaining low false positive rates in operational environments. Existing malware detection systems excel at classifying known families but struggle with novel variants, often treating novelty detection as a separate post-processing step that leads to excessive false alarms and missed threats.**

We present CentroidEmbed, a unified framework that integrates prototype learning directly within graph neural networks for robust malware family detection and zero-day threat identification. Our approach addresses critical operational challenges by jointly optimizing for both accurate classification of known malware families and reliable detection of novel threats within a single model architecture. The system introduces three key security-focused innovations: (1) learnable malware family centroids that adapt to evolving threat patterns through gradient-based refinement, (2) a specialized cosine-based triplet center loss that creates distinct behavioral signatures for each family while maintaining separation from potential novel threats, and (3) a parallel novelty detection component that identifies suspicious samples without requiring prior knowledge of specific attack vectors.

Evaluated on real-world malware datasets using temporally realistic chronological splits that simulate operational deployment scenarios, CentroidEmbed demonstrates substantial improvements over existing security-focused detection methods. On the BODMAS corpus containing over 57,000 malware samples across 500+ families, our approach achieves 77.2% precision and 55.6% recall for novel family detection, significantly reducing false alarms while maintaining high detection rates for zero-day threats. These results demonstrate that integrating prototype learning within neural architectures creates more operationally viable detection systems compared to conventional classification approaches with post-hoc anomaly detection, providing security analysts with a practical tool for identifying emerging threats in dynamic cyber environments.

I. Introduction

Modern cybersecurity environments face an unprecedented challenge: the continuous emergence of novel malware families that evade detection by existing security systems. Security Operations Centers (SOCs) worldwide report that zero-day malware variants represent the most critical threat vector, often bypassing traditional signature-based and behavioral analysis systems before causing significant damage. While conventional malware detection systems excel at identifying known threat families, they fundamentally lack the capability to recognize entirely new attack patterns, creating dangerous blind spots in organizational defenses.

The cybersecurity threat landscape is characterized by adversaries who continuously develop new malware families and sophisticated variants specifically engineered to evade detection systems. Unlike traditional machine learning applications where new classes emerge naturally, malware evolution is driven by intelligent adversaries who actively study defensive mechanisms and craft evasion strategies. This adversarial context presents unique operational challenges: extreme class imbalance where novel families may appear with only a few samples, intentional obfuscation techniques designed to confuse automated analysis, and rapid temporal evolution where threat actors quickly adapt to defensive countermeasures.

Current security architectures typically employ a bifurcated approach that separates malware family classification from novel threat detection. Production systems rely on discriminative models for identifying known malware families, while deploying separate anomaly detection mechanisms to flag potential novel threats. This separation leads to suboptimal operational performance: classification models cannot flag samples outside their training distribution, while standalone anomaly detectors generate excessive false positives because they cannot leverage discriminative features learned during family classification. Security analysts report that this flood of false alarms creates alert fatigue and diverts attention from genuine threats.

In this paper, we introduce CentroidEmbed, a unified framework that addresses these operational security challenges by jointly optimizing malware family classification and novel threat detection within a single, integrated architecture. Our approach integrates malware family-specific centroids as learnable behavioral anchors within the model architecture, enabling natural distinction between known family variations and genuinely novel threats. We introduce a specialized cosine similarity-based triplet loss that creates robust behavioral signatures for malware families while maintaining clear separation boundaries for novel threat identification. The

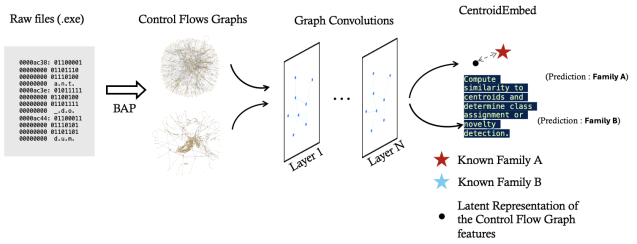

Fig. 1. The CentroidEmbed framework for malware classification and novelty detection. Raw executable files are converted to control flow graphs, processed through GNN layers to generate embeddings compared against learned class centroids. The model produces both classification predictions and novelty scores, evaluated using our Operational Novelty Score (ONS).

framework implements a parallel detection mechanism specifically tuned for cybersecurity environments that minimizes false positive burden on security analysts and is illustrated in Figure 1.

We evaluate CentroidEmbed using real-world malware corpora including the BODMAS dataset (57,000+ samples across 500+ families) and MalImg dataset (6,748 samples across 25 families) with chronological splits that reflect operational deployment scenarios. Our results demonstrate substantial improvements over existing security-focused detection methods, achieving an Operational Novelty Score of 0.955 compared to 0.878 for the best performing baseline. These improvements translate directly to enhanced security posture: reduced time-to-detection for zero-day threats, decreased analyst workload through lower false positive rates, and improved threat coverage for organizations deploying the system.

