Computer Security Symposium 2019
21 - 24 October 2019

Androidマルウェア分類器に対するブラックボックス型回避攻撃手法の検討

古川 和祈[1,a)]　畑田 充弘[2]　吉浦 裕[1]　市野 将嗣[1]

概要：スマートフォンの普及に伴い，スマートフォンを対象としたマルウェアが増加している．そこで未知のマルウェアを検知できるように，機械学習を用いた検知手法が提案されるようになった．他方で近年，機械学習に対する攻撃手法が多く提案されている．機械学習に対する攻撃手法の 1 つに回避攻撃というものが存在する．回避攻撃は，あるクラスに分類される入力に対して摂動を与えることで，別のクラスに分類されるようにする攻撃である．マルウェア検知に対する回避攻撃が実現すると，機械学習を用いることで検知することが出来ていた未知マルウェアの検知が困難となる脅威が生まれ，この脅威に対する防御手法を確立することが求められる．本研究では，標的モデルが用いている正確な特徴量の情報を必要としないブラックボックス型の回避攻撃手法を提案する．評価実験では，約 80%のマルウェアを良性として回避できることを確認した．

キーワード：Android マルウェア, 回避攻撃, 特徴量, 強化学習

Investigation of Black-Box Based Evasion Attack Method Against Android Malware Classifier

Kazuki Furukawa[1,a)]　Mitsuhiro Hatada[2]　Hiroshi Yoshiura[1]　Masatsugu Ichino[1]

Abstract: For dealing with a large amount of malware, detection methods using machine learning are useful. On the other hand, one of the attack methods called evasion attack is proposed. Evasion attack makes it possible to be classified into another class by adding a perturbation to the input classified into one class. When an evasion attack on a malware classifier is realized, there is a threat that makes it difficult to detect unknown malware that could be detected using machine learning, and it is required to establish a defense method against the threat. In this research, we propose a black-box based evasion attack technique that does not require exact feature information used by a target model. In the experiment, it was confirmed that about 80% of malwares can be avoided as benign.

Keywords: Android Malware, Evasion Attack, Features, Reinforcement Learning

1. はじめに

スマートフォンの普及に伴って，スマートフォンを対象としたマルウェアが増加している．増加するマルウェアへの対策として，マルウェア検知の必要性が高まっている．従来のシグネチャ型検知の欠点である未知マルウェア検知において，機械学習を用いたマルウェア検知が広く使われるようになった．

他方で近年，Adversarial Examples をはじめとし，機械学習に対する攻撃手法が多く提案されている．機械学習に対する攻撃手法の 1 つに回避攻撃というものが存在する．回避攻撃は，あるクラスに分類される入力に対して摂動を与えることで，別のクラスに分類されるようにする攻撃である．この攻撃を機械学習を用いたマルウェア検知に適用

[1] 電気通信大学
The University of Electro-Communications
[2] 日本電信電話株式会社
Nippon Telegraph and Telephone Corporation
[a)] k-furukawa@uec.ac.jp

する研究も提案されるようになった．

マルウェア検知に対して回避攻撃を適用すると，悪性ファイルとして検知されたマルウェアを良性に誤分類するように仕向けることが可能となる．よって，実際にマルウェア検知に対する回避攻撃が実現すると，機械学習を用いることで検知することが出来ていた未知マルウェアの検知が困難となる脅威が存在する．

Android マルウェア分類器に対する回避攻撃の研究では勾配型攻撃の研究が主流だが，スコア型攻撃やブラックボックス型攻撃の研究も行われている．

Wang らは，Adversarial Examples が持つ Transferability という性質を用いて Android マルウェア分類器に対しブラックボックス型攻撃を行う手法を提案した [9]．Transferability とは，あるモデルを元に作成した Adversarial Examples は同じタスクを学習した異なるモデルに対しても攻撃が成功するという性質である．Wang らの手法は標的モデルに関する情報を必要としない手法であるが，識別器が用いている正確な特徴量を既知としていた．そこで本研究では，強化学習を適用することで前提知識として識別器が用いている正確な特徴量を必要としない回避攻撃手法を実現した．

