博士論文

サイバーセキュリティ領域における機械学習の適切な利用に関する研究

Research on the Appropriate Use of Machine Learning in Cyber Security Domain

国立大学法人 横浜国立大学

大学院環境情報学府

新井 悠

Yu ARAI

責任指導教員 松本 勉 教授

2024 年 3 月

概要

近年，機械学習の発展と社会への浸透はますます進むばかりである．たとえばスマートフォンの写真機能には被写体の識別機能が付属するようになったり，映り込んだ不要なものを消去して違和感のないように修正してくれるといった機能が追加されている．ほかにも録音した音声の翻訳・文字起こしや，電話中に周囲の騒音を相手方に伝わらないように低減する，といったさまざまな応用事例に枚挙にいとまがない．スマートフォン以外にもビデオカメラやドローン，自動車やビデオゲームなど，機械学習が利活用されることで効率や精度の向上に寄与する実世界の事例は確実に増加している．

このように機械学習が効率や正確性の向上のために他分野で取り入れられてきているのと同様に，サイバーセキュリティ領域においても同技術が使用されるようになってきている．しかしサイバーセキュリティの領域では機械学習の使用は従来からある対策技術を一部代替させるものが過半で，他の領域のような広がりをみせてはいない．そこで本研究ではまず，機械学習をサイバーセキュリティの領域でより積極的に活用することで，解決できる課題はないか検討を重ねた．そして検討の結果，社会的な問題であるダークウェブの犯罪関連サイトの自動検出に使用することで，犯罪などの社会問題の低減に貢献できる巡回システムを提案した．同システムを使用して効率的にダークウェブから犯罪に関連したフォーラム，いわゆる闇掲示板を自動的に特定することを確認した．

次に，機械学習をすでにサイバーセキュリティ領域で使用している製品やサービスのなかで隘路となっている事項はないか検討した．こうした隘路を明らかにする目的は，同課題を解消した，よりよい製品やサービスが社会に提供され，結果としてさらに安全な社会となることが期待できるためである．検討の結果，市販されているウイルス対策ソフトで機械学習による検出を主たる検知手法としている製品に対して，その検知を回避する方法を提案した．同手法により，機械学習による検出を単体で組み込んでいる商用製品に対して回避攻撃を実現できることを明らかにした．加えて，従来型のパターンファイル検出と機械学習による検出の 2 種類をハイブリッドで使用している商用製品に対しても回避攻撃を実現できることも明らかにした．本研究では，こうした機械学習の適切な利用についてのユースケースを示す．

Abstract

In recent years, machine learning has only continued to develop and permeate society. For example, smartphone photo functions now include a function to identify the subject of the photo, or to erase unwanted objects in the image and correct them so that they do not look out of place. Other features include translation and transcription of recorded voice, and reduction of ambient noise so that it is not transmitted to the other party during a phone call, to name just a few. In addition to smartphones, there is a steadily increasing number of real-world examples where machine learning is used to improve efficiency and accuracy, such as in video cameras, drones, automobiles, and video games.
Just as machine learning has been adopted in other fields to improve efficiency and accuracy, it is also being used in the cyber security domain. However, the use of machine learning in cybersecurity is not as widespread as in other fields, as it is mostly used to replace some of the existing countermeasure techniques.
Therefore, in this study, we first examined whether there are issues that can be solved by more actively using machine learning in the cyber security domain. As a result, we proposed a patrol system that can contribute to the reduction of social problems such as crime by automatically detecting crime-related sites on the Dark Web, which is a social problem. We confirmed that the system can efficiently and automatically identify crime-related forums, or so-called dark bulletin boards, on the Dark Web.
Next, we examined the bottlenecks in products and services that are already using machine learning in the cyber security domain. The purpose of identifying these bottlenecks is to provide society with better products and services that solve the same problems, which will result in a more secure society. As a result, we proposed a method for avoiding detection of commercial antivirus software that uses machine learning as its primary detection method. We found that the proposed method can achieve evasion attacks against commercial products that incorporate machine learning detection as a stand-alone feature. In addition, we also showed that our method can achieve evasion attacks against commercial products that use a hybrid of conventional pattern file detection and machine learning detection. We present a use case for the appropriate use of machine learning.

目次

第1章　序論

1.1 機械学習の進展と社会への浸透

機械学習の社会実装は進む一方であり，もはや同技術が使用されていることは特筆することではなくなりつつある．たとえば機械学習による画像認識の領域では，内視鏡画像からがん病変の検出を支援するソフトウェアが考案されている [1]．同ソフトウェアでは，内視鏡医による所見が付けられた 25 万件以上の内視鏡画像を学習データとすることで，98%の病原発見率を得ることができたという．ほかの分野でも，たとえば水道管の劣化状況の予測・診断を実施することで，老朽化した水道管の補修の優先順位付けを可能にする商用サービスが自治体向けに提供されている [2]．同サービスにおいては，配管素材，使用年数，過去の漏水履歴といった水道管に関連するデータに加えて，独自に収集した土壌，気候，人口など1000 以上のパラメータからなる学習データを使用して予測を実現しているという．

1.2 サイバーセキュリティ領域における機械学習

前記の情勢はサイバーセキュリティ領域においても同じ様相を呈しており，例えば NGAV（Next Generation Anti-Virus）や EDR（Endpoint Detection and Response），WAF（Web Application Firewall），SIEM（Security Information and Event Management）といった製品に機械学習を使用したサイバー攻撃検出の機能が組み込まれるようになってきている [3]．

1.3 本研究の目的

前記のように機械学習を利活用したソリューションの社会実装は進む一方であり，サイバーセキュリティの領域もその例外ではない．今後もこの風潮は続くものと思われ，生成 AI などによるアシスタント機構などもかなり近い未来に実装されることが確実視されている．そのような中で本研究はサイバーセキュリティの領域において，既存の対策だけではなく新しい領域に関しても機械学習の有効性を試すべく，ダークウェブにおける違法物品取引サイトの検知に関する実験を行った．

その上で，こうした機械学習の社会実装が進んだことが逆に引き起こす盲点や，課題について検討を行った．なぜならば，こうしたソリューションが進展する一方で，機械学習を使用しているがゆえに生じうる問題点について知悉しておかなければ，より大きな問題を引き起こすことになりかねない．そのことを端的に確認する事例として，商用の機械学習ベースのウイルス対策ソフトに対する回避攻撃を行った．

第2章 ダークウェブ内の違法物品取扱サイトの自動検出

2.1 ダークウェブとは

近年，違法薬物，児童ポルノ，あるいはサイバー攻撃ツールやサイバー攻撃代行サービスといった違法物品ならびにサービスが，いわゆるダークウェブ内に構築された Web サイトで取引されている．ダークウェブは Tor [4]，Freenet [5]，I2P [6]といった秘匿ネットワークを実現するソフトウェアによって構成されている．ダークウェブは基本的にこうしたソフトウェアを利用することでしかアクセスできなかったが，近年ではプロキシサーバなどを通じて，こうしたソフトを使用しなくてもアクセスすることができる [7]．本研究の対象である Tor によるダークウェブの模式図を図 1 に示す．Tor は，Tor ネットワークを構成する中継ノードを，アクセスする毎にランダムに選択することによって，アクセス元の IP アドレスを秘匿し，同時に，アクセス先のホストの IP アドレスも秘匿するような仕組みを持っている．

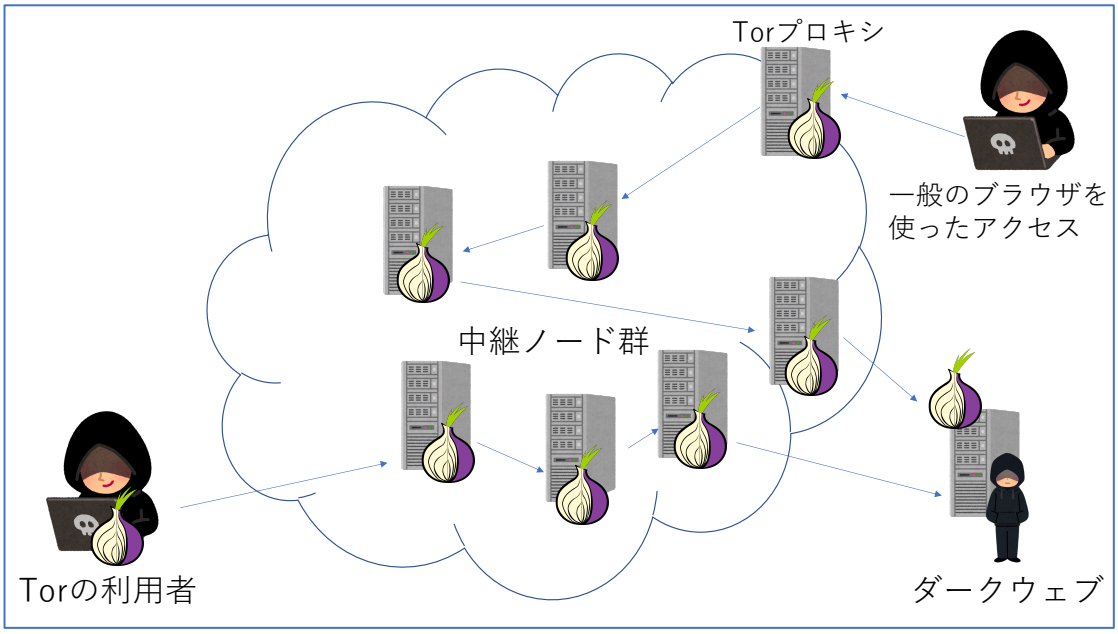

図 1: ダークウェブの模式図

こうしたダークウェブ内に設けられた取引所を利用することで，その利用者は違法薬物販売者との接点を持つことなく違法薬物を入手することが可能である．あるいは特別な情報処理の知識を持たずとも，販売されているサイバー攻撃を実行できるツールや，サイバー攻撃のアウトソーシングサービスを，仮想資産等を使用して購入することで，サイバー攻撃を実行することが可能となってきている．このため，海外では 2014 年に FBI や Europol が中心となり「Operation Onymous」を実施し，400 以上のダークウェブ内のサイトを停止させたと明らかにしている [8]．日本国内においてもこうした違法行為が問題と

なっており，京都府警が 2018 年の 6 月に児童ポルノサイトをダークウェブ内において運営していた被疑者を検挙しており [9]，2019 年 11 月にも同様の容疑で別の被疑者を検挙している [10]．加えて，2019 年 5 月には，経済産業省の職員がダークウェブ内の違法物品取扱サイトを使用することで，米ロサンゼルスから成田空港に国際郵便で到着した雑誌の袋とじの中に，覚せい剤を入れたものを同年 4 月に受け取っていた容疑で検挙されている [11]．

このようにダークウェブで売買されている物品の社会問題が浮上していくなかで，法執行機関による違法物品取扱サイトのテイクダウンが継続的になされているが，テイクダウンが行われると，また別の違法物品取扱サイトが誕生するという循環が生まれてしまっている．その一例として，「silk road」という悪名高い違法物品取扱サイトがある．図 2 に，同サイトの時系列的な変遷を示す．

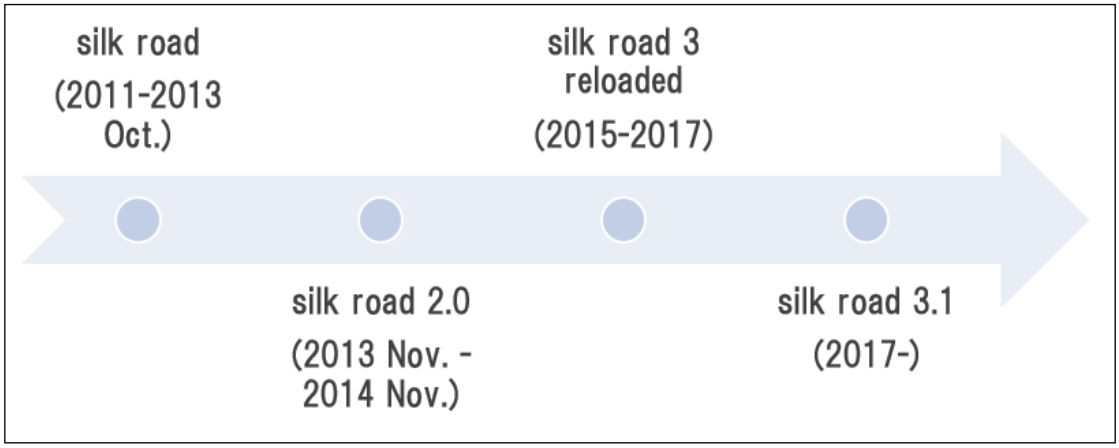

図 2: silk road の変遷

「silk road」は，2011 年ごろに違法物品取引サイトとして出現し，買い手と売り手を仲介し，その手数料をとることで利益を上げていた．2013 年に FBI などによって「silk road」の運営者が検挙されたが，その後まもなく「silk road2.0」を名乗るサイトが出現し，同じ手口で仲介手数料を得ていた．翌年，ふたたび FBI などによって「silk road2.0」の運営者が検挙された．しかしその翌年「silk road 3 reloaded」を名乗るサイトが出現した．このサイトは 2017 年に一旦，その運営者等によって閉鎖されたが，その後「silk road 3.1」が出現した．ほかにも，Wegberg [12]らによる違法物品取扱サイトの長期観測結果によれば，大規模な違法物品取扱サイトの最短生存期間は 6 ヶ月であった．また，Soskara ら [13]の長期観測結果によれば，違法物品取扱サイトの信頼性は 70%以下で，稼働率が悪く，意図したタイミングでアクセスすることができないこともあることを示唆している．このように，違法物品取引サイトが 1 つ消えると，別の 1 つが現れるようなエコシステムに関しては，変化が定期的にあり，またアクセスできないことも念頭に，継続的に違法物品取扱サイトを継続的に監視し，圧力を強めていくことが前記のような社会問題の解決に肝要であると思料される．

2.2 関連研究との差異

まず，ダークウェブではその仕組みによりアクセス先サーバの所在を IP アドレスで特定することを困難にしているため，これを悪用して違法な物品を取り扱うサイトが多数存在しており，これを特定把握し，犯罪インフラとして使用されないような取り組みを案出することが社会貢献につながりうる．また一般的なクローリングとは異なり，ダークウェブのクローリングには Tor のような特殊なソフトが必要である．さらにダークウェブに存在するサイトは，主要な検索エンジンにはインデックスされておらず，登録されていない．このため，サイトを捜索した上で巡回する専用の手法が必要となる．

ダークウェブにおける違法物品取扱サイトを調査するための手段として，クローリングならびにスクレイピングを行ってデータ収集を行った先行研究として，脆弱性などの脅威情報の収集を目的としたもの [14]や，収集した HTML データの分類を目的としたもの [15]がある．しかし，いずれの研究も，Web ページのテキストに着目したものであり，ダークウェブを構成している各ホストの HTTP ヘッダの特徴等については示されていない．

このほか，ダークウェブをクロールし，収集した結果を機械学習やルールベースのアルゴリズムによって分類した先行研究が存在する．Ghosh [16]らは，Bag-of-Words [17]による特定の単語の出現頻度を特徴量とすることで，ダークウェブをクローリングした結果を分類した．しかし，この際に使用したデータセットは 529 サイト分と，やや少ない印象を覚える．また，この先行研究では，ダークウェブのサイトを”Drugs”，”Hacker”，”Weapons”の 3 つのカテゴリに分類するが，実際のところこれらのカテゴリに分類できないサイトもダークウェブには存在している．さらには，Bag-of-Words に指定している単語が隠語の使用に変化した場合，単語の再定義と再学習をする必要が発生してしまう．そこで，本章ではこうした単語や隠語の変化の影響を受けにくい，HTTP ヘッダに着目した特徴量を作成することで，違法物品取引サイトの検知を可能とする手法を提案する．

2.3 データセットの作成手法と分析

2.3.1 クローリングによるデータの収集

本章では，監視方法とその評価を達成するために，次のような手順を提案し，Tor におけるダークウェブの違法物品取扱サイトのデータを収集する．

① ダークウェブで使用されている.onion ドメインを持つ URL を，ダークウェブ内の検索サイト，Wiki ページ等から収集し，クローリング先 URL の初期巡回先リストを作成する
② 同リストの URL に対してクローリングを行い，HTTP ヘッダのデータを蓄積する
③ 蓄積したデータにアノテーションを実施する
④ 同データの分析を行い，違法物品取引サイトと他のサイトで使用されているミドルウェアの特徴に差異がないかといった観点での，詳細な調査を行う

ダークウェブのクローリングを行うためには，.onion という，特有のドメインを持つ URL を事前に収集しておき，かかる URL に対してアクセスをしなくてはならない．このため，2019 年 6 月 12 日に，こうしたダークウェブの URL を Hidden Service 専門の検索サイト [18]，およびダークウェブ内に設けられた Wiki ページ等から収集し，クローリング先 URL の初期巡回先リストを作成した．その結果，初期巡回先リストとして 5763 サイト分の.onion ドメインを持つ URL を収集した．その上で，巡回用のプログラム（クローラ）に同初期巡回先リストを使用して，クローリングを実施した．

先の手順で作成した初期巡回先リストを使用し，2019 年 6 月 14 日にクローリングを行った．クローラには Python3.6 を使用し，クローリング結果の出力には JSON [19]を用いてファイルに出力した．クローラの通信には requests パッケージ [20]を使用し，Tor Browser Bundle の User-Agent を設定して HTTP リクエストを送信した．クローリング結果の JSON 出力例を図 **3** に示す．それぞれの JSON 要素については表 1 の通りである．なお forum 要素は，後のアノテーションで使用するため，収集時点ではデフォルト値として 0 を設定している．


```
{
  "headers": {
    "Server": "nginx",
    "Date": "Fri, 14 Jun 2019 01:26:06 GMT",
    "Content-Type": "text/html",
    "Content-Length": "162",
    "Connection": "keep-alive"
  },
  "snapshot": "<html>\r\n<head><title>403 Forbidden</title></head>
r><h1>403 Forbidden</h1></center>\r\n<hr><center>nginx</center>\r\
  "forum": 0
}
```


図 3: **クローリング結果の JSON 出力例**

クローリングを実施した結果，4340 サイトから HTTP レスポンスデータを得ることができた．なお，Sarah Jamie Lewis の報告 [21]によれば，2017 年の 3 月の時点で，ダークウェブ全体のサイト数はおよそ 4400 程度であったという．また，本研究のクローリングを実施したのと同時期に，Fresh Onions [22]と呼ばれる，ダークウェブをクローリングし，その結果を表示することのできるオープンソースツールが運用され，ダークウェブ上に蔵置されていた．その結果によると，同サイトが本研究のクローリングを行ったのと同時期にアクセス可能とリスト表示されたダークウェブのサイト数は 1490 であった．また，インターネットの情報収集のためのソフトウェアを開発している Hunchly [23]によると，本研究のクローリングを行ったのと同時期にアクセス可能なダークウェブのサイト数は 4584 であった．したがって，Hunchly の収集結果と比較して本研究のクローリング結果は，単純なサイト数の比較として 94.67%と，やや少ないものの，ダークウェブのサイトは頻繁に停止したり，アクセス不能になったりする傾向があるため，ダークウェブのクローリング結果の網羅性としては充分であるという蓋然性は高いと考えられる．

表 1: 各 JSON 要素の詳細

headers	収集した HTTP ヘッダをオブジェクトで格納
snapshot	収集した HTML ページを文字列で保存
forum	違法物品取扱サイトかどうかのフラグ

クローリング結果のデータに対して，目視で違法物品取扱サイトかどうか，アノテーションを行った．アノテーションを行うにあたっては，アノテーション専用 Web アプリケーションを独自に開発し，収集した JSON データをサイト毎にロードして確認したうえで，アノテーションできるようにした．こうした作業を行った結果，ダークウェブのクローリングの結果得られた 4340 サイトのうち，41%にあたる 1799 サイトが違法物品取扱サイトと確認できた．

アノテーションを行う上で，違法物品取引サイトであると判断したのは，次のようなサイトである．

A) 違法薬物の取扱
B) ・サイバー攻撃サービスの提供
C) ・重火器の販売
D) ・偽造クレジットカードや偽造身分証明書の販売
E) ・暗号資産ミキサー [24]
F) ・詐欺サイト
G) ・児童ポルノ
H) ・著作権侵害コンテンツの取扱
I) ・その他，犯罪活動に結びついていることが思料されるサイト

なお「その他」には，たとえばマネーロンダリングを企図していると思料される，異常に安価な販売価格のスマートフォンなどの電子ガジェット等の販売サイト等も含まれる．

一方，違法物品取扱サイトではない，クローリング結果の 59%に含まれるサイトはおよそ次のとおりである．

(ア) 個人のブログサイト
(イ) 反政府的な主張を掲載しているサイト
(ウ) 新聞社などの報道機関が運営しており，投稿者は Tor を使用することで自身の匿名性を維持したまま，報道機関に対して情報提供ができるサイト
(エ) 自作の詩や小説，パロディなどの紹介
(オ) ダークウェブのリンク集サイト
(カ) 一般的な掲示板サイト

(キ) 法執行機関によってテイクダウンされ，閉鎖されたというメッセージが記載されたサイト

2.3.2 HTTP ヘッダの概要

HTTP および HTTPS プロトコルにおいては，サーバに通信を行う際，クライアントからどういった情報を要求し，どのようなコンテンツを応答するのかを定義する文字列が HTTP ヘッダとして設定されている．HTTP ヘッダにはクライアントからの要求を示すリクエストヘッダと，サーバからの応答を示すレスポンスヘッダの 2 種類が存在する．本章においては，特に示さない限り後者の HTTP レスポンスヘッダを「HTTP ヘッダ」として使用する．

表 2: 代表的な HTTP レスポンスヘッダの例

レスポンスヘッダ名称	内容
Server	Webサーバの名称やそのバージョン
Date	現在の日付（GMT）
Last-Modified	アクセス先ファイルの更新日
Content-Length	レスポンスのバイト単位の長さ
Content-Type	レスポンスのMIMEタイプ
Expires	リソースの有効期限
Pragma	キャッシュの有効化・無効化

表 2 に代表的な HTTP レスポンスヘッダの例を示す．このように HTTP ヘッダにはサーバの状態や，当該のサーバで使用されているソフトウェアの名称やバージョンなどが含まれている．この為，ダークウェブ内で使用されているサーバの特徴を含んでいると思料される．そのための調査を次に行った．

2.3.3 収集したデータに含まれる HTTP レスポンスヘッダの傾向分析

全体的な傾向を把握するため，クローリングの結果として得られたデータのうち，各サイトから応答された HTTP ヘッダをもとにした分析を行った．非違法物品取扱サイトでの HTTP ヘッダの出現割合のうち，上位 30 を対象に違法物品取扱サイトのものと比較した．単純な出現数では母数が異なるために出現割合を使用して比較した．その結果を図 4 に示す．

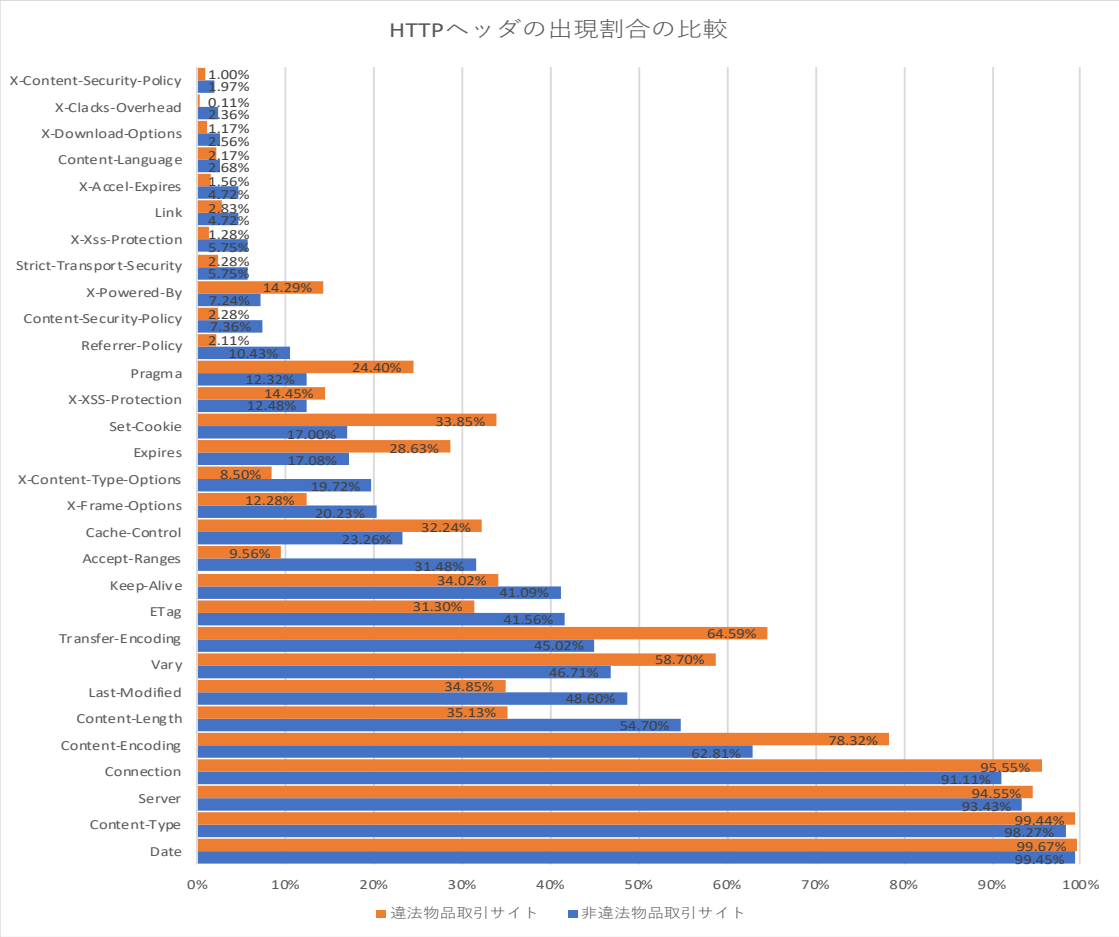

図 4: 非違法物品取扱サイトでの HTTP ヘッダの出現割合のうち，上位 30 種類を対象に違法物品取扱サイトのものと比較した結果

このように，一部のヘッダにおいては違法物品取扱サイトのほうが顕著に出現する傾向があり，そのまた逆のこともある．たとえば Pragma ヘッダは非違法物品取扱サイトの 12.32%で出力されたが，一方，違法物品取扱サイトが当該ヘッダを伴う割合は 24.40%であった．Pragma ヘッダを出力する違法物品取扱サイト 439 のうち，キャッシュを残すように指定しているサイトは 2 サイトのみであった．これは，違法物品取扱サイトがキャッシュをクライアント側に残さないよう指定していることで，匿名性を高めるような配慮がなされていると思料される．

ほかにも，X-powered-by ヘッダは非違法物品取扱サイトの 7.24%出力されたが，一方，違法物品取扱サイトが当該ヘッダを伴う割合は 14.29%であった．当該のヘッダを伴う違法物品取扱サイト 257 サイトのうち，207 サイトが当該ヘッダに含まれる文字列として”PHP”を含んでいた．これは違法物品取扱サイトが PHP ベースの Web アプリケーションによって運営されていることを示している．Set-Cookie ヘッダも非違法物品取扱サイトの 17.00%に比べて，違法物品取扱サイトが当該ヘッダを伴う割合は 33.85%であった．これは何らかのログイン方法が違法物品取扱サイトには存在することで，セッション維持のために使用されていると思料される．

一方で，Content-Length ヘッダに関しては非違法物品取扱サイトの 54.70%が出力しているのに対して，違法物品取扱サイトでは 35.13%であった．同様に，Last-Modified ヘッダでは非違法物品取扱サイトの 48.60%に対し，違法物品取扱サイトでは 34.85%であった．このような差が何故生まれているかは現時点においては未詳であるが，前記のような非違法物品取扱サイトと違法物品取扱サイトの HTTP ヘッダの出力内容には何らかの傾向や偏りがあると思料される．

2.3.4 隠語の変化の影響を受けない特徴量の設計

前記のような HTTP ヘッダの出現傾向を取り入れて，違法物品取扱サイトを自動的に検出する方法について検討し，次の 2 点を満たすモデルを構築することを目的とした．

① 実務的な有用性を考慮し，見逃しの少ないモデルの構築
② 隠語の変化に左右されないモデルの構築

前者については，誤検出よりも，見逃してしまうことによる，いわばサイバー犯罪等の端緒を把握できなくなってしまう事態を抑止につなげることができる．後者については，ダークウェブをクロールした結果を見ると，たとえば違法薬物について”Drug”としていることもあれば”Cristal”，”Powder”などと表記されていることもある．また，偽造クレジットカードであれば”Fake Cards”であることもあれば単に”Plastic”と表記されることもある．こうした隠語の変化に追従し，Bag-of-Words などの単語抽出を繰り返し，さらに再学習をすることは運用負荷が高い．このため，隠語の変化に左右されないモデルの価値が高いといえる．そこで今回は，HTTP ヘッダを特徴量に使用することでこうした目的を達成することを提案する．

2.4 分類手法

分類にはアンサンブル学習を使ったランダム木と，その一種である LightGBM を使用した．

2.4.1 ランダム木

ランダム木は Leo Breiman によって 2001 年に提案された [24]アルゴリズムである．特徴は「バギング」と呼ばれる，ランダムにサンプリングされた訓練データを用いることで，複数の決定木を学習することにある．このようにして得られた複数の決定木の結果を組み合わせるアンサンブル学習によって識別，分類などを行うものとなっている．

2.4.2 LightGBM

LightGBM は 2017 年に発表された手法であり，前記のランダム木に勾配ブースティングを組み合わ

せた，いわゆる GradientBoosting Decision Tree [25]の手法の一つである．GradientBoosting Decision Tree は，ランダム木に勾配ブースティングを組み合わせることによって性能を向上させたアルゴリズムである．LightGBM は，この GradientBoosting Decision Tree に，データを削減する Gradient-based One-side Sampling と特徴を減らす Exclusive Feature Bundling をさらに組み合わせたものである．Gradient-based One-side Sampling では，各反復における勾配が小さいデータはよく学習できているとしてデータを無視し，その結果，データ数を削減する．Exclusive Feature Bundling では疎なデータに関して，データを複数にまとめ，そのまとめた単位毎に学習を行うことで計算量を減らすことができる．

2.5 分類器の評価指標

本研究における分類では，対象が違法物品取扱サイトであるか，そうでないかの 2 値分類を行う．このとき，ラベルが正であり，予測ラベルも正で正しい場合は True Positive(TP)と呼ばれる．さらにラベルは正であり，予測ラベルが負で誤りの場合は False Negative(FN)と呼ばれる．そしてラベルが負であり，予測ラベルが負で正しい場合は True Negative(TN)と呼ばれる．ラベルは負であり，予測ラベルが正で誤りの場合は False Positive(FP)と呼ばれる．分類結果の正誤評価を表 3 に示す．

表 3: 分類結果の正誤評価

		真のラベル	
		正例	負例
予測ラベル	正例	TP (True Positive)	FP (False Positive)
	負例	FN (False Negative)	TN (True Negative)

また，表 3 から次の式を用いて正解率，適合率，再現率を算出し，分類結果の評価指標とする．

$$正解率 = \frac{\mathrm{TP} + TN}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN} + \mathrm{TN}}$$

$$適合率 = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

$$再現率 = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}$$

正解率は，分類結果の精度を示している．適合率は分類器の正確性を表す指標である．再現率は網羅性の指標であり，分類器が正解をどの程度の割合で特定できているかを示す．よって再現率が高いことは，分類器の性能として見逃しが少ない，より性能の高い分類を行っていることを示すことになる．

2.6 HTTP ヘッダを特徴量に採用したモデルの案出と検証

実際に特徴量に使用した HTTP ヘッダを図 5 に示す．本章では図 5 の HTTP ヘッダの有無を特徴量にしたものと，長さと行数を特徴量にしたものの 2 種について，より見逃しの少ないモデルを目的に比較検討を行った．

特徴量に使用したヘッダ
Date, Server, Expires, Cache-Control, Vary, Content-Encoding, Content-Length, Keep-Alive, Connection, Content-Type, Last-Modified, Transfer-Encoding, ETag, X-Powered-By, Content-type, X-Content-Type-Options, X-Frame-Options,Referrer-Policy,X-Xss-Protection,X-Clacks-Overhead,Surrogate-Key,X-XSS-Protection,X-Cache-Status,Content-Language,X-Accel-Expires, pragma, expires, Access-Control-Allow-Origin, Access-Control-Allow-Methods, Access-Control-Allow-Credentials, Access-Control-Allow-Headers, Feature-Policy, Strict-Transport-Security, x-xss-protection, x-content-type-options, Content-Security-Policy,X-Nginx-Cache-Status, X-Server-Powered-By, Status, X-Request-Id, X-Runtime, Cache-control, X-Robots-Tag, X-Pad, Link, P3P, X-Content-Security-Policy, X-WebKit-CSP, X-Cache, X-Check-Tor, X-Generator, X-ID, Public-Key-Pins-Report-Only, X-Pingback, X-UA-Compatible,Content-Location, TCN, X-Garden-Version, StickyNotes-Url, Composed-By, X-Spip-Cache, X-Varnish-Ttl, X-Varnish, Via, grace, X-Varnish-Age,X-AspNetMvc-Version, X-AspNet-Version, X-Frontend, X-Page-Speed, access-control-allow-origin, access-control-allow-headers, referrer-policy, Age, X-Served-By, X-Cache-Hits, X-Timer, content-encoding, X-Download-Options, Content-language, content-security-policy, X-FB-Debug, WWW-Authenticate, Expect-CT, X-DNS-Prefetch-Control, Etag, X-Rack-Cache, X-GitHub-Request-Id, X-Fastly-Request-ID, CF-RAY, Clear-Site-Data, X-IPS-LoggedIn, X-IPS-Cached-Response, Access-Control-Allow-Method, x-nyt-data-last-modified, X-PageType, X-VI-Compatibility, x-nyt-route, x-nyt-backend, X-Origin-Time, x-gdpr, x-nyt-fastly-info-state, x-nyt-final-url, X-API-Version, debug-var-nyt-env, debug-var-nyt-force-pass, x-nyt-continent, x-nyt-country, x-nyt-region, x-nyt-latitude, x-nyt-longitude, x-nyt-city, x-nyt-gmt-offset, x-nyt-postal-code, x-nyt-geo-hash, device_type, Authorisation, X-Cache-Lookup, X-Cloud-Trace-Context, Public-Key-Pins, last-modified, Easter-Egg, alt-srv, Onion-Location, MS-Author-Via, It-Vends-Execution-Time, It-Vends-Memory-Usage, x-amz-id-2, x-amz-request-id, Alt-Svc, X-Contact, Frame-Options, content-type, connection,cache-control, Access-Control-Expose-Headers, Accept-CH, Accept-CH-Lifetime, Proxy-Connection, Bitcoin-Payment-URI, X-DIS-Request-ID, X-Permitted-Cross-Domain-Policies, X-Amz-Cf-Pop, X-Amz-Cf-Id, Upgrade, set-cookie, X-RateLimit-Limit, X-RateLimit-Remaining, etag, X-Soup, Content-Security-Policy-Report-Only, Serve, X-Fuck, sn, timer, X-Dns-Prefetch-Control, ws, X-Application-Context, X-Fry, X-Content-Digest, ID, SESSION, x-powered-by, x-frame-options, x-nyt-service-id-backend-name, x-nyt-backend-ip, x-nyt-backend-port, X-Follow-The-White-Rabbit, Proxy-Agent, X-App-Name, X-VarnishCacheDuration, X-ESI, X-App-Response-Time, X-Rank-Age, X-Rank-Timestamp, X-Stream-Age, X-Stream-Timestamp, Fastly-Restarts, x-download-options, x-permitted-cross-domain-policies, X-Ua-Compatible, Key, X-Device, X-Loggable, X-Domain, X-XRDS-Location, vary, Content-Disposition, Content-MD5, refresh, X-Do-A-Kickflip, Charset, X-Content-Type, Cache-Tags, X-Zendesk-User-Id, X-Zendesk-Origin-Server, Protocol, CF-Cache-Status, X-Nginx-Fastcgi-Cache, Allow, X-Sorting-Hat-PodId, X-Sorting-Hat-ShopId, X-ShopId, X-ShardId, X-Alternate-Cache-Key, X-Shopify-Stage, X-Dc, NEL, Report-To, X-Hyper-Cache, X-xxxx, X-nginx-Cache, Refresh, X-Queued-Time-Spent, X-Sql-Time-Spent, X-Memcached-Queries, X-Sql-Queries, X-Python-Time-Spent, X-Memcached-Time-Spent, X-FRAME-OPTIONS,X-ABLATIVE-HOSTING, X-GEO, Accept-Charset, X-XF-Debug-Stats, Surrogate-Control, Tk, X-Content-Powered-By, X-Logged-In, CF-Chl-Bypass,X-Mod-Pagespeed, Content-Range, X-Use-Gopher, X-Future, X-Irritate, If-By-Whiskey, X-GUploader-UploadID, X-Discourse-Route, X-Discourse-TrackView,server, X-ob_mode, strict-transport-security,X-Origin-Host, X-Backend, X-BF-cdn-url, X-Drupal-Cache, X-Account-Management-Status, content-length, X-Drupal-Dynamic-Cache, X-Backend-Status, X-Instance-ID, PICS-Label, X-CSRF-Token, X-Openbazaar, X-hacker, X-rq, X-Tweakers-Server, X-Evil-Bit,Upgrade-Insecure-Requests, X-TTL, X-BB-ID, X-Galaxy3, Content-length, x-goog-generation, x-goog-metageneration, x-goog-stored-content-encoding,x-goog-stored-content-length, x-goog-hash, x-goog-storage-class, X-Api-Version, Calibre-Uncompressed-Length, content-disposition, x-now-cache, x-now-trace, x-now-id, access-control-allow-credentials, access-control-expose-headers, x-request-id, X-Backend-Server, Access-Control-Max-Age, X-Varnish-Cache, X-Frame_options, X-XSS-Protestion, X-Hudson-Theme, X-Hudson, X-Jenkins, X-Jenkins-Session, X-Hudson-CLI-Port, X-Jenkins-CLI-Port, X-Jenkins-CLI2-Port, X-Instance-Identity, superkuh, Client-Peer, Client-Response-Num, Client-Transfer-Encoding, X-Debug-Channel, X-Debug-GoToStatic, X-Debug-GoToTypo3, X-Debug-PATH-INFO, X-Debug-TraceDeco, X-Debug-Vary, X-Debug-Vimeo, X-Taz-Debug-20160113, X-Taz-Debug-20160113a, X-Taz-Mode, X-Taz-Server, X-Debug-ResponseTrace, X-Debug-PATH_INFO, Where, date, X-Clearnet-URL, Host, X-OCCRP-Fasada-Content, X-Fasada-Cache, X-Matomo-Request-Id, X-Proxy-Cache, Content-script-type

図 5: 特徴量に採用した HTTP ヘッダー一覧

2.6.1 HTTP ヘッダの出力有無を特徴量に使用したデータセットを使用した場合

これらの HTTP ヘッダの有無を特徴量に変換した．具体的には特定のヘッダが存在した場合には 1 を，なかった場合は 0 を設定した行列を特徴量とした．この方法による特徴量変換の例を図 5 に示す．なお，”Rogue”列は違法物品取扱サイトか，そうでないかのラベルである．分類にはランダム木と LightGBM を採用し，データセットをシャッフルした上で，その 80%を学習用とし，残りの 20%をテスト用に分割し，訓練と検証を行った．表 4 にその結果を示す．

	Rogue	Date	Server	Expires	Cache-Control	Vary	Content-Encoding	Content-Length	Keep-Alive	Connection	Content-Type
0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0
1	0.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	1.0	1.0
2	0.0	1.0	1.0	0.0	0.0	1.0	1.0	0.0	0.0	1.0	1.0
3	0.0	1.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	1.0
4	0.0	1.0	1.0	1.0	1.0	0.0	0.0	1.0	0.0	0.0	1.0

図 6: HTTP ヘッダの有無を特徴量に変換した例

適合率と再現率の差があり，適合率のほうが高いということは，見逃しが多く存在しているために，分類器全体の性能に影響を及ぼしている蓋然性が高い．このため，特徴量エンジニアリングを再度行い，さらに見逃しの少ない特徴量を設計する必要があるという結論に至った．

表 4: HTTP ヘッダの出力有無を特徴量にした実験結果

	正解率	適合率	再現率
ランダム木	82.0%	84.2%	67.7%
LightGBM	78.1%	77.5%	63.7%

2.6.2 HTTP ヘッダの長さと行数を特徴量に採用した実験

特徴量の設計について再検討を行うため，各ヘッダに設定された値の長さでヒストグラムを取得した．その結果，いくつかのヘッダでは値の長さに如実に傾向がみえることがわかった．図 7 に，その一例として Content-Length ヘッダの値の長さのヒストグラムを示す．なお，図中のグラフ青色が非違法物品取扱サイトのものであり，橙色が違法物品取扱サイトのものである．このようにヘッダの値の長さに傾向があると仮定し，ヘッダの値の長さをもとにした特徴量を取得した．

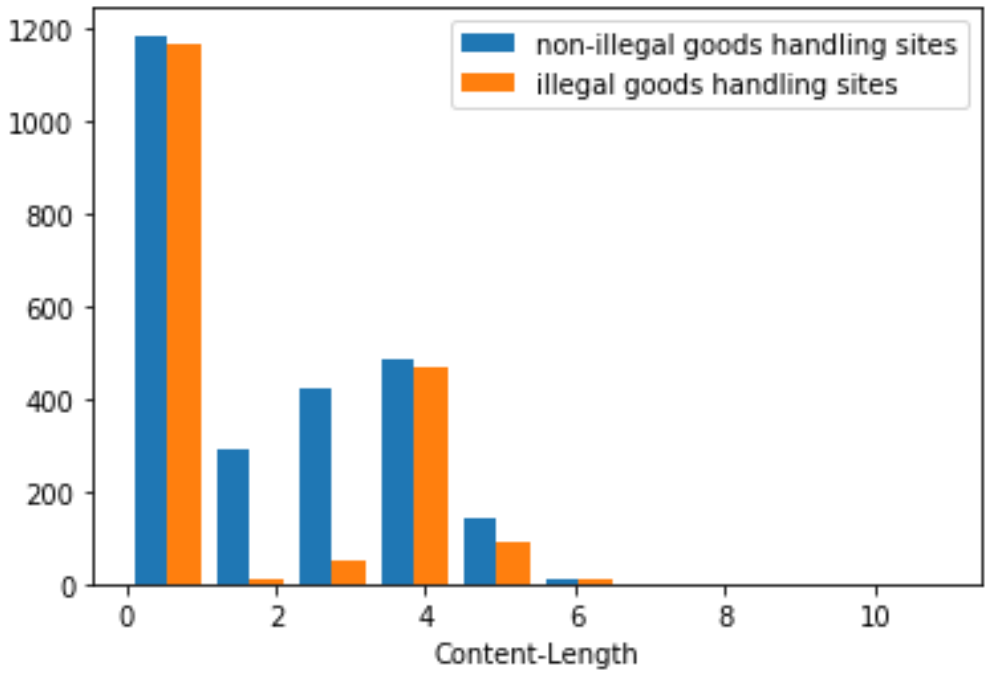

図 7: Content-Length ヘッダの値の長さのヒストグラム

また，HTTP ヘッダの行数もヒストグラムを取得したところ，いくらかの傾向があることもわかった．図 8 にそのヒストグラムを示す．このように傾向が得られたことから，HTTP ヘッダの値の長さと，HTTP ヘッダの行数を特徴量とした．この特徴量の例を図 9 に示す．

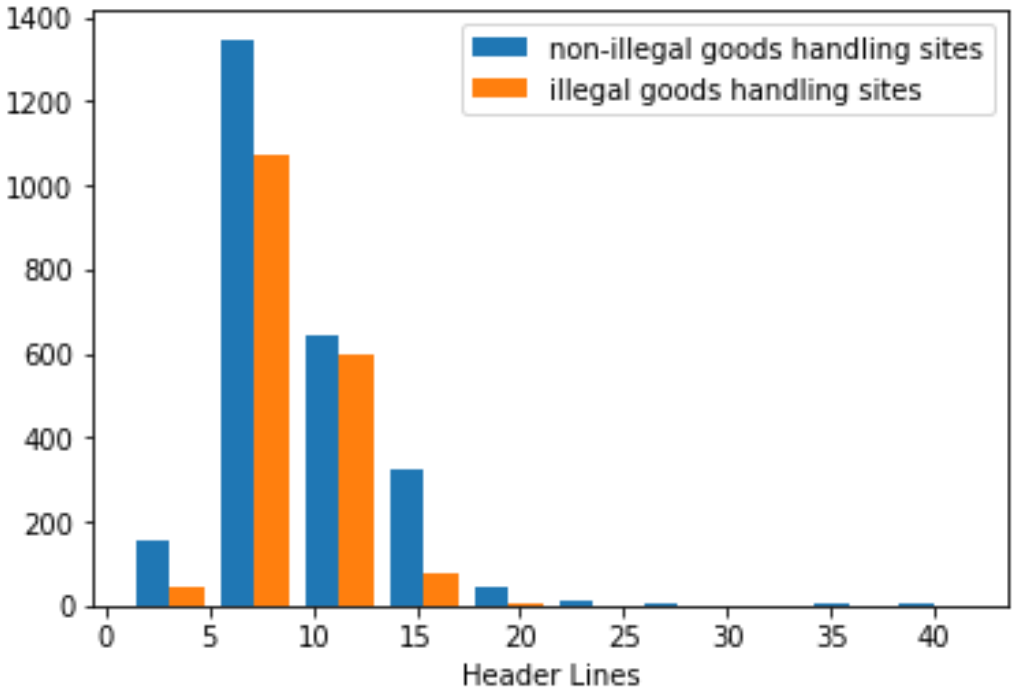

図 8: HTTP ヘッダの行数のヒストグラム

	Rogue	Date	Server	Expires	Cache-Control	Vary	Content-Encoding	Content-Length	Keep-Alive	Connection	Content-Type
0	0.0	29.0	15.0	0.0	0.0	0.0	0.0	4.0	0.0	0.0	24.0
1	0.0	29.0	20.0	0.0	0.0	0.0	0.0	2.0	18.0	10.0	9.0
2	0.0	29.0	5.0	0.0	0.0	15.0	4.0	0.0	0.0	10.0	24.0
3	0.0	29.0	12.0	0.0	0.0	0.0	4.0	0.0	0.0	10.0	24.0
4	0.0	29.0	12.0	29.0	18.0	0.0	0.0	3.0	0.0	0.0	30.0
5	1.0	29.0	12.0	0.0	0.0	15.0	4.0	0.0	0.0	10.0	9.0
6	0.0	29.0	5.0	29.0	35.0	23.0	4.0	0.0	0.0	10.0	24.0
7	1.0	29.0	0.0	29.0	35.0	0.0	4.0	0.0	0.0	10.0	24.0
8	1.0	29.0	5.0	0.0	0.0	15.0	4.0	0.0	0.0	10.0	9.0

図 9: HTTP ