IoTマルウェア検知精度向上のためのパラメータに応じたOne-Class SVMの性能調査

川越　彪雅[1]　高崎　球宇我[1]　臼崎　翔太郎[1]　油田　健太郎[1]　山場　久昭[1]　朴　美娘[2]
岡崎　直宣[1]

概要：近年，我々の日常生活の中で，「モノ」と「インターネット」がつながるInternet of Things(IoT)が様々な分野で活用されている．一方，IoTを利用することで，インターネットを通して外部から攻撃される危険性が生じる．このような現状から，IoTセキュリティの重要性はより高まっていると考え，本研究ではマルウェアに感染したデバイスの有無を検出するシステムを提案する．本研究ではOne-Class SVMのパラメータ探索を行い，パラメータが異なる複数のOne-Class SVMを組み合わせた検知手法を検証し，検知精度向上を目指した．評価実験では，様々なパラメータのOne-Class SVMの性能を調査，選定し，パラメータが異なる9つの検知モデルを用いて検知精度の検証を行ったが，検知精度は改善できなかった．

キーワード：Iot, One-Class SVM, n-gram, アンサンブル法

Performance Analysis of One-Class SVM Based on Parameters for Improving IoT Malware Detection Accuracy

Hyoga Kawagoe[1]　Kuga Takasaki[1]　Shotaro Usuzaki[1]　Kentaro Aburada[1]　Hisaaki Yamaba[1]
Mirang Park[2]　Naonobu Okazaki[1]

Abstract: In recent years, the Internet of Things (IoT), which connects “things” and the “Internet”, has been utilized in various fields in our daily lives. However, the use of IoT also introduces the risk of external attacks via the Internet. Given this situation, the importance of IoT security is increasing. In this study, we propose a system to detect the presence of malware-infected devices. This research explores the parameter tuning of One-Class SVM and examines a detection method that combines multiple One-Class SVM models with different parameters to improve detection accuracy. In the evaluation experiments, we investigated and selected the performance of various One-Class SVM models and validated detection accuracy using 9 different detection models with varying parameters. However, the detection accuracy did not improve.

Keywords: IoT, One-Class SVM, n-gram, ensemble

1. はじめに

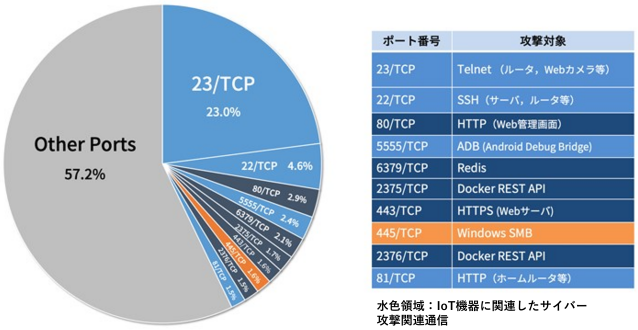
近年，我々の日常生活の中で，「モノ」と「インターネット」がつながるInternet of Things(IoT)が農業や医療，製造業など様々な分野で活用されている．IoTを活用することで，さまざまな「モノ」の操作や状態の把握，「モノ」同士の通信を遠隔で行うことができるようになる．京都市の交通局では，IoTシステムを用いて，バスに発信機を搭載し，運行データをリアルタイムで発信する仕組みを整えて，自分の乗りたいバスの現在地を知ることができるようにしてバスの待ち時間の不満を解消する試みが行われている[1]．このように，IoTは我々の生活に様々な利便性をもたらしてくれている．

[1] 宮崎大学
University of Miyazaki
[2] 神奈川工科大学
Kanagawa Institute of Technology

一方，IoT を利用することで，インターネットを通して外部から攻撃される危険性が生じる．IoT の脆弱性を突いて，悪意ある攻撃者が IoT 機器をサイバー犯罪へ利用した事例として 2016 年 10 月に米国の DNS サービスプロバイダである Dyn が大規模な DDoS 攻撃を受けた事例がある．この DDoS 攻撃では，GitHub，Netflix，Spotify，Twitter(現：X) などをはじめとする多くの大手ウェブサイトがダウンした．この攻撃は，「Mirai」と呼ばれるマルウェアに感染した，およそ 50 万台にも及ぶ IoT 機器によって構築されたボットネットによるものであった [2]．この「Mirai」による攻撃は，多くのユーザが IoT 機器のデフォルトユーザ名とパスワードを変更していないことが悪用された．同年 10 月には，「Mirai」の製作者として有力視されている人物がソースコードを公開したため，「Mirai」の亜種である IoT マルウェアは爆発的に増大した [3]．「Mirai」の亜種による攻撃は近年も観測されており，2021 年 1 月に Android OS を搭載した IoT 機器を狙う「Matryosh」，同年 2 月にハニーポット機能を感染拡大に利用する「ZHtrap」の攻撃が観測された [4]．