本研究の貢献をまとめると，以下の通りである．

- 識別器の用いている正確な特徴量を攻撃者が知らなくても攻撃可能な手法を提案した．
- 提案手法は強化学習を用いることで任意のモデルを攻撃可能である．
- 実験を通して，提案手法によって線形 SVM を用いた Android マルウェア分類器に対して回避攻撃が成功することを確認した．

本稿の構成を以下に示す．第 2 章は関連研究を紹介する．第 3 章は，提案手法について説明する．第 4 章は評価実験の手法およびその結果について説明する．第 5 章は，検証実験の結果と提案手法に関する考察を行う．最後に第 6 章は本稿のまとめを述べる．

2. 関連研究

2.1 回避攻撃

回避攻撃とは，あるクラスに分類される入力に対して摂動を与えることで，別のクラスに分類されるようにする攻撃である．

近年は特に Deep Neural Network(以下 DNN) を用いた分類器に対する回避攻撃が研究されている．Szegedy らは画像分類を行う DNN に対して人間が認識できないような小さな摂動を適切に選び，入力に与えると回避攻撃が実現できることを示し，この摂動を与えた入力のことを Adversarial Examples と定義した [8]．

Anderson らは，機械学習を用いたマルウェア検知に対する回避攻撃手法を，攻撃者が得られる識別器の情報の量に応じて以下の 3 種類に分類した [1]．

(1) 勾配型攻撃
識別器の構造やパラメータを攻撃者が持っていることを前提とした攻撃．攻撃者は識別器のモデルの情報を元にして，入力に与える摂動を求めることが可能である．また，標的モデルは微分可能なモデルである必要がある．

(2) スコア型攻撃
識別結果に関するスコアを出力する識別器に対する攻撃．攻撃者は識別器の構造やパラメータに関する知識は無いが，任意の入力に対する識別結果を得ることが可能である．

(3) ブラックボックス型攻撃
識別結果が良性か悪性かを示す単一ビットの識別器に対する攻撃．スコア型攻撃と同様，攻撃者は識別器に関する知識は無い．また，任意の入力に対する識別結果を得ることが可能である．

2.2 Android マルウェア分類器に対する回避攻撃

先行研究では静的解析によって得られる二値特徴量を用いた二値分類器に対する回避攻撃手法が提案されてきた [3][9]．二値特徴量とは，ある特徴量についてサンプルがその条件を満たすか，満たさないかを二値で示した特徴量を指す．

Grosse らは, DNN を用いた Android マルウェア分類器に対する Jacobian Based Saliency Map Attacks(以下 JSMA) をベースとした勾配型攻撃手法を提案した [3]．JSMA は Papernot らが提案した DNN を用いた識別器に対する勾配型攻撃手法であり，DNN の勾配をもとに，入力のうち出力を変更しやすい変数を探し出して変更するという手法である．JSMA は画像分類のモデルを想定して作成されたため，そのままマルウェア分類のモデルに適用すると，破壊的な特徴量変更を引き起こす可能性が高い．破壊的な特徴量変更とは，変更後の特徴量を元にマルウェアを再構築することができないような変更，又は再構成できたとしてもマルウェアとしての機能が果たせなくなってしまうような変更を指す．Grosse らはこの破壊的な特徴量変更を防ぐため，入力へ変更を適用する際に 0 から 1，つまりマルウェアの機能が削減されない変更に制限することで，マルウェアの動作可能性を保証している．

Liu らは，Android マルウェア分類器に対するスコア型攻撃手法を提案した [4]．この手法では，入力に対する摂動を識別器の出力を元に評価を行う遺伝的アルゴリズムで探索することで Adversarial Examples を作成する．このとき，Grosse らの手法と同様にマルウェアの機能が削減されない変更に制限することで，マルウェアの動作可能性を保証している．

Wang らは，Adversarial Examples が持つ Transferabil-

表 1　関連研究との比較
Table 1　Comparison with related works.

