Androidマルウェア検知器への回避攻撃に対するSHAPとマルチタスク学習を用いた防御手法

松元 優斗[1,a)]　畑田 充弘[2]　吉浦 裕[1]　市野 将嗣[1]

概要：Android マルウェア検知システムは，Adversarial Example(AE) と呼ばれる巧妙な回避攻撃に対して脆弱である．既存の防御手法である Adversarial Training(AT) は，通常検体への精度低下や未知の攻撃への汎化性能の限界，さらに検知後の解析支援機能の欠如という課題を抱えている．本稿では，これらの課題を解決するため，SHAP とマルチタスク学習を統合した包括的な防御フレームワークを提案する．16 種類の攻撃手法を用いた評価実験において，提案手法は通常検体に対して F1 スコア 0.9986 を維持しつつ，AE に対して平均 0.9901 の Robust Accuracy を達成した．さらに，AE 検知時にマルウェアカテゴリを平均 Recall 0.8129 で分類可能であり，SHAP 値の分析により回避攻撃手法の解析支援の可能性を示した．本研究は，堅牢な検知と実用的な解析支援を単一フレームワークで実現した初の包括的防御手法である．

キーワード：回避攻撃，Adversarial Example，SHAP，マルチタスク学習

Defense Method Using SHAP and Multi-Task Learning Against Evasion Attacks on Android Malware Detectors

Yuto Matsumoto[1,a)]　Mitsuhiro Hatada[2]　Hiroshi Yoshiura[1]　Masatsugu Ichino[1]

Abstract: Android malware detection systems are vulnerable to sophisticated evasion attacks known as Adversarial Examples (AE). Existing defense methods based on Adversarial Training (AT) suffer from accuracy degradation on natural samples, limited generalization to unknown attacks, and lack of post-detection analysis support. In this paper, we propose a comprehensive defense framework that integrates SHAP and multi-task learning to address these challenges. In evaluation experiments using 16 attack methods, our approach maintains an F1 score of 0.9986 on natural samples while achieving an average Robust Accuracy of 0.9901 against AEs. Furthermore, the system can classify malware categories with an average Recall of 0.8129 during AE detection, and SHAP value analysis demonstrates the potential for evasion attack analysis support. This work presents the first comprehensive defense method that achieves both robust detection and practical analysis support in a unified framework.

Keywords: Evasion Attack, Adversarial Example, SHAP, Multi-task Learning

1. はじめに

Android 端末の普及に伴い，それらを標的とするマルウェアの脅威は深刻化している．AV-TEST によれば，2021 年から 2024 年の 4 年間で新たに発見された Android マルウェアは約 690 万に上り [1]，個人のプライバシーから企業活動まで広範な領域に影響を及ぼしている．この脅威の増大は，マルウェア検体を迅速かつ正確に分析するセキュリティアナリストの負担を高めている．従来のシグネチャベースの検知手法は日々巧妙化する未知のマルウェアに対応できないため [2]，機械学習を応用した検知技術の研究が盛んに行われている．しかし，機械学習モデルは

[1] 電気通信大学
The University of Electro-Communications
[2] NTT ドコモビジネス株式会社
NTT DOCOMO BUSINESS, Inc.
[a)] matsumoto@uec.ac.jp

Adversarial Example（AE）と呼ばれる新たな脆弱性を抱えている [3]．AE とは，人間には知覚できない微小な摂動をデータに加えることで，モデルに誤った判断をさせる入力である．攻撃者はマルウェア検体を意図的に改変し，本来悪性と判定されるべき検体を良性と誤認識させることが可能となる [4]．こうした回避攻撃への防御策として，AE を学習データに加える Adversarial Training が広く研究されている．しかし，この手法は未知の攻撃への汎化性能が限定的である [5] ほか，実運用で必要な解析支援機能を提供しない．単に AE を悪性と分類できても，属するマルウェアファミリや使用された攻撃手法が不明では，迅速な対応や将来の対策立案が困難となる．

本稿は，AE の効率的な検知と解析支援の両立を目指し，マルチタスク学習と SHAP に基づく防御手法を提案する．提案モデルは，AE 検知，マルウェア検知，カテゴリ分類の 3 タスクを同時学習することで汎化性能を向上させ，AE 検知時にマルウェアの属性情報も提供する．さらに，XAI 技術である SHAP を適用することで，攻撃の痕跡となった特徴量を特定可能とし，アナリストへ実用的な解析支援を提供する．

本研究の主な貢献は以下の通りである．

- AE の検知と解析支援を両立させる，マルチタスク学習に基づいた包括的な防御フレームワークの提案
- マルウェア検知，カテゴリ分類，AE 検知の 3 タスクを統合学習することによる検知精度と解析効率の向上
- SHAP の適用による攻撃の解釈性向上
- 16 種類の攻撃手法を用いた評価実験による，提案手法の堅牢性と有効性の実証

2. 関連研究

2.1 回避攻撃に対する防御手法

回避攻撃に対する防御手法として，Adversarial Training(AT) が最も代表的かつ効果的な手法として広く研究されている [6]．AT は，攻撃手法を用いて生成した AE をモデルの学習に組み込むことで，堅牢性の獲得を目指す．この手法は，モデルパラメータに関する損失の最小化と，入力データに関する損失の最大化を同時に行う min-max 最適化問題として定式化され，これによりモデルは AE に対して堅牢な決定境界を学習することが期待される．