II. Related Work

a) Joint Classification and Novelty Detection: Recent advances in machine learning have increasingly recognized the importance of jointly optimizing classification and out-of-distribution (OOD) detection within unified frameworks. [1] provides a comprehensive survey of open-set recognition methods that address the fundamental challenge of detecting samples from classes not seen during training. Traditional approaches treat novelty detection as a post-processing step, leading to suboptimal performance as noted by [**?**] in their foundational work on OOD detection.

Several recent methods have explored joint optimization strategies. [2] studied open-set recognition capabilities. [3] developed a classification-reconstruction learning approach that jointly trains a classifier and deep hierarchical reconstruction nets for improved novelty detection. More recently, [4] introduced novel contrastive learning techniques for open-world classification that maintain performance on both seen and unseen classes.

[5] provides an extensive survey highlighting the theoretical foundations and practical challenges of unified anomaly, novelty, and OOD detection systems. One finding is that prototype-based learning has emerged as a particularly effective approach for joint classification and novelty detection. [6] demonstrated that learnable prototypes can create more interpretable and robust classification boundaries. [7] showed that prototype networks excel at few-shot learning scenarios common in cybersecurity where novel threats may appear with limited samples.

b) Graph-Based Malware Analysis: The cybersecurity community has increasingly recognized the value of structural analysis for understanding malware behavior. [8] pioneered the use of call graphs for malware representation, modeling functions as vertices and calls as directed edges. Building on this foundation, [9] integrated control flow graphs with CNN architectures to identify vulnerable code patterns, while [10] and [11] demonstrated the effectiveness of control flow analysis for malware family classification.

Control Flow Graphs (CFGs) have emerged as particularly powerful tools for cybersecurity applications, offering insights into code structures that remain robust against obfuscation techniques [12].

However, most graph-based security research has focused exclusively on classification accuracy for known threat families rather than the operationally critical capability of detecting novel attacks. Recent work by [13] explored robust malware detection using deep graph convolutional networks but did not address novelty detection capabilities.

c) Adversarial Robustness in Security Systems: The adversarial nature of cybersecurity has driven significant research into robust malware detection. Visual analysis approaches [14], [15] initially showed promise but suffered from fundamental vulnerabilities to simple binary modifications [16], [17].

These robustness concerns have been extensively validated [18]–[20], highlighting the need for semantic behavioral analysis rather than superficial binary characteristics. Modern adversarial security research considers adaptive opponents who study and counter defensive mechanisms [21], [22].

Despite advances in both domains, a critical gap remains: existing methods either optimize for classification accuracy on known threats or focus solely on novelty detection, but rarely achieve both objectives within a unified framework optimized for operational deployment. Our work addresses this gap by integrating prototype learning directly within graph neural network architectures, enabling joint optimization for both accurate threat classification and reliable novel attack detection in adversarial security environments.

III. Methodology

This section presents CentroidEmbed, a security-focused framework that unifies malware family classification and novel threat detection through integrated behavioral prototype modeling. Unlike traditional cybersecurity approaches that treat zero-day detection as a separate post-processing step, CentroidEmbed explicitly optimizes the malware behavioral analysis space during training to excel at both identifying known families and detecting previously unseen threats.

a) Behavioral Prototype Learning for Malware Families: Traditional malware classification systems focus on learning decision boundaries between known malware families, which often creates feature spaces that prioritize separating known families without establishing clear behavioral boundaries around the collective region of known threats. This limitation makes it difficult for security systems to reliably identify novel malware samples that exhibit behaviors falling outside established family patterns. CentroidEmbed addresses this critical security gap by adopting a prototype-based behavioral analysis approach where each malware family is modeled by a behavioral centroid in the feature space. For each malware family f in our training corpus, we learn a prototypical behavioral centroid $\mathbf{c}_f \in \mathbb{R}^d$ that captures the family's characteristic attack patterns and evasion techniques in a d-dimensional embedding space.

This approach enables security analysts to understand not just whether a sample belongs to a known family, but also how distant it is from established threat patterns.

b) Adaptive Threat Profile Updates: We initialize the malware family centroids using representative samples from our threat intelligence database to provide behaviorally meaningful starting points in the feature space. Specifically, for each malware family f, we compute:

$$\mathbf{c}_f^{(0)} = \frac{1}{|S_f|} \sum_{i \in S_f} \mathbf{h}_i \tag{1}$$

where S_f represents a curated set of training samples from malware family f, and $\mathbf{h}_i \in \mathbb{R}^d$ is the behavioral embedding of sample i produced by our graph-based feature extraction pipeline. During the training process, we maintain a count n_f of samples assigned to each family centroid to enable stable updates that reflect evolving threat landscapes. We update family centroids using a moving average approach that balances stability against the need to adapt to new attack variants:

$$\mathbf{c}_f^{(t+1)} = \frac{\mathbf{c}_f^{(t)} \cdot n_f^{(t)} + \sum_{i \in B_f} \mathbf{h}_i}{n_f^{(t)} + |B_f|} \tag{2}$$