	勾配型	スコア型	ブラックボックス型	本研究
制約 1	Yes	No	No	No
制約 2	Yes	No	No	No
制約 3	No	Yes	No	No
制約 4	Yes	Yes	Yes	No

ity という性質を用いて Android マルウェア分類器に対しブラックボックス型攻撃を行う手法を提案した [9]．Transferability は Goodfellow らが 2014 年に発見し [2]，Papernot らによって DNN だけでなく線形回帰や決定木といったモデル間でも起こることが分かっている [6]．Wang らは DNN を代理モデルとしてマルウェア分類の学習を行い，JSMA を用いて代理モデルを元に Adversarial Examples を作成し，DNN や決定木，SVM 等のモデルに対して回避攻撃を行った．Wang らの手法は標的モデルの構造やパラメータを必要とせず，また識別器の出力は二値なのでブラックボックス型攻撃に分類される．

Liu らの手法と Wang らの手法において，標的モデルの構造やパラメータは必要とされないが，どちらの手法も標的モデルの用いている正確な特徴量を攻撃者が知っているという制約が暗黙的に存在する．

2.3　本研究の位置づけ

本研究はブラックボックス型攻撃に分類される攻撃手法を提案する．

関連研究と本研究の差異を識別器の構造やパラメータが必要（制約 1），識別器が微分可能なモデルである必要（制約 2）識別器の出力がスコアである必要（制約 3），識別器が用いている正確な特徴量の情報が必要（制約 4）という 4 つの観点で比較した結果を**表 1** に示す．

本研究の攻撃の前提条件は Grosse らの勾配型攻撃，Liu らのスコア型攻撃，Wang らのブラックボックス型攻撃と比べ，より現実的な攻撃の前提条件であり，既存研究と異なる．

また多くの論文では，明示的に識別器の学習に用いたデータセットと攻撃器の学習に用いたデータセットとを切り分けていなかった．よって，本研究では攻撃者が識別器の学習データセットを手に入れられないという条件で評価を行った．

3.　提案手法

本研究では標的モデルが用いている正確な特徴量の情報を必要としないブラックボックス型の回避攻撃手法を提案する．

先行研究 [9] では暗黙的に攻撃者が識別器の正確な特徴量を知っているという前提を置いていた．そこで，提案手法では攻撃者が識別器が用いている特徴量を含んだ特徴量の候補の集合を知っているという前提を置いた．この識別器が用いている特徴量を含んだ特徴量の候補の集合を特徴量候補と呼ぶこととする．**図 1** に先行研究と提案手法のそれぞれにおいて，攻撃者が持つ識別器の特徴量情報の比較した結果を示す．先行研究において攻撃者は実線で示した識別器の特徴量と完全に同じ集合の情報を持っているが，提案手法では破線で示した特徴量候補の情報を持っている．

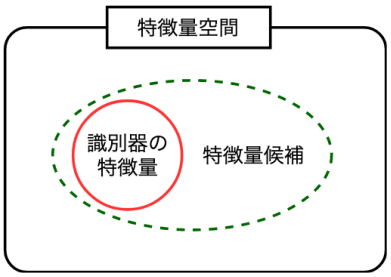

図 1　攻撃者が持つ識別器の特徴量情報の比較
Fig. 1　Comparison knowledge of features.

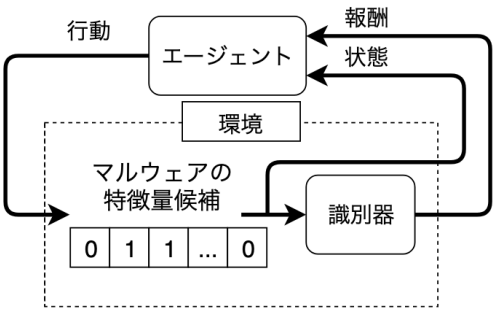

図 2　提案手法の概要
Fig. 2　Overview of proposed method.