マルウェア検知分野においても，AT の有効性は複数の研究で実証されている．Grosse ら [4] は，Android マルウェア分類器に対する効果的な AE 生成手法（Grosse Attack）を提案し，これを組み込んだ AT の有効性と限界を検証した．Al-Dujaili ら [7] は，悪意のある機能が維持された Windows PE の AE を生成する攻撃手法を提案し，rFGSM^k を用いた AT モデルが最も堅牢性が高く，他の回避攻撃に対しても一定の効果を示すことを実証した．

より高度な AT として，Li ら [8] は複数の AE 生成手法と操作セットを組み合わせた Mixture of Attacks を提案し，これを組み込んだ Adversarial Deep Ensemble (ADE) による堅牢な DNN アンサンブル防御を構築した．実験において計 26 種類の回避攻撃に対し，ADE は顕著な堅牢性を示した．Wang ら [9] は，マルウェア検知とファミリ分類の性能向上，および回避攻撃への堅牢性向上を目指し，AT に基づく AdvAndMal を提案した．この手法は，マルウェア検出とファミリ分類の総合的な精度向上と，AE への堅牢性向上を同時に達成している．最近の研究では，Lucas ら [10] が Windows PE マルウェアの検知領域において，AE の生成効率と規模を大幅に向上させる手法を検討し，IPR 攻撃，Disp 攻撃，Kreuk 攻撃に対し高い堅牢性を示した．さらに，Li ら [11] は Provable Defense の観点から，堅牢性を数学的に保証する PAD という AT 手法を提案し，27 種類の回避攻撃に対して高い検知精度を達成した．

また，AT 以外のアプローチも検討されている．Adversarial Purification は，AE に付加された摂動を除去して無害化を図る手法である．Zhou ら [12] の MalPurifier は，正常な入力分布への復元により高い防御性能を示している．Adversarial Detection は，入力の統計的異常性やモデルの活性化状態を利用して AE 自体を検知する．Zhang ら [13] の HagDe や Li ら [14] の RAMDA は，AE の知識を必要とせず高い検知性能を達成している．これらの手法は既存分類器への追加モジュールとして機能し，AT とは異なる設計思想を持つ．Melis ら [15] は勾配ベースの解釈手法を用いて，解釈の均一性と堅牢性の相関を分析し，XAI 技術の防御への応用可能性を示唆している．

2.2 包括的防御フレームワークに求められる要件

既存研究の多くは特定の技術的指標の最適化に焦点を当ててきたが，実運用では，より包括的な機能が求められる．本節では，これらの課題を踏まえ，実用的な防御フレームワークが満たすべき 4 つの要件を定義する．

要件 (i) 通常検体の検知精度の維持: 実運用において誤検知の増加は業務効率を著しく低下させる．しかし，AT では堅牢性獲得の代償として通常検体への精度が低下するトレードオフが頻繁に観察される [5], [16]．一方，防御機構の導入により，通常検体の検知精度が低下することは許容されない．ベースラインモデルと同等以上の検知精度を維持する必要がある．

要件 (ii) 未知の攻撃への汎化性能: 攻撃者は常に新たな回避手法を開発し続けている．しかし，多くの AT 手法は学習時に使用した攻撃手法に過剰適合し，汎化性能が低い点が指摘されている [5]．学習に用いていない攻撃手法で生成された AE に対しても，安定した検知性能を発揮する必要がある．特定の攻撃手法の過適合を避け，攻撃の本質的な特性を捉える汎用的な検知メカニズムが求められる．

要件 (iii) 迅速なトリアージ支援: アナリストは日々大

量のアラートに対応している．しかし，既存の AT 手法の多くは，AE を単に”悪性”と分類することに焦点を当てており，検知後の解析を考慮していない．実運用では，具体的な脅威カテゴリを特定し，リスク評価と対応優先度の決定を迅速化する必要がある．これは限られたリソースで効率的なインシデント対応を実現するために不可欠である．

要件 (iv) 回避攻撃手法の分析支援: 効果的な再発防止策の立案には，使用された攻撃手法の理解が不可欠である．しかし，AT は AE を正しく分類する能力の獲得に終始し，判断根拠や攻撃に利用された特徴の情報を提供しない．実運用では，操作された特徴量を明らかにすることで，攻撃シグネチャの作成や，脆弱性を持つモデルの改善につながる実用的な知見を提供する必要がある．

2.3 関連研究の要件対応状況と本研究の位置づけ

前節で定義した 4 要件に対する既存手法の充足度を表 1 に示す．

既存の研究は要件 (i) から (iv) の一部を満たすに留まり，4 つの要件を同時に満たす包括的な解決策は提示されていない．AT を基盤とする手法群 [4], [7], [8], [9], [10], [11] は要件 (i), (ii) の両立を目指しているが，実運用における要件 (iii), (iv) への対応は AdvAndMal [9] の要件 (iii) のみである．Adversarial Purification や Adversarial Detection を目的とする手法郡 [12], [13], [14] は要件 (ii) において優れた性能を示すが，既存分類器の追加モジュールという位置づけのため要件 (iii), (iv) に対応しない．Meils ら [15] の研究は，要件 (ii) の原理的な評価を可能にし，要件 (iv) に寄与する可能性を示唆するが，実装には至っていない．

この状況は，既存研究が堅牢なモデルの構築 (要件 (i), (ii)) と，実運用における効率的な脅威対応 (要件 (iii), (iv)) を分離して扱ってきたことを示している．本研究は，この課題解決に貢献するものであり，4 要件を満たす包括的な防御能力の実現を目指す．