また，情報通信機構 (NICT) が公表した「NICTER 観測レポート 2022」[5] では，NICT の構築した大規模サイバー観測網にて観測したサイバー攻撃関連通信の宛先ポート番号別パケット数分布のうち，約 34%が IoT 機器に関連したものであることが分かった．宛先ポート番号別のサイバー攻撃観測数トップ 10 の内訳を図 1 に示す．

トップ 10 のポート番号のうち 4 つ (IoT 機器の Web インターフェイスが動作する 80/TCP を含めると 5 つ) が IoT 機器で利用されているポート番号である．また，グラフの約 6 割を占める Other Ports の中には IoT 機器で使用されるポートも多数含まれていることが分かっているため，それらも含めると IoT 機器に関連したサイバー攻撃通信は非常に多いことが分かる．以上より，IoT 機器を標的としたサイバー攻撃通信は非常に多いことから，IoT のセキュリティの重要性はより高まっているといえる．IoT 機器のマルウェア感染を検出するシステムの研究に，中川の研究 [7] がある．同文献では，マルウェアの感染の判別にアンサンブル法が導入されたが，検知精度向上は達成されなかった．この理由として，機械学習におけるパラメータ探索が行われていなかったために，低性能なモデルとなってしまっていた可能性が考えられた．そこで，本研究では，様々なパラメータの One-Class Support Vector Machine(以下 OCSVM) の性能を調査し，指針に基づいて選択したパラメータの OCSVM を用いて，アンサンブル法を導入し，その有効性を検証する．



ポート番号	攻撃対象
23/TCP	Telnet（ルータ，Webカメラ等）
22/TCP	SSH（サーバ，ルータ等）
80/TCP	HTTP（Web管理画面）
5555/TCP	ADB (Android Debug Bridge)
6379/TCP	Redis
2375/TCP	Docker REST API
443/TCP	HTTPS (Webサーバ)
445/TCP	Windows SMB
2376/TCP	Docker REST API
81/TCP	HTTP（ホームルータ等）

水色領域：IoT機器に関連したサイバー攻撃関連通信

図 1: NICT が観測した宛先ポート番号別サイバー攻撃通信の内訳（文献 [6] より引用）

2. 関連研究

文献 [8] では，PAYL というアノマリ型マルウェア検知システムが提案された．PAYL はパケットからの特徴量の抽出に n-gram 解析を用いる．学習フェーズでは，各ホストに対するパケットのポート，ペイロード長ごとに n-gram の出現回数の平均と分散を算出し，それらを用いてモデルを構築する．テストフェーズでも同様に，検知対象のパケットのペイロードにおける n-gram の出現回数の平均と分散を算出し，そのパケットのポート，ペイロード長に該当するモデルとのマハラノビス距離を測定する．マハラビノス距離が閾値以上であれば，そのパケットを異常として検出する．

また，文献 [9] ではマルウェアに感染した機器が C&C サーバとの通信を行う際のパケットのペイロード情報が，正常通信のペイロード情報と異なる特徴を持つことが述べられている．異なる特徴の一つとしてペイロード内の ASCII 文字コードの出現頻度が挙げられており，n-gram 解析によりペイロードから特徴量を抽出することはマルウェア検知に有効であると結論付けている．

文献 [10] では，n-gram を改良した 2ν-gram 法により特徴量の抽出を行い，教師なし学習アルゴリズムである OCSVM を用いて学習・検知を行うシステムを提案している．同文献では，ペイロードの解析と OCSVM を組み合わせることで良好な検知結果が得られていた．このため，本研究における学習アルゴリズムに OCSVM を採用した．また，複数の分類器を組み合わせて使用することで検知精度が向上するという結果が示されたことから，システムの検知精度の向上を図るため，中川の研究 [7] では，文献 [8] [9] [10] を元に，n-gram 解析と OCSVM を用いて IoT デバイスの正規通信とマルウェアに感染しボット化した後のデバイスの通信とを区別するシステムが提案され，システムの検知精度の向上を図るためにアンサンブル法が導入された．ただし，検知精度の向上は達成されず，その理由として、OCSVM におけるパラメータ探索が行われていなかったために，低性能なモデルとなってしまっていた可能性が考えられた．