図 2 に提案手法の概要を示す．提案手法は強化学習を用いて回避攻撃を起こすような摂動を作るエージェントを学習させることで回避攻撃を実現する．

強化学習とは，エージェントと環境で構成される機械学習モデルである．各ターン t において，エージェントは戦略 $\pi(\boldsymbol{a}|\boldsymbol{s}_t)$ と環境状態ベクトル $\boldsymbol{s}_t$ をもとに，行動 $\boldsymbol{a}_t \in \mathcal{A}$ を選択する．環境は，選ばれた行動と新しい環境状態ベクトル $\boldsymbol{s}_{t+1}$ に応じて報酬 $r_t \in \mathbb{R}$ を返す．報酬 r_t および観測された環境の状態 $\boldsymbol{s}_{t+1}$ は，戦略 $\pi(\boldsymbol{a}|\boldsymbol{s}_{t+1})$ に基づいて新しい行動を選択するためにエージェントへフィードバックされる．エージェントは探索を通じて，環境の状態を考慮してどの行動を起こすべきなのかを徐々に学習する．エージェントの目的は，$Q^\pi(\boldsymbol{s}_t, \boldsymbol{a}_t) = \mathbb{E}_{\boldsymbol{s}_{t+1:\infty}, \boldsymbol{a}_{t+1:\infty}}[R_t|\boldsymbol{s}_t, \boldsymbol{a}_t]$ と $R_t = \sum_{i\geq 0}\gamma^i r_{t+i}$ をもとに，期待収益 $V^\pi(\boldsymbol{s}_t) = \mathbb{E}_{\boldsymbol{a}_t}[Q^\pi(\boldsymbol{s}_t, \boldsymbol{a}_t)|\boldsymbol{s}_t]$ を最大化する戦略を見つけることである．

提案手法では，各ターンにおいてエージェントはどの特徴量候補のどの変数を変更するか選択する．このとき，既に 1 を示している特徴量が選ばれた場合は変更を行わな

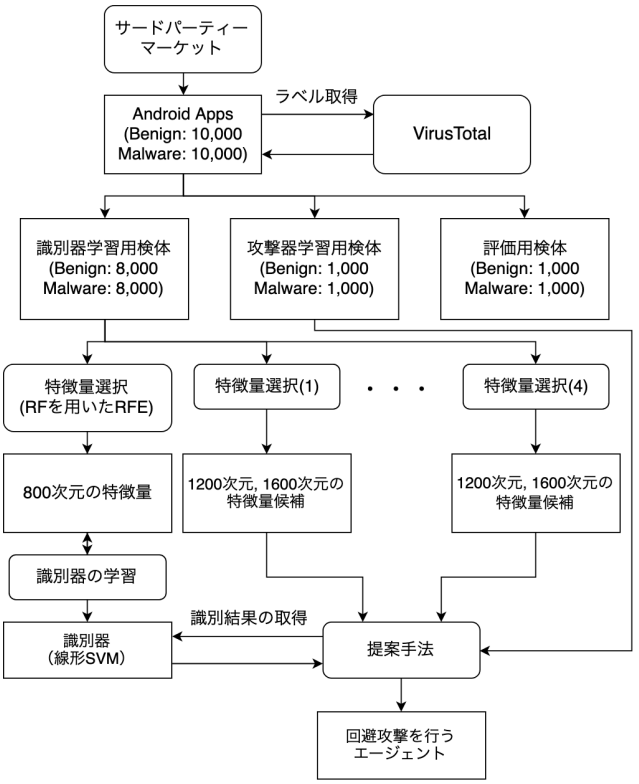

図 3 評価実験の概要
Fig. 3 Overview of experiment.

い．よって，マルウェアの機能が削減されることが無く，マルウェアの動作可能性が保証される．この行動が反映されたマルウェアの特徴量をエージェントが状態として受け取る．また，その特徴量を入力として受け取った識別器の識別結果をエージェントが報酬として受け取る．このプロセスを複数ラウンド進めることでエージェントは回避攻撃を起こしやすい特徴量候補を見つけ，回避攻撃を起こすような摂動を作ることができるようになる．

4. 評価実験

提案手法の回避攻撃性能を評価するため，評価実験を行った．評価実験の概要を**図 3** に示す．

4.1 データセット

評価実験で用いるデータセットは，5 つのサードパーティーマーケットから Android アプリケーションを収集した．VirusTotal[11] を用いて，収集したアプリケーションに対して良性か悪性かのラベル付けを行った．

実験に際して，マルウェアを 10,000 検体，良性アプリケーションを 10,000 検体用意した．これのうち，80%を識別器の学習，10%を攻撃の学習，10%を識別器の評価及び攻撃の評価に用いた．

データセットのマルウェア 10,000 検体に対して VirusTotal が示した検知名の出現数の上位 30 件を**表 2** に示す．

表 2 データセットのマルウェアに対する検知名の出現数
Table 2 Number of malware labels in dataset.