3. 提案手法

本稿では，2.2 項で定義した 4 要件を包括的に満たすため，マルチタスク学習と SHAP を組み合わせた防御手法を提案する．本手法は単一モデルでマルウェアの検知と分類から回避攻撃の検知と分析支援までを行うことで，実運用における堅牢性と効率性の両立を目指す．

本提案手法は，以下のアプローチにより各要件を満たす．それぞれの詳細なメカニズムは，後続する節で詳述する．

要件 (i) 通常検体の検知精度の維持: マルチタスク学習により，AE 検知専用タスクを設けることで，マルウェア検知への悪影響部分のみを取り除くことができる．

要件 (ii) 未知の攻撃への汎化性能: モデルの判断根拠に着目する SHAP を用いた AE 検知により実現する．

要件 (iii) 迅速な脅威トリアージ支援: マルチタスク学習にマルウェアカテゴリ分類を組み込むことで実現する．

要件 (iv) 回避攻撃手法の分析支援: SHAP が持つ高い解釈性を応用することで実現する．

表 1 関連研究における防御手法の要件対応表

著者	要件			
	(i)	(ii)	(iii)	(iv)
Grosse ら (2017) [4]				◑
Al-Dujaili ら (2018) [7]	✓	✓		
Li ら (2020) [8]	◑	◑		◑
Wang ら (2021) [9]	✓		✓	
Li ら (2021) [14]	✓	✓		
Lucas ら (2023) [10]	◑	✓		◑
Li ら (2024) [11]	✓	✓		
Zhou ら (2025) [12]	—	✓		
Zhang ら (2025) [13]	—	✓		
Melis ら (2021) [15]	—	✓	—	◑
提案手法	✓	✓	✓	✓

✓: 要件を満たす, ◑: 要件を部分的に満たす, —: 要件の目的範囲外

3.1 モデルアーキテクチャ

本提案手法は，マルチタスク学習と SHAP を組み合わせた防御手法である．マルチタスク学習は，複数の関連するタスクを単一のモデルで同時に学習する機械学習のアプローチである [17].

本稿で提案するマルチタスク学習モデルのアーキテクチャを図 1 に示す．

バックボーンネットワーク: 入力されたアプリケーションのバイナリ特徴ベクトルから，特徴表現を抽出する全結合ネットワークである．このバックボーンは 3 つのタスクで共有されており，各タスクに共通した有用な特徴表現の学習を期待する．

マルウェア検知ヘッド: バックボーンから抽出された特徴表現を入力とし，入力されたアプリケーションがマルウェアかクリーンウェアかを二値分類するヘッドである．これにより AE ではない通常のマルウェア検知に対応する．

マルウェアカテゴリ分類ヘッド: バックボーンからの特徴表現を入力とし，マルウェアカテゴリを特定するマルチラベル分類を行うヘッドである．どのマルウェアカテゴリを採用するかは必要に応じて決定する．本実験で用いたマルウェアカテゴリについては 4.1 にて後述する．

AE 検知ヘッド: マルウェア検知ヘッドの予測に対する SHAP 値を入力とし，入力検体が AE であるか否かを判定する二値分類ヘッドである．SHAP 値を学習することでモデルの意思決定プロセスの妥当性を基に AE を検知する．

3.2 SHAP 値に基づく AE 検知

画像分野の先行研究において，AE と正常な入力とで，モデルの判断根拠となる解釈パターンが異なる点に着目し，

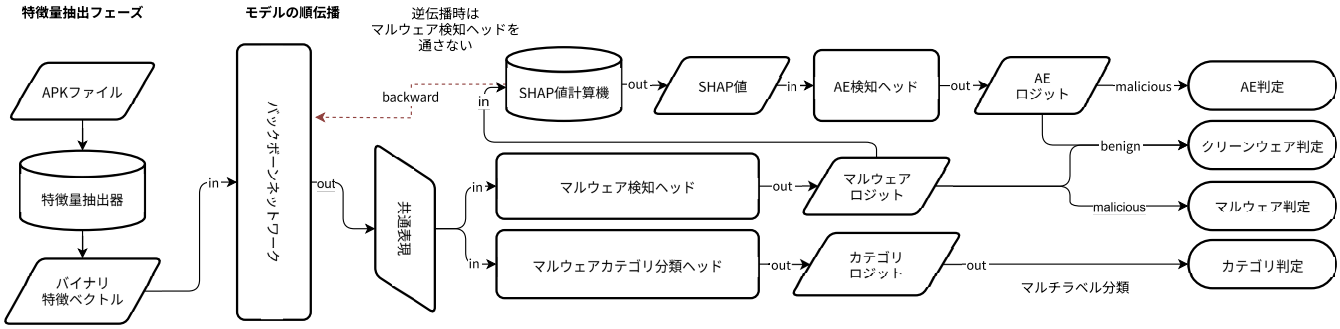

図 1 提案するマルチタスク学習モデルの全体構造

SHAP を用いた AE 検知の有効性が示されている [18], [19]. この解釈パターンの差異は，人間には知覚困難だがモデルの予測に強く相関する”非堅牢な特徴”の存在によるものだと指摘されている [20]．しかし，このアプローチを Android マルウェアの回避攻撃対策へ応用した研究は，我々の知る限り存在しない．マルウェアの AE 生成は，実行可能性や機能性維持の制約から，本来不要な機能の追加などといった特徴操作に限定されやすい．その結果，モデルが挙動に寄与しない作為的な特徴を判断材料とするため，画像分野と同様に解釈パターンが乖離すると我々は仮定する．そこで，本研究では Lin らの研究 [19] に倣い，モデルの判断根拠を包括的に捉えるために複数の中間層から SHAP 値を抽出する手法を採用する．