3. 提案手法

本研究では，中川の手法を元に OCSVM とアンサンブル

法を用いた攻撃検知を行うが，次のような工夫を行って検知精度の向上を目指す．

中川の研究では，検知精度を向上させる方策として，OCSVM の適切なパラメータの探索（適切な nu と gamma の選択）が挙げられていた．しかし，パラメータの値の組を様々に変更して性能評価を行ったところ，見逃し（異常パケットを誤って正常パケットと判定しまうこと）の低下と，誤検知（正常パケットを誤って異常パケット判定してしまうこと）の低下を両立することが困難であり，前者で性能が良いパラメータの組では後者の性能が悪い，後者で性能が良い組では前者の性能が悪いという結果であった．そこで，見逃しの少ないパラメータの組，誤検知の少ない組，双方の性能の調和がとれた組をそれぞれを 3 組ずつ，合計 9 つの OCSVM を用意し，その多数決により判定を行うというアンサンブル法を利用した手法を提案し，その評価を行う．

提案手法により，双方の性能の調和がとれた OCSVM における見逃しと誤検知の数が，それぞれ見逃しの少ない OCSVM と誤検知の少ない OCSVM に近づき、検知精度が改善されることを期待している．

3.1 本手法が実装される環境

IoT 機器は，一般的に低コストかつ省電力であり，機器には最小限のハードウェアリソースしか搭載されていないことが多い．そのため，セキュリティ対策のために機器個別にシステムを実装するのではなく，ゲートウェイなどの中継機器や，クラウド上に実装することが現実的であると考える．図 2 に本研究で想定している IoT ネットワークを示す．

3.2 提案手法

3.2.1 アノマリ型検知手法

本研究では，数が増大している「Mirai」の亜種である IoT マルウェアに対応するため，アノマリ型の検知手法を採用する．

マルウェアの検知手法は大きく分けてシグネチャ型とアノマリ型の 2 つに分類される．シグネチャ型は過去にマルウェアに感染した機器の通信パターンやバイナリをシグネチャとして記録し，このシグネチャと一致するものを異常

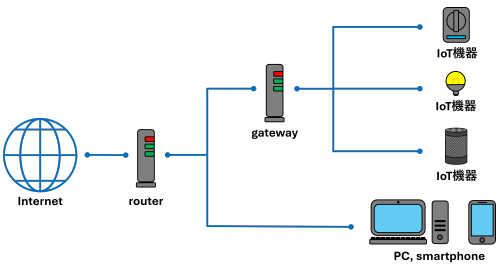


図 2: 本研究で想定している IoT ネットワーク

として検知する手法である．シグネチャ型検知手法はシグネチャとして定義してある既知のマルウェアを確実に検知することができるが，未知のマルウェアについては検知することができない．一方で，アノマリ型は正常時の通信パターンを正常として定義し，その定義から外れる異常なパターンを検知する．アノマリ型検知手法は正常パターンの定義を適切に行うことができれば未知のマルウェアであっても検知できる．また学習に正常時の通信パターンのみを用いるため，マルウェアに感染した機器の通信パターンを収集する必要が無く，実装が容易である．

3.2.2 n-gram 解析による特徴量抽出

本研究では，機械学習に用いる特徴量の抽出を n-gram 解析によって行う．n-gram 解析は，テキストを任意の文字数 n の単語で分割する手法である．分割した文字列の出現頻度を求めることで，テキスト中の任意の文字列の出現頻度パターンを得られる．以下に n-gram 解析による文字列の分割例を示す．

文字列: 猿も木から落ちる

$N = 1$**(1-gram):** 「猿」「も」「木」「か」「ら」「落」「ち」「る」

$N = 2$**(2-gram):** 「猿も」「も木」「木か」「から」「ら落」「落ち」「ちる」

$N = 3$**(3-gram):** 「猿も木」「も木か」「木から」「から落」「ら落ち」「落ちる」

文献 [9] にて，n-gram 解析によって通信パケットから特徴量を抽出することの有効性が示されていることから，本研究では n-gram 解析を採用した．具体的には，パケットの n 個の連続したバイト列の出現頻度から出現回数の総和・平均・標準偏差を計算し，この 3 つを機械学習における入力データとする．n-gram 解析の特性上，n の値を大きくするほど計算量の増加と精度の低下が見られるため，本研究では 2-gram 法により特徴量を抽出する．