VirusTotal での検知名	出現数
AppRisk:Generisk	2729
a variant of Android/Packed.TencentProtect.B potentially unsafe	1335
AdLibrary:Igexin	1151
Trojan/Android.TSGeneric	926
Adware.Gexin.2.origin	842
AdLibrary:Generisk	754
Trojan.AndroidOS.Generic.C!c	391
AdLibrary:Jpush	350
Android.Malware.General (score:9)	207

4.2 特徴量

先行研究 [9] の評価実験では次の 6 種類の静的特徴量を用いて識別器の学習が行われた．また，これらの特徴量は対象のアプリケーションに存在するかどうかを示す二値表現の特徴量である．

(1) Permissions
Android はデフォルトで，すべてのアプリケーションが他のアプリケーションや OS，ユーザーに悪影響を及ぼすような操作を行うための権限を持っていない設計である．もし，ユーザーの個人情報（連絡先や E メール情報）や他のアプリケーションファイルの読み取りまたは書き込み，ネットワークアクセスといった権限が必要な場合にはアプリケーションが明示的にユーザー及び端末に権限を要求する必要がある，これらの権限を要求しているか，いないかを特徴量として扱う．

(2) Filtered Intents
アクティビティやサービスが取り扱うことができるインテントを指定するもの．インテントとは Android においてアプリケーション間で連携を行うために用いられる機構である．アクション，カテゴリ，データ (URI) をフィルタとして指定することが出来る．特定のアクションをフィルタするか，しないかを特徴量として扱う．

(3) Application Attributes
AndroidManifest と呼ばれるアプリケーションの設定ファイルで指定でできる，debuggable や description といったアプリケーションに関する設定項目のこと．それぞれの設定について行われているか，いないかを特徴量として扱う．

(4) API calls
Android がデフォルトで提供しているライブラリの特定の関数を呼び出しているか，いないかを特徴量として扱う．

(5) New-Instances
Android がデフォルトで提供しているライブラリの特

表 3　評価実験に使用した特徴量選択アルゴリズム
Table 3　Feature selection algorithms used in the experiment.

特徴量選択 (1)	カイ二乗値を用いた Filter Method
特徴量選択 (2)	ANOVA の F 値を用いた Filter Method
特徴量選択 (3)	線形 SVM を推定器とした RFE
特徴量選択 (4)	Variance Threshold

定のクラスのインスタンスを作成しているか，いないかを特徴量として扱う．

(6) Exceptions
特定の Exception がエラーハンドルされるか，されないかを特徴量として扱う．

そこで，本研究では学習データセットとしたアプリケーションに存在する上記の特徴量を全て抽出し，それらに対して特徴量選択を行うことで，800 次元の特徴量を識別器の学習に用いることとした．また，識別器が用いる 800 次元の特徴量に対し，異なる特徴量選択で取得した相異なる特徴量を加えることで特徴量候補を用意した．

4.3　特徴量選択

学習データセットから取得した全特徴量は 1,384,543 次元だった．そこで，単変量統計に基づいて絞り込んだ後，Random Forest(以下 RF) を推定器とした Recursive Feature Elimination(以下 RFE) を用いて 800 次元の特徴量を選ぶこととした．単変量統計における各特徴量の評価指標には分散を用いた．つまり，学習データセットの 99 %以上のサンプルで 1 または 0 を示す特徴量を削減した．この結果，特徴量の候補を 1,384,543 次元から 49,762 次元に絞り込んだ．次に，RF を推定器とした RFE を用いて 49,762 次元の特徴量を 800 次元の特徴量に絞り込んだ．

特徴量候補の作成に際して，複数の特徴量選択アルゴリズムで作成し評価することで異なる特徴量候補を用いた場合の性能評価を行うこととした．特徴量候補の作成に用いた特徴量選択アルゴリズムを**表 3** に示す．これらのアルゴリズムを用いて 400 次元と 800 次元の特徴量を選択した．次に識別器の学習に用いる 800 次元と併せて 1200 次元と 1600 次元の特徴量候補とした．