3.3 マルチタスク学習戦略

3 つの異なるタスクを同時に安定して学習させるため，本研究では統合損失関数，タスク間の競合を緩和する機構と段階的学習スケジュールを導入する．

3.3.1 統合損失関数

モデル全体の損失 L_{total} は，マルウェア検知，カテゴリ分類，AE 検知の損失 L_{MD}, L_{CC}, L_{AED} の重み付き和として定義される．

$$L_{\text{total}} = w_{\text{MD}} L_{\text{MD}} + w_{\text{CC}} L_{\text{CC}} + w_{\text{AED}} L_{\text{AED}}$$

ここで，$w_{\text{MD}}, w_{\text{CC}}, w_{\text{AED}}$ は各タスクの重要度を調整する重み係数である．この重み係数は，3.3.2 項で後述する Uncertainty Weighting によって動的に調整される．

3.3.2 タスク間競合の緩和

マルチタスク学習では，タスク間で勾配の方向が競合し学習が不安定化することがある．本モデルでは，この問題を解決するために 3 つの手法を適用する．第一に，Kendall ら [21] の Uncertainty Weighting により，各タスクの損失の重みを学習可能なパラメータとして扱い，タスクの不確実性に応じて動的に調整する．これにより学習が難しいタスクの重みが自動的に軽減される．第二に，Yu ら [22] の Projecting Conflicting Gradients により，タスク間で競合する勾配成分を除去し，全タスクの性能向上に貢献する更新方向を見つける．第三に，段階的学習スケジュールを採用し，初期エポックではマルウェア検知とカテゴリ分類ヘッドのみを学習し，その後 AE 検知ヘッドの学習を開始する．これにより初期段階でのモデルの安定性を確保し，AE 検知ヘッドが他タスクに悪影響を与えることを防ぐ．

3.4 モデルの統合判定

マルウェア検出ヘッド，カテゴリ分類ヘッド，AE 検知ヘッドの判定結果をそれぞれ H_{MD}，H_{CC}，H_{AED} とする．統合判定は表 2 のように 5 つの判定に分類される．

表 2 モデルの統合判定対応．「—」は don’t care 項を示す．判定列において，MW はマルウェアを示す．

判定番号	$\mathbf{H}_{MD}$	$\mathbf{H}_{CC}$	$\mathbf{H}_{AED}$	統合判定
1	悪性	ラベルなし	—	カテゴリなし MW
2	悪性	ラベル付与	—	カテゴリ付き MW
3	良性	—	良性	クリーンウェア
4	良性	ラベルなし	悪性	カテゴリなし AE
5	良性	ラベル付与	悪性	カテゴリ付き AE

4. 評価実験

本章では，提案手法の有効性を検証するための包括的な実験について述べる．

4.1 データセット

実験では，Androzoo [23] が公開している APK のリストに基づいて，2022 年 4 月から 2024 年 4 月までに VirusTotal で検査された検体を収集した．収集は，Pendlebury らの提言 [24] に基づき，実環境を模してクリーンウェアとマルウェアの比率が 9:1 になるよう行った．

また，悪性のラベル付けには VirusTotal の検知レポートで得られる検知エンジン数 ρ を指標とした．Miller らの研究 [25] に従い，クリーンウェアを $\rho = 0$，マルウェアを $\rho \geq 4$ としてラベル付けした．最終的に，クリーンウェアが 112,835 検体，マルウェアが 13,321 検体の合計 126,156 検体となった．

特徴量には，Android マルウェア分類器に対する回避攻

撃の研究で広く使用される Drebin 特徴量 [26] を使用した．

Drebin 特徴量セットに従い特徴量抽出を行ったところ，126,156 検体から合計 2,475,650 個の特徴量が抽出された．高次元な特徴ベクトルであるため，既存研究 [8] に基づき高頻度上位 10,000 次元の特徴量を選択した．

マルウェアカテゴリ分類に用いるラベルを得るため，AVClass [27] の出力結果を基にマルウェアカテゴリを定義した．その結果，当初 15 種類のマルウェアカテゴリに分類されたが，そのうち検体数が 100 未満の 5 カテゴリ (Cryptominer, Exploitkit, MisleadingScareware, Ransomware, Worm) はデータセットから除外した．最終的にデータセットとして用いた 10 カテゴリとその検体数を表 3 に示す．単一の検体が複数のマルウェアカテゴリに属する可能性があるため，マルチラベル分類を採用する．

表 3 マルウェアカテゴリデータセットの内訳 10 クラス

マルウェアカテゴリ	検体数*1	マルウェアカテゴリ	検体数
Adware	8,658	Downloader	2,231
BackdoorRAT	200	Grayware	11,022
BankingTrojan	451	RiskwarePUA	1,099
Botnet	451	SpywareInfostealer	377
ClickFraud	430	Virus	533
ラベル付き: 12,302		ラベルなし: 1,019	**総検体数**: 13,321

これら合計 126,156 検体の検体をホールドアウト法により，学習用データセット (60%, 75,694 検体), 検証用データセット (20%, 25,231 検体), テスト用データセット (20%, 25,231 検体) に層化分割した．

4.2 学習，評価に用いる攻撃手法

各モデルの評価には，モデルの勾配を利用する Gradient-based Attack 12 種類 [4], [7], [8], [28], [29], [30] およびモデルの勾配を利用しない Gradient-free Attack 4 種類 [8], [31], [32] の計 16 種類を用いた．