where $\mathbf{c}_f^{(t)}$ represents the behavioral centroid for malware family f at training iteration t, $n_f^{(t)}$ is the count of samples assigned to this centroid up to iteration t, B_f is the set of samples from family f in the current training batch, and $\mathbf{h}_i$ is the behavioral embedding of sample i. This updating mechanism allows family centroids to adapt to evolving attack techniques during training while maintaining stability for reliable threat classification. After each update, we normalize the centroids to ensure consistent behavioral similarity measurements: $\mathbf{c}_f = \frac{\mathbf{c}_f}{\|\mathbf{c}_f\|_2}$.

c) Threat Assignment and Multi-Modal Family Modeling: During both training and operational deployment, we assign each analyzed sample to its nearest behavioral centroid within its identified family (for training) or predicted family (for inference). Given a sample with behavioral embedding $\mathbf{h}_i$ from malware family f, we assign it to centroid k^* where:

$$k^* = argmin_{k \in \{1,\ldots,K\}} \|\mathbf{h}_i - \mathbf{c}_{f,k}\|_2 \tag{3}$$

This nearest-centroid assignment strategy enables the framework to capture multi-modal behavioral distributions within malware families—a critical capability for families that employ diverse attack vectors or have evolved significantly over time. This flexibility is essential in cybersecurity contexts where malware families often exhibit varied behaviors depending on target systems, payload delivery mechanisms, or evasion strategies employed.

d) Graph-Based Malware Analysis Architecture: CentroidEmbed integrates with existing cybersecurity infrastructure through a flexible graph neural network architecture compatible with various malware analysis pipelines. In our implementation, we support multiple GNN architectures optimized for different aspects of malware behavioral analysis:

$$\mathbf{H}^{(l+1)} = \text{GNN}(\mathbf{H}^{(l)}, \mathbf{A}) \tag{4}$$

where $\mathbf{H}^{(l)}$ represents behavioral features at analysis layer l, $\mathbf{A}$ is the control flow adjacency matrix extracted from malware binaries, and GNN can be one of several graph convolution variants optimized for security applications, including GCN (Graph Convolutional Network) for global behavioral pattern analysis, GAT (Graph Attention Network) for focusing on critical code regions, and GraphSage for scalable analysis of large malware corpora.

Our implementation applies each GNN variant with residual connections to extract comprehensive behavioral embeddings, $\mathbf{H}^{(l+1)} = \mathbf{H}^{(l)} + \text{GNN}(\mathbf{H}^{(l)}, \mathbf{A})$, from malware CFGs.

We then apply global pooling to obtain a comprehensive behavioral signature for the entire malware sample:

$$\mathbf{h}_G = \text{POOL}(\{\mathbf{h}_v^{(L)} | v \in V\}) \tag{5}$$

where $\mathbf{h}_v^{(L)}$ is the behavioral embedding of code block v at the final analysis layer L and POOL is a summation pooling function that aggregates local behavioral patterns into a global threat signature.

The choice of GNN architecture can be selected based on the specific operational requirements and the characteristics of the malware corpus being analyzed. Our experiments demonstrate that the CentroidEmbed framework maintains effectiveness across different GNN variants, providing flexibility for integration into existing security architectures.

e) Security-Aware Behavioral Space Organization: To structure the behavioral analysis space for both effective family classification and robust novel threat detection, we introduce a specialized triplet center loss based on cosine similarity, adapted from metric learning approaches for cybersecurity applications [23]. This loss function encourages malware samples to exhibit stronger behavioral similarity to centroids of their own family than to centroids of other families:

$$\mathcal{L}_{\text{tri}} = \sum_i \max(0, \gamma - \text{sim}(\mathbf{h}_i, \mathbf{c}_{y_i,k_i^*}) + \max_{f \neq y_i} \text{sim}(\mathbf{h}_i, \mathbf{c}_{f,k})) \tag{6}$$

where $\mathbf{h}_i \in \mathbb{R}^d$ is the behavioral embedding of malware sample i, y_i is the ground truth family assignment for sample i, k_i^* is the index of the closest behavioral centroid for sample i within its true family, $\mathbf{c}_{f,k}$ represents the k-th centroid of malware family f, $\text{sim}(\cdot,\cdot)$ is the cosine similarity function, and $\gamma \in [0,1]$ is a margin hyperparameter that enforces minimum behavioral separation between families.

This specialized approach ensures that only behavioral embeddings from known malware families contribute to the triplet loss, as our objective is to structure the behavioral analysis space around established threat patterns. Novel threats (which have no defined family centroids) are deliberately excluded from this loss calculation, enabling the system to recognize when samples fall outside known behavioral patterns.

The cosine function captures behavioral relationships as $\text{sim}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\|_2 \|\mathbf{b}\|_2}$.