3.2.3 One-Class Support Vector Machine

異常値の検出を行うために本研究では OCSVM を採用する．これは文献 [10] で，テキスト分類において n-gram 解析と組み合わせると良好な精度を示すことが報告されていたためである．

OCSVM は教師無し学習による 1 クラス分類手法であり，正常値（+1）のクラスのベクトルのみを学習し，与えられたベクトルがそのクラスに属するか否かで判定を行う．通常，正常値はデータ空間において密度の高い領域にあり，異常値は密度が低い領域にある．そこで，データ間の距離を (3.1) 式のガウシアンカーネルを用いて特徴空間へ写像することで，異常値を原点近くに写像することが可能となる [11]．

この性質を利用して，原点付近のデータ群とその他の

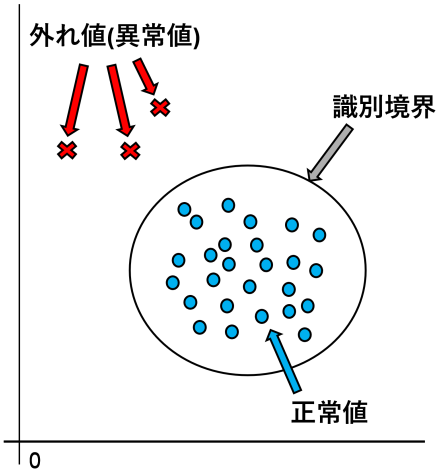


図 3: OCSVM におけるクラス分類のイメージ

データ群を識別するマージンが最大となるような超平面を設定し，外れ値の外れ具合の指標を求める．(3.2) 式は識別関数 [12] であり，Φ で写像された非線形空間での内積は (3.1) 式となる．学習データのうち m の割合のデータが原点付近にくるような識別境界を (3.3) 式の最適化問題によって求める [13].

$$K(x, x') = \exp\left(-\frac{||x - x'||^2}{\sigma^2}\right) \tag{1}$$

$$f(\Phi(x)) = \mathrm{sgn}(\omega \cdot \Phi(x) - \rho) \tag{2}$$

$$\min_{\omega,\xi,\rho}\left(\frac{1}{2}||\omega||^2 + \frac{1}{ml}\sum_i \xi_i - \rho\right) \tag{3}$$

subject to

$$(w \cdot \phi(x_i)) \geq \rho - \xi_i, \xi_i \geq 0$$

(3.2) 式の識別関数が正であれば正常，負であれば異常と判別する．

通常の SVM では複数クラスのデータを学習データとして用いることから，分類器としての役割を強く持つのに対して，正常データのみ学習して識別境界を生成する OCSVM は，正常・異常データを識別する外れ値検知手法としての役割を強く持つ．

本研究では，正常通信時のパケットの特徴量ベクトルを正常値として学習しておき，このクラスから外れたパケットをマルウェアに感染したデバイスの通信パケットと判定する．OCSVM において識別境界を決定した後の分類結果のイメージを図 3 に示す．

また，OCSVM を運用する上で学習の際に設定すべきパラメータに nu と $gamma$ がある．nu は識別境界の決定時に考慮しないデータの割合の上限を示すパラメータである．nu の値を高めると識別境界の決定時に考慮しないデータの割合が増加し，一般的に識別境界が内包する領域は狭まる．$gamma$ は識別境界の複雑さを決定するパラメータである．$gamma$ の値を高めると，識別境界がより複雑になる．

本研究では，nu と $gamma$ の 2 つのパラメータを様々に変えた時の OCSVM の性能を評価し，その結果に基づいてアンサンブル法で使用する OCSVM の選定を行う．

3.2.4 アンサンブル法

アンサンブル法とは，複数の予測モデルで物事を予測し，それらのモデルの予測結果を組み合わせて最終的な予測結果を出す方法である．複数の予測モデルの予測結果をまとめて出力するので、個々の単独な予測モデルよりも一般的に性能が高い [14]．もっともよく知られているアンサンブル法として，多数決の原理を利用するものがある．これは，分類機の過半数によって選択されているクラスを選択する方法である．

3.3 提案システムの流れ

本研究の提案システムの流れは，3.3.1 特徴ベクトル抽出，3.3.2 学習フェーズ，3.3.3 テストフェーズの 3 つからなる．

3.3.1 特徴ベクトル抽出

データの前処理として，n-gram 解析を用いて特徴量の抽出を行う．はじめに，パケット内のバイト列を文字列とみなし，2-gram を生成する．そして，2-gram の出現頻度をカウントし，出現回数の総和・平均・標準偏差の 3 つを 1 パケットの特徴ベクトルとして抽出する．パケットから 2-gram を生成する例を図 4 に示す．

3.3.2 学習フェーズ

学習フェーズでは，正常データの特徴ベクトルのみを用いて OCSVM の学習を行い，パケットが正常か異常かを識別する予測モデルを生成する．つまり，このフェーズで正常値と異常値を区別するための識別境界が構築され，正常クラスの領域が設定される．本研究の提案システムでは，複数のパラメータの異なる OCSVM を検知器として用いるが，同一の学習データを用いてそれぞれ予測モデルを構築する．学習フェーズの処理の流れを図 5 に示す．

3.3.3 テストフェーズ

テストフェーズでは，正常・異常データの両方を用いて学習フェーズで生成した予測モデルから対象パケットの正常・異常を判別する．判別は，対象パケットの特徴ベクト

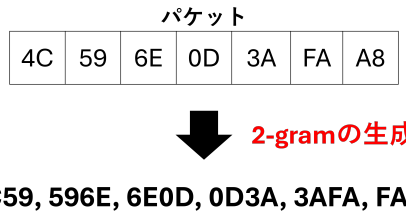


図 4: パケットから 2-gram を生成する例

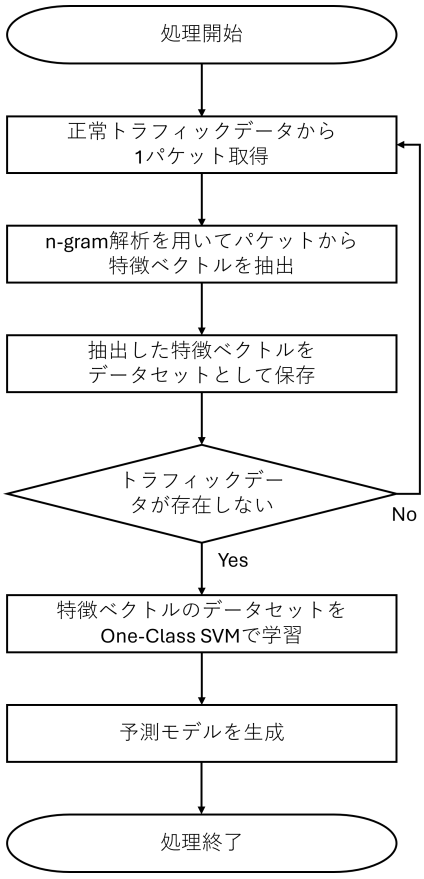


図 5: 学習フェーズの処理の流れ

ルが正常クラスの領域内に分類されたときは正常と判別し，正常クラスの領域外に分類されたときは異常と判別する．本研究の提案システムでは，まず，構築した複数の予測モデルそれぞれでテストを行う．そして，パケットに対するそれぞれの予測モデルの判別結果で多数決を行い，その結果を最終的な判別結果としている．図 6 に，テストフェーズの処理の流れを示す．

本研究ではパラメータの異なる複数の OCSVM を組み合わせて多数決を行うが，TPR と FPR の調和がとれた OCSVM と TPR が高い OCSVM，FPR が低い OCSVM を組み合わせることで，単体の OCSVM と比べて，検知精度が向上するのではないかと考えている．

4. 評価実験

4.1 実験目的

提案システムの検知精度向上をめざし，まず，パラメータに応じた OCSVM の性能調査を行い，適切なパラメータの組み合わせを発見する．ついで適切なパラメータの OCSVM においてアンサンブル法を導入する．単一の OCSVM の検知精度と，提案手法の検知精度の比較を行い，有効性を検証する．