これらの多くは，攻撃者がモデルの内部情報を完全に把握していると仮定する，攻撃者にとって最も有利な脅威モデルであるホワイトボックス攻撃に分類される．

提案モデルの学習には，dFGSM^k [7], rFGSM^k [7], Grosse Attack [4], PGD-ℓ_1 [6], [30] の 4 つを用い，評価時にモデルにとって未知の攻撃手法 12 つに対する堅牢性を検証する．

4.3 微分可能な SHAP 値の算出

SHAP 値の計算には，PyTorch の解釈性ライブラリ Captum の GradientSHAP を用いる．ただし，AE 検知ヘッドの学習のため，本来勾配を伝搬しない GradientSHAP を拡張し，SHAP 値の計算過程を含めて計算グラフを構築することで微分可能とした．なお，パラメータ更新時はマルウェア検知ヘッドへの順伝播経路をバイパスし，バックボーンのみを更新することで，マルチタスク学習におけるタスクの独立性を担保する．

4.4 比較対象モデルと提案手法の学習

我々の提案手法と比較するために，以下のモデルを構築，学習した．本研究では，最も広く研究され，多くの派生手法が存在する AT を採用し，公平な比較評価を行う．

Basic DNN: 攻撃手法の有効性を検証するための AT を行わない通常の DNN モデルである．5 層の全結合層から構成され，ReLU 活性化関数と Cross Entropy Loss を用いる．学習率 1×10^{-4}，バッチサイズ 256, エポック数 150 で学習を行い，検証誤差が最小だったチェックポイントを用いた．

AT-rFGSM [7]: Al-Dujaili らが提案した rFGSM^k で AT されたモデルであり，一般的な AT のベースモデルとして用いられる．Li ら [8] の実験に従いモデルと学習のパラメータを決定した．

AT-MA [8]: Mixture of Attacks による AT を行ったモデルである．Li らが提案したモデルであり，複数の攻撃手法を組み合わせ，その中から最も強い AE を選択することでより堅牢なモデルを学習する．Li らの実験に従いモデルと学習のパラメータを採用した．

ADE-MA [8]: Li らが提案した DNN のアンサンブルモデルである．アンサンブルモデル全体の損失と，Mixture of Attacks によって生成された AE に対する損失の両方を最小化するように学習が行われる．Li らの実験に従いモデルと学習のパラメータを採用した．

提案手法: バックボーンが 2 層の全結合層であり，これにマルウェア検知ヘッド，カテゴリ分類ヘッド，AE 検知ヘッドが接続される．各ヘッドはそれぞれ 3 層，4 層，4 層の全結合層で構成される．各ヘッドには ReLU 活性化関数と Cross Entropy Loss を用いる．学習率 5×10^{-5}, バッチサイズ 256, エポック数 150 で学習を行い，検証誤差が最小だったチェックポイントを用いた．

4.5 評価指標

本稿では，提案手法の有効性を多角的に評価する．

4.5.1 検知性能と堅牢性評価の評価

モデルの基本的な検知性能と，回避攻撃に対する堅牢性を評価するため，回避攻撃研究の分野で広く使われている以下の 2 つの指標を用いる．

Natural Accuracy: 摂動が加えられていない通常のテストデータに対するモデルの基本的な検知性能を指す．本稿では，Recall, Precision, F1 スコアを用いて評価する．

Robust Accuracy: 回避攻撃によって生成された AE に対し，モデルが正しく悪性と判断できる割合を示す指標である．分類器 f，攻撃手法を A とする．有限個のテストセット $S = \{(x_i, y_i)\}_{i=1}^{n}$ が与えられたとき，攻撃 A に対

する分類器 f の Robust Accuracy は次のように表される．

$$\mathrm{RA}_{S,A}(f) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\Big[f\Big(A(x_i, y_i, f)\Big) = y_i\Big]$$

4.5.2 マルウェアカテゴリ分類性能の評価

本稿では，不均衡なマルチラベル分類のため，少数カテゴリの性能を公平に評価する Macro-Averaged Precision, Recall を使用する．なお，これらの指標の計算には scikit-learn ライブラリの標準実装を用いた．

5. 実験結果と考察

本章では，提案手法の有効性を検証した実験結果を示し，2.2 節で定義した 4 要件の達成状況を考察する．

5.1 要件 (i) 通常検体の検知精度の維持

表 4 に各モデルの Natural Accuracy を示す．提案手法は F1 スコア 0.9986 を達成し，Basic DNN と同等以上の精度を維持した．一方で，AT モデルは Natural Accuracy が低下しており，特に AT-MA は Precision が 0.8915 まで低下した．

従来の AT が抱えるこの課題は，正常検体と AE という本質の異なる 2 つのデータ分布を，単一の決定境界で分離することの難しさに起因すると考えられる．AT は AE に対する堅牢性を高める一方で，決定境界を正常検体にとっての最適位置から歪ませる可能性がある．これに対し，本稿で提案するマルチタスク学習モデルは，この問題を”Decoupling”[33] というアプローチで解決する．AE 検知を独立タスクとして分離することで，マルウェア検知ヘッドは通常検体に対する最適な決定境界を維持でき，AT が抱える精度と堅牢性のトレードオフ [5] を回避した．これにより，提案手法は要件 (i) を満たした．

表 4 各モデルの Natural Accuracy

Model	Recall	Precision	F1
Basic DNN	0.9986	0.9985	0.9985
AT-rFGSM [7]	0.9981	0.9966	0.9973
AT-MA [8]	0.9985	0.8915	0.9419
ADE-MA [8]	0.9988	0.8954	0.9443
提案手法	0.9986	0.9987	0.9986

5.2 要件 (ii) 未知の攻撃に対する堅牢性

表 5 に 16 種類の攻撃手法に対する Robust Accuracy を示す．

提案手法の総合判定 (Our_{INT}) の Robust Accuracy の平均は 0.9901 であり，比較対象のすべてのモデルを上回っている．マルウェア検知ヘッド単体 (Our_{MD}) においては平均 0.2184 と Basic DNN と同水準であるが，この検知を回避した AE を，後段の AE 検知ヘッド (Our_{AED}) が平均 0.9891 という極めて高い精度で検出している．さらに，本モデルは学習していない 12 種類の未知の攻撃全てに対し，0.9643 以上の Robust Accuracy を維持した．

この汎化性能の差異は，検知アプローチの根本的な違いに起因すると考えられる．AT が AE のデータそのものを学習対象とするのに対し，提案手法は AE がモデルに与える影響，すなわち判断根拠の歪みを SHAP 値を通じて検知する．マルウェアの AE 生成は，機能維持という制約から挙動に寄与しない作為的な特徴操作に限定されやすく，これは攻撃手法の詳細によらず共通の判断根拠の歪みを生む．

提案手法は，この攻撃手法に依存しない普遍的な副作用を捉えるため，未知の攻撃に対しても高い汎化性能を発揮し，要件 (ii) を満たした．

5.3 要件 (iii) AE 検知後の迅速なインシデント対応

AE として検知された検体 (表 2 の判定 5 に該当) に対するカテゴリ分類性能を表 6 に示す．

表 6 の通り，AE に対するカテゴリ分類で平均 Precision 0.7524, Recall 0.8129 という高い性能を維持した．

これにより，本手法は単なる悪性という判定に留まらず，Banking Trojan か Adware かといった具体的な脅威情報を提供できる．この機能はアナリストの迅速な意思決定を直接支援するものであり，要件 (iii) を満たした．

5.4 要件 (iv) SHAP 値に基づく解析補助

提案手法が要件 (iv) にどう貢献するかを例証するため，Grosse Attack [4] がモデルの判断根拠に与える影響を AE 検知ヘッドに入力された SHAP 値を用いて追加分析を行った．分析の結果，攻撃がモデルの学習上の脆弱性を突き，本来は悪性度と無関係な特徴量の重要性を，良性方向へ意図的に増大させるという顕著なパターンが明らかになった．

SHAP 値の変動が特に大きかった特徴は, 著名な認証 SDK の開発者向けドキュメントを指す URL, プライベート IP アドレス (`urls::10.0.9.1`, `urls::13.0.1.0`)，公式ハードウェア機能 (`features::android.hardware.faketouch`) などであった．これらは実際には悪性判定とは無関係な情報であり，モデルは本来の特徴量の意味を無視し，多くの正規アプリに共通する特徴とクリーンウェアであることを疑似相関として学習してしまったと考えられる．

本分析の価値は，単に攻撃の痕跡を可視化するに留まらない．SHAP を組み込むことで，我々の仮定であった”非堅牢な特徴”の存在を具体的に特定し，モデルの脆弱性を突き止めることができる点にある．この知見は，的を絞った攻撃シグネチャの作成や，これら”非堅牢な特徴”に依存しないモデルへの再設計といった，実践的な防御策に直接繋がる．以上より，提案手法はアナリストに実用的な解析支援を実現するものであり，要件 (iv) を満たした．

表 5 各攻撃手法に対する各モデルの Robust Accuracy．下線はそのモデルが学習に用いた攻撃手法を示す．Mixture of Attack は内部で PGD-ℓ_p を用いるため，破線で示す．

攻撃手法	Basic DNN	AT-rFGSM [7]	AT-MA [8]	ADE-MA [8]	Our$_{MD}$	Our$_{AED}$	Our$_{INT}$
dFGSM^k [7]	0.0586	0.9955	0.9981	**0.9985**	0.0933	0.9970	0.9977
rFGSM^k [7]	0.0586	**0.9977**	0.9831	0.9962	0.0000	0.9797	0.9793
BGA^k [7]	0.0000	0.9992	**1.0000**	0.9996	0.0000	**1.0000**	**1.0000**
BCA^k [7]	0.6561	0.9936	0.9989	0.9992	0.0376	0.9985	**1.0000**
Grosse Attack [4]	0.0579	0.9271	0.9932	0.9767	0.0060	**1.0000**	**1.0000**
JSMA [28]	0.2841	0.9274	0.9932	0.9767	0.0060	**1.0000**	**1.0000**
GDKDE [29]	0.0000	0.9857	**0.9992**	0.9977	0.3789	0.9650	0.9684
PGD-ℓ_1 [6], [30]	0.0586	0.2184	0.9719	0.9625	0.0060	**1.0000**	**1.0000**
PGD-ℓ_2 [6], [30]	0.5787	0.8639	**0.9974**	0.9966	0.2190	0.9695	0.9710
PGD-ℓ_∞ [6], [30]	0.0586	0.2556	**0.9951**	0.9921	0.6373	0.9827	0.9876
PGD-Adam [30]	0.8162	0.9387	0.9711	0.9643	0.0726	**0.9970**	0.9959
Mixture of Attack [8]	0.0586	0.1906	0.9523	0.9336	0.0000	**0.9955**	0.9951
Random Attack	0.8005	**1.0000**	0.9981	0.9996	0.9090	0.9959	0.9992
Salt and Pepper [8], [32]	0.0060	**1.0000**	**1.0000**	**1.0000**	0.8529	**1.0000**	**1.0000**
Mimicry Attack [31]	0.2514	0.8846	0.5550	0.7376	0.1377	0.9575	**0.9643**
Pointwise [8]	0.2488	0.8827	0.5505	0.7462	0.1377	**0.9868**	0.9828
平均	0.2495	0.8163	0.9348	0.9548	0.2184	0.9891	**0.9901**

表 6 AE に対するマルウェアカテゴリ分類性能

攻撃種別	Recall$_{macro}$	Precision$_{macro}$
Gradient-based (11 種)	0.8476	0.7629
Gradient-free (3 種)	0.7064	0.7211
全体平均	**0.8129**	**0.7524**

5.5 研究倫理

本研究における倫理的側面について，その社会的影響と実験手順の正当性の観点から検討した．

本稿は，巧妙化する回避攻撃に対する新しい防御手法を提案するものである．手法の公開には，攻撃者による分析リスクが常に伴うが，本稿が提供する新しい防御アーキテクチャと攻撃解析の知見は，コミュニティ全体の防御能力を高めるという公益性が上回ると判断した．また，本研究は特定の商用アンチマルウェア製品を対象としておらず，その影響は限定的である．

さらに，本稿における実験手順の正当性についても検討した．実験で使用したクリーンウェアおよびマルウェアは，公開されている学術研究用データセット Androzoo [23] から入手した．マルウェアのラベル付けには，VirusTotal および AVClass [27] を用いており，透明性と再現性を担保している．回避攻撃の生成を含む全ての実験は，外部から隔離された安全な研究環境で実施したため，第三者の製品やサービス，ネットワークに対して攻撃が行われることは一切ない．

6. おわりに

本研究では，Android マルウェア検知器への回避攻撃という脅威に対し，SHAP とマルチタスク学習を統合した新たな防御手法を提案した．評価実験を通じ，本手法がマルチタスク学習によるタスクの分離によって Natural Accuracy を維持しつつ，攻撃手法に依存しない判断根拠の歪みを検知することで未知の攻撃にも高い汎化性能を示すことを実証した．さらに，AE 検知時にマルウェアカテゴリを復元し，SHAP で攻撃の痕跡を可視化することで，実用的なトリアージと解析支援が可能であることを確認した．

本研究の貢献は，堅牢な検知と実用的な解析支援を単一のフレームワークで両立させた点にある．今後の課題として，複数のデータセットを用いたさらなる検証や，最新のブラックボックス攻撃に対しても SHAP が有効であるかの検証が挙げられる．

参考文献

[1] AV-TEST: “TOTAL AMOUNT OF MALWARE AND PUA UNDER ANDROID” (2025).

[2] Aslan, c. A. and Samet, R.: A Comprehensive Review on Malware Detection Approaches, *IEEE Access*, Vol. 8, pp. 6249–6271 (2020).

[3] Goodfellow, I. J., Shlens, J. and Szegedy, C.: Explaining and Harnessing Adversarial Examples (2015).

[4] Grosse, K., Papernot, N., Manoharan, P., Backes, M. and McDaniel, P.: Adversarial Examples for Malware Detection, *Computer Security – ESORICS 2017* (Foley, S. N., Gollmann, D. and Snekkenes, E., eds.), Cham, Springer International Publishing, pp. 62–79 (2017).

[5] Tramèr, F. and Boneh, D.: *Adversarial training and robustness for multiple perturbations*, Curran Associates Inc. (2019).

[6] Madry, A., Makelov, A., Schmidt, L., Tsipras, D. and Vladu, A.: Towards Deep Learning Models Resistant to Adversarial Attacks (2019).

[7] Al-Dujaili, A., Huang, A., Hemberg, E. and OReilly, U.-M.: Adversarial Deep Learning for Robust Detection of Binary Encoded Malware , *2018 IEEE Security and Privacy Workshops (SPW)*, Los Alamitos, CA, USA, IEEE Computer Society, pp. 76–82 (2018).

[8] Li, D. and Li, Q.: Adversarial Deep Ensemble: Evasion Attacks and Defenses for Malware Detection, *IEEE Transactions on Information Forensics and Security*, Vol. 15, pp. 3886–3900 (2020).

[9] Wang, C., Zhang, L., Zhao, K., Ding, X. and Wang, X.: AdvAndMal: Adversarial Training for Android Malware Detection and Family Classification, *Symmetry*, Vol. 13, No. 6 (2021).

[10] Lucas, K., Pai, S., Lin, W., Bauer, L., Reiter, M. K. and Sharif, M.: Adversarial Training for Raw-Binary Malware Classifiers, *32nd USENIX Security Symposium (USENIX Security 23)*, Anaheim, CA, USENIX Association, pp. 1163–1180 (2023).

[11] Li, D., Cui, S., Li, Y., Xu, J., Xiao, F. and Xu, S.: PAD: Towards Principled Adversarial Malware Detection Against Evasion Attacks, *IEEE Transactions on Dependable and Secure Computing*, Vol. 21, No. 2, pp. 920–936 (2024).

[12] Zhou, Y., Cheng, G., Chen, Z. and Yu, S.: MalPurifier: Enhancing Android Malware Detection with Adversarial Purification against Evasion Attacks (2025).

[13] Zhang, Y., Gao, C., Wu, Y., Dou, S., Wu, C., Zhang, Y., Yuan, W. and Liu, Y.: Fighting Fire with Fire: Continuous Attack for Adversarial Android Malware Detection, *Proceedings of the 34th USENIX Security Symposium*, Seattle, WA, USA, USENIX (2025).

[14] Li, H., Zhou, S., Yuan, W., Luo, X., Gao, C. and Chen, S.: Robust Android Malware Detection against Adversarial Example Attacks, *Proceedings of the Web Conference 2021*, WWW '21, New York, NY, USA, Association for Computing Machinery, p. 3603–3612 (2021).

[15] Melis, M., Scalas, M., Demontis, A., Maiorca, D., Biggio, B., Giacinto, G. and Roli, F.: Do Gradient-based Explanations Tell Anything About Adversarial Robustness to Android Malware? (2021).

[16] Tsipras, D., Santurkar, S., Engstrom, L., Turner, A. and Madry, A.: Robustness May Be at Odds with Accuracy, *International Conference on Learning Representations* (2019).

[17] Caruana, R.: *Multitask learning*, p. 95–133, Kluwer Academic Publishers (1998).

[18] Fidel, G., Bitton, R. and Shabtai, A.: When Explainability Meets Adversarial Learning: Detecting Adversarial Examples using SHAP Signatures, *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8 (2020).

[19] Lin, Y.-C. and Yu, F.: DeepSHAP Summary for Adversarial Example Detection, *2023 IEEE/ACM International Workshop on Deep Learning for Testing and Testing for Deep Learning (DeepTest)*, pp. 17–24 (2023).

[20] Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B. and Madry, A.: Adversarial Examples Are Not Bugs, They Are Features, *Advances in Neural Information Processing Systems* (Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. and Garnett, R., eds.), Vol. 32, Curran Associates, Inc. (2019).

[21] Kendall, A., Gal, Y. and Cipolla, R.: Multi-task learning using uncertainty to weigh losses for scene geometry and semantics, *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7482–7491 (2018).

[22] Yu, T., Kumar, S., Gupta, A., Levine, S., Hausman, K. and Finn, C.: Gradient Surgery for Multi-Task Learning, *Advances in Neural Information Processing Systems* (Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M. and Lin, H., eds.), Vol. 33, Curran Associates, Inc., pp. 5824–5836 (2020).

[23] Allix, K., Bissyandé, T. F., Klein, J. and Le Traon, Y.: AndroZoo: Collecting Millions of Android Apps for the Research Community, *Proceedings of the 13th International Conference on Mining Software Repositories*, MSR '16, New York, NY, USA, ACM, pp. 468–471 (2016).

[24] Pendlebury, F., Pierazzi, F., Jordaney, R., Kinder, J. and Cavallaro, L.: TESSERACT: Eliminating Experimental Bias in Malware Classification across Space and Time, *28th USENIX Security Symposium (USENIX Security 19)*, Santa Clara, CA, USENIX Association, pp. 729–746 (2019).

[25] Miller, B., Kantchelian, A., Tschantz, M. C., Afroz, S., Bachwani, R., Faizullabhoy, R., Huang, L., Shankar, V., Wu, T., Yiu, G., Joseph, A. D. and Tygar, J. D.: Reviewer Integration and Performance Measurement for Malware Detection, *Proceedings of the 13th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment - Volume 9721*, DIMVA 2016, Berlin, Heidelberg, Springer-Verlag, p. 122–141 (2016).

[26] Arp, D., Spreitzenbarth, M., Hübner, M., Gascon, H. and Rieck, K.: DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket (2014).

[27] Sebastián, S. and Caballero, J.: AVclass2: Massive Malware Tag Extraction from AV Labels, *Proceedings of the 36th Annual Computer Security Applications Conference*, ACSAC '20, New York, NY, USA, Association for Computing Machinery, p. 42–53 (2020).

[28] Chen, X., Li, C., Wang, D., Wen, S., Zhang, J., Nepal, S., Xiang, Y. and Ren, K.: Android HIV: A Study of Repackaging Malware for Evading Machine-Learning Detection, *IEEE Transactions on Information Forensics and Security*, Vol. 15, pp. 987–1001 (2020).

[29] Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G. and Roli, F.: Evasion Attacks against Machine Learning at Test Time, *Machine Learning and Knowledge Discovery in Databases* (Blockeel, H., Kersting, K., Nijssen, S. and Železný, F., eds.), Berlin, Heidelberg, Springer Berlin Heidelberg, pp. 387–402 (2013).

[30] Li, D., Li, Q., Ye, Y. and Xu, S.: A Framework for Enhancing Deep Neural Networks Against Adversarial Malware, *IEEE Transactions on Network Science and Engineering*, Vol. 8, No. 1, pp. 736–750 (2021).

[31] Demontis, A., Melis, M., Biggio, B., Maiorca, D., Arp, D., Rieck, K., Corona, I., Giacinto, G. and Roli, F.: Yes, Machine Learning Can Be More Secure! A Case Study on Android Malware Detection, *IEEE Transactions on Dependable and Secure Computing*, Vol. 16, No. 4, pp. 711–724 (2019).

[32] Schott, L., Rauber, J., Bethge, M. and Brendel, W.: Towards the first adversarially robust neural network model on MNIST (2018).

[33] Wang, H. and Wang, Y.: Generalist: Decoupling Natural and Robust Generalization, *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20554–20563 (2023).