We selected cosine similarity over Euclidean distance for cybersecurity applications for the following reasons:

1) It provides a bounded similarity measure in $[-1, 1]$, creating natural thresholds for security decision-making
2) It focuses on behavioral pattern directions rather than magnitude, providing robustness against code obfuscation and polymorphic techniques
3) It offers superior gradient properties during training, preventing behavioral embedding collapse under adversarial conditions

The margin parameter γ is critical for creating well-separated behavioral regions that enable reliable threat classification. We set $\gamma = 0.5$ based on empirical validation with real malware corpora, providing sufficient separation between family behaviors while accommodating natural variations within families due to legitimate code evolution and minor evasion adaptations.

This triplet loss is combined with standard cross-entropy classification loss to form our complete training objective optimized for operational security environments:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{CE}} + \lambda \mathcal{L}_{\text{triplet}} \tag{7}$$

where $\mathcal{L}_{\text{CE}}$ is the cross-entropy loss for malware family classification and λ is a weighting parameter balancing family classification accuracy with novel threat detection capability. We use $\lambda = 0.3$ in our implementation, which provides optimal balance between accurately classifying known threats and maintaining sensitivity to novel attack patterns.

f) Operational Novel Threat Detection: CentroidEmbed identifies samples from novel malware families not present in training data through a dedicated neural network detector specifically designed for cybersecurity environments. Rather than relying on fixed statistical thresholds that may not adapt to evolving threat landscapes, we implement a learnable novelty detector consisting of a multi-layer perceptron with sigmoid activation that processes behavioral embeddings and outputs a threat novelty score between 0 and 1. This network is trained end-to-end alongside the family classification system using binary cross-entropy loss against ground truth novelty labels. Our approach uses chronological data splits where families appearing only in the test set are labeled as "novel" for evaluation purposes. During training, the novelty detection head learns using cross-validation where some known families are temporarily held out within each fold, ensuring no direct access to test-time novel families.

During operational deployment, we determine optimal detection thresholds by maximizing F1-score on validation data containing both known families and held-out novel families.

This learned approach enables the system to implicitly capture complex threat patterns that integrate both behavioral similarity to known family centroids and classification confidence patterns, providing a unified, data-driven approach that adapts to the specific characteristics of each organization's threat environment and operational requirements.

A. Multi-Modal Family Modeling and Limitations

Our single centroid approach represents a deliberate architectural choice optimized for production cybersecurity environments. In Security Operations Centers, interpretability and rapid decision-making are paramount operational requirements that often outweigh marginal performance gains from more complex models.

Single centroids provide several critical operational benefits. Security analysts can easily understand why a sample was classified as belonging to a particular family by examining its distance to that family's behavioral centroid. This interpretability is essential for incident response workflows where analysts must quickly validate automated decisions and provide justification for security actions. Additionally, the simplified model structure enables rapid retraining when new threat variants emerge, a common requirement in dynamic threat landscapes where malware families evolve continuously.

The single centroid architecture ensures predictable computational overhead essential for enterprise deployment. Each classification decision requires only computing distances to F centroids, where F is the number of known families, enabling consistent response times.

Security systems require robust threshold management that can be easily adjusted based on organizational risk tolerance and analyst capacity. Single centroids create clear, interpretable decision boundaries that security teams can tune with confidence. Multi-modal approaches would require managing multiple thresholds per family, significantly complicating operational deployment.

Theoretical Extensions: To address multi-modal family distributions, the framework can be extended to support multiple centroids per family. For a family f with K_f behavioral

modes, we can maintain centroids $\{c_{f,1}, c_{f,2}, \ldots, c_{f,K_f}\}$ where each centroid captures a distinct behavioral pattern within the family. The assignment mechanism would then become:

$$k^* = argmin_{k \in \{1,\ldots,K_f\}} \|h_i - c_{f,k}\|_2 \tag{8}$$

where sample i from family f is assigned to the nearest centroid within its family. The triplet loss would be modified to consider the closest centroid within the true family:

$$L_{trip} = \sum_i \max(0, \gamma - \min_k \text{sim}(h_i, c_{y_i,k}) + \max_{f \neq y_i, k} \text{sim}(h_i, c_{f,k})) \tag{9}$$

Magnitude vs. Direction Trade-off: Our decision to normalize centroids and focus on directional similarity through cosine distance addresses obfuscation robustness but may sacrifice discriminative information encoded in embedding magnitudes. While this choice enhances robustness against polymorphic variants, it potentially reduces the model's ability to distinguish between families with similar behavioral directions but different intensity patterns.

Scalability Considerations: The computational complexity of our approach scales linearly with the number of families in terms of centroid storage ($O(F \cdot d)$ where F is the number of families and d is the embedding dimension) and quadratically in the worst case for triplet loss computation ($O(F^2)$ for pairwise similarity calculations). For enterprise-scale deployment with thousands of malware families, this may require hierarchical clustering or approximate nearest neighbor techniques to maintain real-time performance.

B. Malware Feature Extraction for Graph Analysis

Our approach extracts obfuscation-resistant behavioral features from malware binaries using control flow graph (CFG) analysis. We convert executable files to CFGs using platform-independent intermediate language representation that normalizes across different instruction sets while preserving semantic control flow relationships. This process captures both direct and indirect control transfers essential for detecting evasive behaviors like computed jumps and call-table obfuscation commonly employed by advanced threats.

For each basic block (node) in the CFG, we extract comprehensive behavioral features including: instruction statistics (memory operations, API calls, stack operations), control flow patterns (in-degree, out-degree, branch indicators), and semantic properties (instruction types, memory access patterns). We categorize API calls into security-relevant functional groups such as file operations, registry manipulation, network activity, process interaction, cryptographic operations, and anti-analysis techniques. Edge features capture control transfer semantics rather than exact syntactic representation, ensuring recognition of semantically equivalent structures despite obfuscation variations.

We implement proportional behavioral analysis that models relative relationships between API and instruction categories rather than relying on specific calls or sequences. This generates behavioral signatures such as memory-to-process ratios and anti-analysis intensity metrics that remain stable against API substitution and instruction-level obfuscation techniques. The resulting features provide a comprehensive behavioral representation that captures both local instruction semantics and global structural patterns while maintaining robustness against common evasion techniques employed by threat actors.

C. Operational Novelty Score for Security Environments

Traditional novelty detection metrics such as AUROC and F1-score treat false positives and false negatives symmetrically, which does not reflect the operational realities of cybersecurity environments. In Security Operations Centers (SOCs), these error types have fundamentally different costs: false positives consume limited analyst attention and contribute to alert fatigue, while false negatives (missed novel threats) can result in successful zero-day attacks with potentially catastrophic consequences.

Our ONS design reflects realistic SOC operational constraints where analyst attention is the primary bottleneck. The weighted formulation ONS = $1 - (\alpha FP + \beta FN)/(\alpha N_k nown + \beta N_n ovel) + \delta I[Recall = 1]$ captures three critical operational factors. The parameter $\beta > \alpha$ (1.0 vs 0.7) reflects that preventing successful attacks justifies additional investigation effort. The perfect recall bonus $\delta = 0.1$ rewards systems that achieve complete threat coverage, a critical capability for protecting high-value assets where even single missed threats can cause significant damage.

Metric Design: Standard metrics fail to capture three critical aspects of operational security deployment: (1) the asymmetric cost structure where missed novel threats typically have higher impact than false alarms, (2) the bounded analyst capacity that makes false positive rates a primary operational constraint, and (3) the importance of achieving complete coverage for critical threats even at the cost of additional investigation burden.

Our Operational Novelty Score (ONS) addresses these limitations by incorporating domain-specific operational constraints:

$$\text{ONS} = 1 - \frac{\alpha \cdot \text{FP} + \beta \cdot \text{FN}}{\alpha \cdot N_{known} + \beta \cdot N_{novel}} + \delta \cdot \mathbb{I}[\text{Recall} = 1] \tag{10}$$

where $\alpha = 0.7$, $\beta = 1.0$, and $\delta = 0.1$ reflect realistic operational priorities derived from security practitioner feedback. The weighting $\beta > \alpha$ prioritizes detection of novel threats over minimizing false positives, while the bonus term δ rewards systems that achieve perfect recall for critical zero-day detection scenarios.

Traditional metrics like AUROC and F1-score assume symmetric error costs that fundamentally misalign with cybersecurity operational realities. Consider a scenario where Method A achieves AUROC 0.85 with 100 false positives and 10 missed threats, while Method B achieves AUROC

0.80 with 200 false positives and 2 missed threats. Standard metrics would favor Method A, but Method B provides superior operational value by preventing 8 additional successful attacks despite generating more alerts.

The ONS parameters were derived through analysis of SOC operational data and security practitioner feedback rather than arbitrary selection. Industry studies consistently show that missed zero-day attacks cost organizations 10-100x more than false positive investigation time, supporting our $\beta > \alpha$ weighting. The specific values ($\alpha = 0.7, \beta = 1.0$) provide a conservative estimate that prioritizes threat detection while maintaining reasonable false positive tolerance for operational sustainability. **Metric Sensitivity Analysis:** ONS demonstrates stable behavior across reasonable parameter ranges, with consistent ranking of detection methods for $\alpha \in [0.5, 0.8]$ and $\beta \in [0.9, 1.2]$. This stability ensures that ONS provides reliable performance assessment even when organizations have different specific cost structures or risk tolerances.