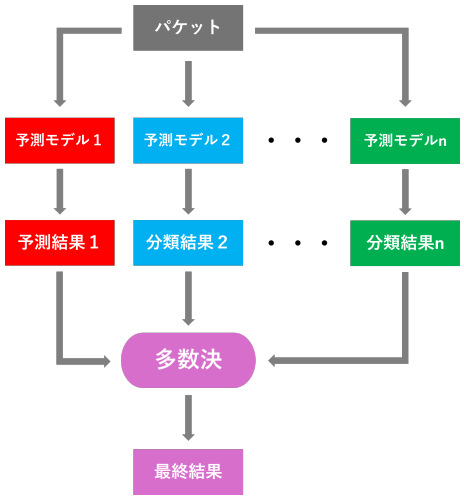


図 6: テストフェーズの処理の流れ

4.2 データセット

今回の実験では，IoT 機器の通信トラフィックのデータセットとして Stratosphere Lab に公開されている IoT-23 [15] を使用した．このデータセットのうち，良性 3 種 (表 1)，悪性 1 種 (表 2) のシナリオを用いる．良性シナリオはモデルの学習用とテスト用に分割する必要があるため，今回は良性シナリオの 8 割を学習用データとして分割した．そして，検知精度の測定の際には，残りの 2 割の良性シナリオと全悪性シナリオをテストデータとしてモデルに予測させる．また，データセットの分割において偶然性を排除するため，3 つの異なる疑似乱数シード値を用いてデータセットを分割し，実験を行うこととする．つまり，同じ手法を 3 回独立して適用し，それぞれの実験において異なるデータセットの分割を行う．

4.3 評価方法

本実験では正常を Negative，異常を Positive とし，以下の指標を用いて評価実験を行う．ACC は全データ内から正しく検知できた割合，TPR は異常データ全体から正しく異常と検知できた割合，FPR は正常データ全体から誤って異常と検知した割合を示す．

表 1: 良性シナリオ詳細

データセット名	キャプチャ期間	パケット数	IoTデバイス名
CTU-Honeypot-Capture-7-1	1.4 hrs	8,276	Somfy Door Lock
CTU-Honeypot-Capture-4-1	24 hrs	21,000	Philips HUE
CTU-Honeypot-Capture-5-1	5.4 hrs	398,000	Amazon Echo

表 2: 悪性シナリオ詳細

データセット名	キャプチャ期間	C&Cサーバとの通信パケット数	マルウェア名
CTU-Malware-Capture-34-1	24 hrs	36,797	Mirai

$$Accuracy(ACC) = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

$$True\,Positive\,Rate(TPR) = \frac{TP}{TP + FN} \quad (5)$$

$$False\,Positive\,Rate(FPR) = \frac{FP}{FP + TN} \quad (6)$$

4.4 パラメータに応じた OCSVM の性能調査

4.4.1 パラメータの候補

OCSVM の判別性能が，2 つのパラメータ nu と $gamma$ の値の組に応じて，どのように変化するのか，値を様々に変えて調査した．

まず，中川の研究で用いられていた値を参考に，いくつかの候補値を選択し，その値での実験を行う．さらにその結果から，性能の良かった値の周辺をより細かく分割するように候補の値を選んで同様の実験を行う．具体的には，まず nu と $gamma$ それぞれについて，0.001, 0.01, 0.05, 0.1, 0.25, 0.5 の 6 つのいずれかとした 36 組で実験を行い，その結果として，OCSVM の性能に対して nu の影響が大きいこと，ただし nu の値が 0.001 と小さいときは性能が低いことから，nu の候補値として 0.025, 0.075, 0.15, 0.2, 0.3, 0.4 を追加した．この 6 つと $gamma$ の 6 つの組み合わせの 36 通りについて，同じ実験を行った．

4.4.2 実験結果

性能調査の結果，nu の値が検知精度に与える影響が顕著であることが確認され，$gamma$ に関しては，nu の値を固定したとき，0.001 の場合のみ検知精度が低く，他の値ではほぼ同等の検知精度が得られた．

- nu = 0.05, 0.075, 0.1

  2 つの異なる傾向が確認された．

  - TPR が 95〜98%で，FPR が 15〜20%
  - TPR が 75〜78%で，FPR が 7〜12%

  ACC は 84〜89%であり，他のパラメータ群と比較して最も高く，TPR と FPR に極端な偏りがない．

- nu = 0.15, 0.2, 0.25, 0.3

  2 つの異なる傾向が確認された．

  - TPR が 98%以上で，FPR が 22〜35%
  - TPR が 77〜79%で，FPR が 14〜27%

  nu = 0.05, 0.075, 0.1 の場合と似た傾向があり，比較すると TPR は高いが，ACC は 74〜83%と低く，FPR も高い．

総括すると，nu = 0.05, 0.075, 0.1 のとき，他のパラメータの組と比較して ACC が最も高く，TPR と FPR の調和がとれた予測モデルが生成される．nu が大きくなると（nu = 0.15, 0.2, 0.25, 0.3），TPR は高くなるが、FPR も高くなり、ACC は低下する．

表 3: 検知精度の比較 (1 回目)

	TPR	FPR	ACC
単一のOCSVM	98.35%	18.25%	86.74%
アンサンブル	98.53%	20.81%	85.00%

表 4: 検知精度の比較 (2 回目)

	TPR	FPR	ACC
単一のOCSVM	96.72%	17.31%	86.91%
アンサンブル	98.53%	20.82%	85.00%

表 5: 検知精度の比較 (3 回目)

	TPR	FPR	ACC
単一のOCSVM	98.74%	17.82%	87.16%
アンサンブル	98.67%	18.17%	86.89%

4.5 提案手法の性能評価実験

4.5.1 実験条件

本研究では, (A) 見逃しと誤検知の調和がとれた OCSVM と，(B) 見逃しが少ない OCSVM，(C) 誤検知が少ない OCSVM を組み合わせることで，単一の OCSVM からの検知精度向上を図る．

したがって 4.4 パラメータに応じた OCSVM の性能調査で得られた結果のうち，以下に示す結果を得た nu と $gamma$ の OCSVM を本実験で使用する．

(A) TPR が 95%以上かつ FPR が 20%以下の中で ACC の上位 3 値

(B) FPR が 10%以下の中で ACC の上位 3 値

(C) TPR が 99%以上の中で ACC の上位 3 値

本実験では，上記の条件を満たす 9 つの OCSVM を組み合わせ，検知精度を検証する．

4.5.2 実験結果

単一の OCSVM の検知精度として (A)TPR が 95%以上かつ FPR が 20%以下の中で ACC の上位 3 値の OCSVM の検知精度の平均を示し，提案手法の検知精度と比較する．1 回目の結果を表 3，2 回目の結果を表 4，3 回目の結果を表 5 に示す．

表 3，表 4，表 5 より，3 回行った実験ののうち 2 回の実験では TPR のみ僅かに向上したが，他の評価指標は向上しなかった．したがって，提案手法によって検知精度は向上しなかったといえる．また，アンサンブル法の導入前と導入後で，いずれの評価指標においても差は 4%を下回り，検知精度は変化しなかったといえる．

4.6 考察

パラメータに応じた OCSVM の性能調査において，nu が 0.05〜0.3 の範囲で異なる 2 つの傾向を持つ予測モデル

が生成された．この現象は，特徴空間において正常パケットと異常パケットが混在している領域が存在し，その領域が正常クラスに含まれるか否かによって生じていると推測される．

さらに，(A) 見逃しと誤検知の調和がとれた OCSVM, (B) 見逃しが少ない OCSVM, (C) 誤検知が少ない OCSVM を組み合わせてアンサンブル学習を行ったが，これらの予測モデルのいずれも ACC が 70%以上であるにもかかわらず，検知精度には変化が見られなかった．この結果から，n-gram 解析と OCSVM によって生成された予測モデルは，同じパケットに対して同じ予測をする傾向が強いと推測される．

以上より，提案された n-gram 解析と OCSVM を用いた検知システムのみでは，検知精度の向上が困難だと考えられる．

検知精度を向上させるためには、混在領域に存在するパケットを正しく分類する必要があり，そのための改善案として，OCSVM とは異なる異常検知アルゴリズムを組み合わせたアンサンブル学習や，よりパケットの特徴を反映させた特徴量の導入などが考えられる．

また，単体の検知モデルにおいて TPR は 98%と高い精度が得られており，15〜20%の FPR の削減が具体的な課題として挙げられる．文献 [16] では，機械学習とホワイトリストを併用することで，機械学習のみの場合と比べて FPR が低下し，検知精度が向上した．提案システムにおいてもホワイトリストを導入することで検知精度の向上が期待される．

5. まとめ

本研究では，IoT マルウェアの検知において，n-gram 解析と OCSVM を組み合わせたシステムを提案し，パラメータに応じた OCSVM の性能調査を行った．また，異なるパラメータを持つ複数の OCSVM を使用して多数決型のマルウェア判定方式を導入し，その有効性を検証した．

パラメータに応じた OCSVM の性能調査を行った，*nu* は 0.05〜0.1，*gamma* は 0.01〜0.5 の範囲で最も優れた検知精度を示すことが確認された．

多数決型の判定方式では，9 つの予測モデルを組み合わせて検知精度を検証したが，提案手法においては検知精度の向上は達成できなかった．今後の改善案として，複数の異常検知アルゴリズムを組み合わせたアンサンブル学習や新しい特徴量の導入，機械学習とホワイトリストの併用を考えている．

謝辞 本研究は JSPS 科研費 JP24K14917 の助成を受けたものです．

参考文献

[1] 京都市交通局さま｜ディスプレイ関連商品｜導入事例｜法人のお客様: シャープ，https://jp.sharp/business/case/display/display_detail_170.html, (最終閲覧日:2024 年 1 月 17 日)
[2] The DDoS Attack on Dyn’s DNS Infrastructure, https://www.thousandeyes.com/blog/dyn-dns-ddos-attack，2016 年 10 月 25 日更新 (最終閲覧日:2024 年 1 月 17 日)
[3] Q.D.Ngo, H.T.Nguyen, V.H.Le, D.H.Nguyen：A survey of IoT malware and detection methods based on static features, *ICTExpress*, Volume6, Issue4, pp.280-286, 2020.
[4] 独立行政法人情報処理推進機構 (IPA)，情報セキュリティ白書 2022，pp.173-175，2022.
[5] 国立研究開発法人情報通信研究機構 サイバーセキュリティ研究所 サイバーセキュリティネクサス，NICTER 観測レポート 2022，pp.2-4，2022.
[6] NICTER 観測レポート 2022 の公開，https://www.nict.go.jp/press/2023/02/14-1.html, 2023 年 2 月 14 日更新 (最終閲覧日：2025 年 2 月 14 日)
[7] 中川拓：n-gram 解析と One-Class SVM を用いた IoT ボットネットワークの検知手法における検知精度向上の検討，2022 年度宮崎大学工学部卒業論文
[8] Wang, Ke, andS．J．Stolfo: Anomalous payload based network intrusion detection，RAID．Vol.4．2004.
[9] 大月優輔, 市野将嗣, 川元研治, 畑田充弘, 吉浦裕: マルウェア感染検知のためのトラヒックデータにおけるペイロード情報の特徴量評価, ComputerSecuritySymposium, 2012.
[10] R. Perdisci, G. Gu, W. Lee: Using an Ensemble of One-ClassSVM Classifiers to Harden Payload-based Anomaly Detection Systems, *Proceedings of the 6th IEEE International Conference on Data Mining(ICDM2006)*, pp.488-498，2006.
[11] 機械学習カーネル法について，https://enakai00.hatenablog.com/entry/2017/10/13/145337，2017 年 10 月 13 日更新 (最終閲覧日:2024 年 1 月 24 日)
[12] サポートベクターマシン，https://home.hiroshima-u.ac.jp/tkurita/lecture/svm/node4.html/，2002 年 7 月 18 日更新 (最終閲覧日:2024 年 1 月 24 日)
[13] One Class Support Vector Machine(One Class SVM) 入門，https://recruit.cct-inc.co.jp/tecblog/machine-learning/one-class-svm/，2019 年 5 月 7 日更新 (最終閲覧日:2024 年 1 月 24 日)
[14] Sebestian Raschka, Vahid Mirjalili, (2019), *Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow2*， Packt Publishing.
(福島真太朗 (監修)，株式会社クイープ (翻訳)，(2020), Python 機械学習プログラミング 達人データサイエンティストによる理論と実践，インプレス．)
[15] Aposemat IoT-23, https://www.stratosphereips.org/datasets-iot23，(最終閲覧日:2024 年 1 月 23 日)
[16] Masataka Nakahara, Norihiro Okui, Yasuaki Kobayashi, Yutaka Miyake: Malware Detection for IoT Devices using Automatically Generated White List and Isolation Forest, Proceedings of the 6th International Conference on Internet of Things, *Big Data and Security 2021*，pp.38-47，2021.