特徴量選択アルゴリズムの実装は機械学習ライブラリの Scikit-learn[7] のものを利用した．

4.4　識別器及び強化学習アルゴリズム

標的モデルには線形 SVM を利用した．線形 SVM は多くの先行研究で評価実験に用いられており，マルウェア分類の研究においても広く使われていることから妥当であると判断した．

また，提案手法における強化学習のアルゴリズムには Actor-critic with experience replay(以下 ACER) を用いた．ACER は，各状態に対する状態行動価値を推定するための戦略モデル π および Q 関数の両方を学習に DNN を利用する．並列化が可能である点や，エージェントが比較的少数の経験から効率的に学習するのに有効的である点を考慮し，ACER を評価で用いる強化学習アルゴリズムとした．

また，強化学習を用いて学習を行う際のパラメータとしてターンの最大数とラウンド数（総学習ターン数）を決める必要がある．ターンの最大数は，1 つの細工対象の特徴ベクトルに対して変更可能な特徴量の最大数と読み替えることが出来る．変更する特徴量の数が多ければ多いほど，より容易に回避が起こりやすくなることが予想される．その一方，変更する特徴量が多くなると動作可能性に影響を及ぼす恐れが生まれるというトレードオフの関係が存在する．ラウンド数は学習が完了するまでに行うターンの数を指し，ラウンド数が多ければ多いほど沢山の検体を用いて学習することが出来るが，ラウンド数が増えるほど学習にかかる時間も多くなる．

評価実験ではラウンド数を 250,000，ターン数を 30 として回避攻撃の精度評価を行った．また，ACER の戦略モデルは乱数のシードによって結果が変化する．そこで今回は 10 回テストを行い，その平均を求めた．更に，回避結果が学習の結果によるものか確認するためにターンの最大数分だけランダムに特徴量を変更した場合の回避精度を求めた．

表 4　標的モデルの性能
Table 4　Performance of target model.

識別クラス	Precision	Recall	F1
良性	0.776	0.778	0.777
悪性	0.777	0.775	0.776

表 5　標的モデルの混同行列
Table 5　Confusion matrix of target model.

	識別結果 (良性)	識別結果 (悪性)
良性	778	222
悪性	225	775

4.5　実験結果

4.6　標的モデルの識別精度

線形 SVM を用いた識別器の性能を**表 4** に，混同行列を**表 5** に示す．表 5 より，回避攻撃の精度評価に用いることが出来るマルウェアの検体数は 775 個である．

4.7　回避攻撃の精度

先行研究と同じ攻撃者の前提での評価を行うために，標的モデルの識別精度評価に用いたマルウェアのうち正しく分類出来た 775 個の検体を用いて，攻撃者が 800 次元の特徴量を知っていた場合の攻撃器の回避攻撃精度を評価した．攻撃器の 250000 ラウンド目時点での精度と，800 次元の特徴量から選んだ 30 個の変数をランダムに変更した

表 6　正確な特徴量を用いたとき回避攻撃精度
Table 6　Accuracy of evasion attack with exact features.

回避攻撃が成功した割合	ランダムに特徴量を変更
0.895	0.133

表 7　提案手法の回避攻撃精度
Table 7　Accuracy of evasion attack of proposed method.

	(1)	(2)	(3)	(4)
1200 次元	0.871	0.787	0.839	0.858
1600 次元	0.794	0.787	0.793	0.787

表 8　ランダムに特徴量を変更した場合の回避精度
Table 8　Evasion rate by changing randomly.

	(1)	(2)	(3)	(4)
1200 次元	0.104	0.099	0.106	0.098
1600 次元	0.088	0.085	0.085	0.079

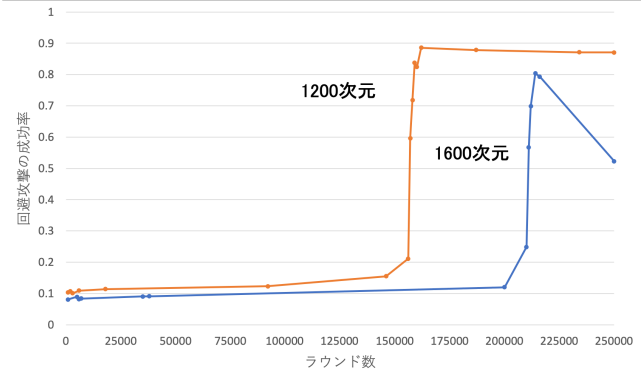

図 4　特徴量選択 (1) の学習経過
Fig. 4　Learning process of feature selection (1).

場合の回避攻撃精度を**表 6** に示す．

また，4 つの異なる特徴量選択で選んだ 1200 次元と 1600 次元の特徴量候補を用いた提案手法の回避攻撃精度を評価した．各攻撃器の 250000 ラウンド目時点での精度を**表 7** に示す．また，1200 次元と 1600 次元の特徴量候補から選んだ 30 個の変数をランダムに変更した場合の回避攻撃精度を**表 8** に示す．

提案手法の学習時の評価で最高スコアを更新した際のエージェントを記録することで学習経過を観測した．ただし，25000 ラウンド目は最高スコアでなくても必ず記録することとした．特徴量選択 (1) の学習経過を**図 4** に，特徴量選択 (2) の学習経過を**図 5** に，特徴量選択 (3) の学習経過を**図 6** に，特徴量選択 (4) の学習経過を**図 7** に示す．

5. 考察

5.1 提案手法の回避攻撃精度

表 7 より，提案手法は 4 つの特徴量選択のどれについても 78%以上のマルウェアを良性クラスに回避することに成功した．ランダムに特徴量を変更した場合の表 8 の結果

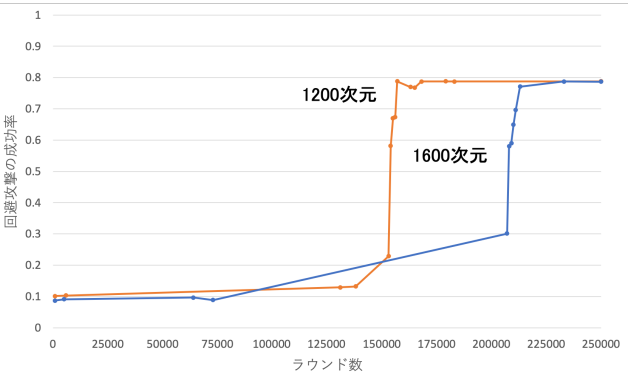

図 5　特徴量選択 (2) の学習経過
Fig. 5　Learning process of feature selection (2).

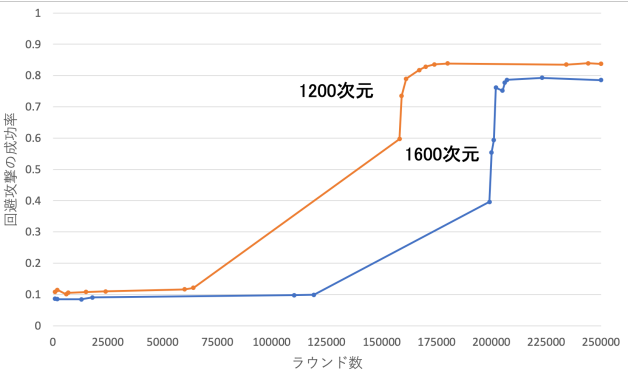

図 6　特徴量選択 (3) の学習経過
Fig. 6　Learning process of feature selection (3).

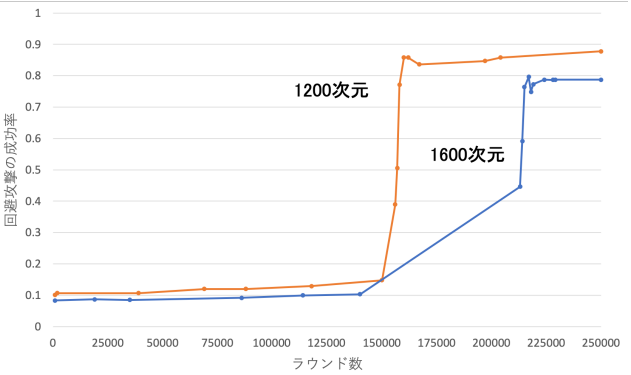

図 7　Learning process of feature selection (4)
Fig. 7　Overview of experiment.

と比較すると提案手法の精度は学習によって得られたものであると言える．また，正確な特徴量を用いたときの回避攻撃精度と比較しても 0.02 から 0.12 の差しかないことから，本研究の目的である前提知識として識別器が用いている正確な特徴量を必要としない回避攻撃手法を実現したと言える．

5.2 特徴量候補の次元数

特徴量候補が 1200 次元の場合と 1600 次元の場合の回避攻撃精度を比較すると，総じて 1600 次元の場合のほうが

精度が悪い傾向がわかる．また，図 4 から図 7 の学習経過を見ると 1200 次元の方が 1600 次元の場合よりも早いラウンド数で十分な回避攻撃精度を得ていることがわかる．これは探索範囲の差によるものであることが考えられ，次元数を更に増やすとより多いラウンド数の学習を行わないと十分な回避攻撃精度を得られないことが予想できる．

5.3 制限

図 4 から図 7 の学習経過より，特徴量候補が 1200 次元の場合は 150,000 ラウンド以上，1600 次元の場合は 200,000 ラウンド以上の学習が必要であることがわかる．これは識別器に対してラウンド数と同数のリクエストが必要であることから，できるだけ少ない方が好ましい．この制限は攻撃対象が二値分類器であり，強化学習の報酬がスパースであることによるものだと考えられる．解決策としては報酬がスパースでも学習が可能なアルゴリズムを適用することが考えられる．

6. おわりに

本研究では，先行研究が暗黙的に前提としていた，標的モデルが用いている正確な特徴量の情報を必要とする制約に着目し，その解決のために強化学習を用いたブラックボックス型の回避攻撃手法を提案した．評価実験では，正確な特徴量の情報を必要とせずに約 80%のマルウェアを良性に回避させることに成功した．今後の課題としては，回避に成功したマルウェアの特徴量から実行可能なマルウェアを再構築することが挙げられる．

参考文献

[1] Anderson, H. S., Kharkar, A., Filar, B., and Roth, P.: *Evading machine learning malware detection*, Black Hat USA briefings 2017.

[2] Goodfellow, I. J., Shlens, J., and Szegedy, C.: *Adversarial examples for malware detection*, International Conference on Learning Representations, (2017).

[3] Grosse, K., Papernot, N., Manoharan, P., Backes, M., and McDaniel, P.: *Adversarial examples for malware detection*, European Symposium on Research in Computer Security, Springer, pp. 62-79, (2017).

[4] Liu, X., Du, X., Zhang, X., Zhu, Q., Wang, H., and Guizani, M.: *Adversarial Samples on Android Malware Detection Systems for IoT Systems*, Sensors, 19(4), 974, (2019).

[5] A.C. Muller and S. Guido.: *Python* ではじめる機械学習 , O'REILLY Japan, (2017).

[6] Papernot, N., McDaniel, P., and Goodfellow, I.: *Transferability in machine learning: from phenomena to black-box attacks using adversarial samples*, arXiv preprint, arXiv:1605.07277, (2016).

[7] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... and Vanderplas, J.: *Scikit-learn: Machine learning in Python*, Journal of machine learning research, 12.Oct (2011): 2825-2830.

[8] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R.: *Intriguing properties of neural networks*, arXiv preprint, arXiv:1312.6199, (2013).

[9] Wang, Y., Liu, J., and Chang, X.: *Assessing transferability of adversarial examples against malware detection classifiers*, Proceedings of the 16th ACM International Conference on Computing Frontiers, ACM, pp. 211-214, (2019).

[10] Wang, Z., Bapst, V., Heess, N., Mnih, V., Munos, R., Kavukcuoglu, K., and de Freitas, N.: *Sample efficient actor-critic with experience replay. arXiv preprint*, arXiv preprint, arXiv:1611.01224, (2016).

[11] VirusTotal - Free Online Virus, Malware and URL Scanner, https://www.virustotal.com/