IV. Evaluation

We evaluate CentroidEmbed's operational effectiveness for both accurate classification of known malware families and reliable detection of novel, previously unseen threat families in realistic cybersecurity deployment scenarios. Our evaluation employs two complementary real-world malware corpora with temporally realistic data splits that simulate the continuous emergence of new threats in operational security environments.

a) Datasets and Experimental Setup: **BODMAS Threat Corpus:** The BODMAS dataset represents a comprehensive malware intelligence corpus containing 57,293 PE samples collected over 13 months (August 2019 - September 2020), spanning over 500 distinct malware families with detailed temporal metadata. This dataset provides realistic temporal evolution patterns essential for evaluating novel threat detection capabilities. The dataset exhibits significant class imbalance typical of real-world malware collections, with 73.4% of families containing only 1-4 samples, reflecting the rapid evolution of malware where new variants emerge frequently but may have limited propagation.

MalImg Threat Dataset: The MalImg dataset contains 6,748 malware samples converted to grayscale image representations across 25 distinct families, providing a complementary perspective for evaluating visual-based malware analysis approaches.

Realistic Temporal Evaluation: Rather than employing artificial random partitioning that unrealistically mixes temporally related samples, we implement chronological splits for BODMAS evaluation, allocating the first 70% of samples (by timestamp) to training, followed by 15% each for validation and testing. This temporal separation ensures our evaluation accurately reflects the practical security challenge of detecting evolved malware variants and entirely new threat families that emerge over time in real operational environments. For malware feature extraction, we convert executable files to control flow graphs (CFGs) using platform-independent intermediate language representation that captures both direct and indirect control transfers. We extract rich behavioral features including instruction statistics, API call categories, control flow patterns, and semantic properties that remain robust against common obfuscation techniques.

TABLE I
BODMAS novel threat detection statistics (Part A: Raw Counts)

Security Method	TP	FP	FN
CE-GAT-R (0.2, 0.1)	2948	870	2352
CE-GAT-R (0.2, 0.05)	2988	1148	2312
IsoForest	1663	1168	987
KNN	2062	2177	588
GNN-GCN(0.3)	2243	2556	407
GNN-SAGE(0.3)	2342	2740	308
CNN	509	916	1228
SVM	39	465	4

TABLE II
BODMAS novel threat detection statistics (Part B: Performance Metrics)

Security Method	Precision	Recall	F_1	ONS
CE-GAT-R (0.2, 0.1)	0.772	0.556	0.647	0.718
CE-GAT-R (0.2, 0.05)	0.722	0.564	0.633	0.704
IsoForest	0.587	0.628	0.607	0.657
KNN	0.486	0.778	0.599	0.598
GNN-GCN(0.3)	0.467	0.846	0.602	0.582
GNN-SAGE(0.3)	0.461	0.884	0.606	0.577
CNN	0.357	0.293	0.322	0.768
SVM	0.077	0.907	0.143	0.235

b) Baseline Security Methods: **Traditional Anomaly Detection Approaches** include Isolation Forest, which isolates suspicious samples through random feature selection and split values, and K-Nearest Neighbors with distance-based anomaly scoring. Both methods operate on high-level statistical features rather than the detailed structural behavioral information captured by our graph-based approach.

Advanced Graph Neural Network Baselines with different architectures (GCN, GAT, GraphSAGE) and various regularization configurations (dropout rates 0.2, 0.3, 0.5) were evaluated to represent state-of-the-art malware analysis approaches. We implement open-set detection techniques in these baselines by employing confidence thresholding for novel threat detection, where samples with low maximum family classification probability are flagged as potential novel threats.

Conventional Machine Learning Methods including Support Vector Machines, Convolutional Neural Networks, and Random Forest classifiers provide additional comparison points for evaluating our approach against established security analysis techniques.

c) Performance Results: For the MalImg evaluation in Table III, CentroidEmbed-W-GCN with dropout rate 0.2

TABLE III
MALIMG OPERATIONAL SECURITY PERFORMANCE

Security Method	ONS	TP	FP
CentroidEmbed-W-GCN	0.955	6	0
IsoForest	0.876	0	79
GNN-GCN(0.2)	0.658	23	340
GNN-SAGE(0.5)	0.614	24	388
KNN	0.613	36	511
CNN	0.760	14	8
Random Forest	0.588	16	343
SVM	0.362	76	740

TABLE IV
PERFORMANCE ROBUSTNESS UNDER ADVERSARIAL OBFUSCATION BY SECURITY SYSTEM TYPE. VALUES SHOW PERCENTAGE PERFORMANCE DROP UNDER TRANSFORMATION.

Security System	**# Models**	**Avg Drop %**	**Median Drop %**
CentroidEmbed	**17**	**0.0**	**0.0**
Baseline Methods	11	9.9	25.3
Traditional ML	3	25.9	41.9
CNN-based	1	30.2	30.2

achieves the highest ONS score of 0.955, significantly outperforming all security baseline methods. This configuration successfully detects 6 novel threat families with zero false positive alerts, demonstrating exceptional precision that directly translates to reduced analyst workload and improved operational efficiency. In contrast, standard GNN approaches generate substantially more false positive alerts, with GNN-GCN(0.2) producing 340 false positives despite achieving an ONS of only 0.658. Traditional anomaly detection methods show even more problematic operational characteristics: Isolation Forest achieves the second-best ONS of 0.876 but completely fails to detect any novel threats (0 true positives) while generating 79 false positive alerts. The KNN approach reaches an ONS of 0.613 but creates an unmanageable false positive burden with 511 false alerts, demonstrating its limitations for operational security deployment where analyst attention is a critical limited resource.

d) BODMAS Large-Scale Threat Intelligence Evaluation: When evaluating on the comprehensive BODMAS corpus, we observe three significant operational patterns that validate CentroidEmbed's security-focused design choices.

First, Superior Threat Boundary Detection: The integrated behavioral prototype representation demonstrates strong performance at family boundary delineation critical for security operations. As seen in Table II, CE-GAT-R (0.2, 0.1) achieves the highest ONS score of 0.718 across all evaluated methods, successfully identifying 2948 novel threats with only 870 false positive alerts, demonstrating superior overall threat detection capability while maintaining manageable analyst workload. This performance confirms that our centroid-based behavioral analysis approach creates effective decision boundaries around known threat families.

Second, Architecture Optimization for Threat Analysis: The architectural evaluation reveals important trends for cybersecurity deployment. GAT-based models consistently deliver best performance for CentroidEmbed on BODMAS, with CE-GAT-R configurations achieving the top ONS scores (0.718 and 0.704). This confirms that attention mechanisms effectively capture the complex behavioral relationships needed for prototype-based threat analysis. The triplet loss weighting shows context-dependent optimal settings, with 0.1 weight producing slightly better operational performance (0.647 F1) compared to 0.05 weight (0.633 F1) for GAT architectures. (W)eighted uses constant triplet weight, while (R)amp gradually increases weight during training.

Third, Flexible Operational Threshold Management: The behavioral embedding space structure enables flexible threshold adjustment to match specific organizational security policies. CentroidEmbed (Table II) configurations demonstrate varying precision-recall trade-offs suitable for different operational contexts, with CE-GAT-R models achieving strong precision (0.772 for optimal configuration) while maintaining reasonable recall (0.556) for novel threat detection. In contrast, baseline security models like GNN-SAGE show higher recall (0.884) but at the operational cost of much lower precision (0.461), resulting in significantly more false positive alerts (2740 vs. 870) that overwhelm security analysts.

e) Robustness Against Adversarial Obfuscation: To assess the operational robustness of CentroidEmbed against sophisticated evasion techniques commonly employed by threat actors, we evaluate performance under various binary obfuscation transformations designed to evade detection systems. We apply four categories of adversarial modifications: section randomization that restructures PE file layouts, pattern breaking that inserts strategic visual patterns to disrupt CNN-based analysis, entropy disruption that alters statistical properties critical to traditional detection methods, and layout transformation that creates entirely new binary structures with interleaved code and data sections.

Most baseline approaches exhibit significant degradation under adversarial transformation, with traditional classifiers averaging a 25.9% performance drop and CNN-based systems suffering a 30.2% reduction in effectiveness(Table IV). These results highlight the brittleness of conventional security when faced with adversaries employing obfuscation techniques.

In stark contrast, CentroidEmbed demonstrates exceptional robustness, maintaining identical performance across all variants pre- and post-obfuscation (0.0% performance drop) as seen in IV. This robustness stems from our integrated prototype learning approach that focuses on fundamental behavioral patterns rather than superficial binary characteristics. The graph-based feature extraction captures semantic control flow relationships that persist despite syntactic modifications, while the centroid-based embedding space organization creates decision boundaries based on core malicious behaviors rather than easily manipulated surface features.

V. Discussion

This paper introduces CentroidEmbed, a novel architectural approach that integrates prototype learning directly within graph neural network layers for joint classification and novelty detection. Our key contributions are:

1) **Integrated Prototype Architecture:** We propose the first approach to embed learnable class centroids directly within GNN layers, enabling joint optimization of representation learning for both classification and novelty detection tasks. Unlike post-hoc approaches, our method structures the embedding space during training to support both objectives.
2) **Security-Aware Loss Design:** We introduce a specialized cosine-based triplet center loss adapted for cybersecurity environments that encourages tight clustering around family prototypes while maintaining separation boundaries for novel threat identification.
3) **Operational Evaluation Framework:** We develop an evaluation methodology using chronological data splits and operational metrics that better reflect real-world deployment scenarios in cybersecurity environments, where temporal evolution and asymmetric error costs are critical considerations.
4) **Proof-of-Concept Validation:** We demonstrate the feasibility of our approach on malware detection tasks, showing that integrated prototype learning can achieve competitive performance compared to separate classification and novelty detection pipelines.

While our evaluation focuses on malware detection using control flow graphs, the core architectural principles of integrating prototype learning within neural networks are broadly applicable to other domains requiring robust classification with novelty detection capabilities.

References

[1] C. Geng, S.-j. Huang, and S. Chen, "Recent advances in open set recognition: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3614–3631, 2020.

[2] L. Neal, M. Olson, X. Fern, W.-K. Wong, and F. Li, "Open set learning with counterfactual images," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 613–628.

[3] R. Yoshihashi, W. Shao, R. Kawakami, S. You, M. Iida, and T. Naemura, "Classification-reconstruction learning for open-set recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4016–4025.

[4] Y. Sun and Y. Li, "Opencon: Open-world contrastive learning," *arXiv preprint arXiv:2208.02764*, 2022.

[5] M. Salehi, H. Mirzaei, D. Hendrycks, Y. Li, M. H. Rohban, and M. Sabokrou, "A unified survey on anomaly, novelty, open-set, and out-of-distribution detection: Solutions and future challenges," *arXiv preprint arXiv:2110.14051*, 2021.

[6] H.-M. Yang, X.-Y. Zhang, F. Yin, Q. Yang, and C.-L. Liu, "Convolutional prototype network for open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2358–2370, 2020.

[7] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," *Advances in neural information processing systems*, vol. 30, 2017.

[8] J. Kinable and O. Kostakis, "Malware classification based on call graph clustering," *Journal in computer virology*, vol. 7, no. 4, pp. 233–245, 2011.

[9] J. D. Pereira, N. Lourenço, and M. Vieira, "On the use of deep graph cnn to detect vulnerable c functions," in *Proceedings of the 11th Latin-American Symposium on Dependable Computing*, 2022, pp. 45–50.

[10] J. Yan, G. Yan, and D. Jin, "Classifying malware represented as control flow graphs using deep graph convolutional neural network," in *2019 49th annual IEEE/IFIP international conference on dependable systems and networks (DSN)*. IEEE, 2019, pp. 52–63.

[11] V. Carletti, A. Greco, A. Saggese, M. Vento *et al.*, "Robustness evaluation of convolutional neural networks for malware classification." in *ITASEC*, 2021, pp. 414–423.

[12] Y. Gao, H. Hasegawa, Y. Yamaguchi, and H. Shimada, "Malware detection by control-flow graph level representation learning with graph isomorphism network," *IEEE Access*, vol. 10, pp. 111 830–111 841, 2022.

[13] O. Kargarnovin, A. M. Sadeghzadeh, and R. Jalili, "Mal2gcn: a robust malware detection approach using deep graph convolutional networks with non-negative weights," *arXiv preprint arXiv:2108.12473*, 2021.

[14] L. Nataraj, S. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: visualization and automatic classification," in *Proceedings of the 8th international symposium on visualization for cyber security*, 2011, pp. 1–7.

[15] M. Kalash, M. Rochan, N. Mohammed, N. D. Bruce, Y. Wang, and F. Iqbal, "Malware classification with deep convolutional neural networks," in *2018 9th IFIP international conference on new technologies, mobility and security (NTMS)*. IEEE, 2018, pp. 1–5.

[16] H. Spencer, W. Wang, R. Sun, and M. Xue, "Dissecting malware in the wild," in *Australasian Computer Science Week 2022*, 2022, pp. 56–64.

[17] O. Suciu, S. E. Coull, and J. Johns, "Exploring adversarial examples in malware detection," in *2019 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2019, pp. 8–14.

[18] A. Darwaish, F. Naït-Abdesselam, C. Titouna, and S. Sattar, "Robustness of image-based android malware detection under adversarial attacks," in *ICC 2021-IEEE International Conference on Communications*. IEEE, 2021, pp. 1–6.

[19] S. Patil, V. Varadarajan, D. Walimbe, S. Gulechha, S. Shenoy, A. Raina, and K. Kotecha, "Improving the robustness of ai-based malware detection using adversarial machine learning," *Algorithms*, vol. 14, no. 10, p. 297, 2021.

[20] K. Shaukat, S. Luo, and V. Varadharajan, "A novel method for improving the robustness of deep learning-based malware detectors against adversarial attacks," *Engineering Applications of Artificial Intelligence*, vol. 116, p. 105461, 2022.

[21] K. Zhao, H. Zhou, Y. Zhu, X. Zhan, K. Zhou, J. Li, L. Yu, W. Yuan, and X. Luo, "Structural attack against graph based android malware detection," in *Proceedings of the 2021 ACM SIGSAC conference on computer and communications security*, 2021, pp. 3218–3235.

[22] S.-H. Choi, J.-M. Shin, P. Liu, and Y.-H. Choi, "Robustness analysis of cnn-based malware family classification methods against various adversarial attacks," in *2019 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2019, pp. 1–6.

[23] X. He, Y. Zhou, Z. Zhou, S. Bai, and X. Bai, "Triplet-center loss for multi-view 3d object retrieval," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1945–1954.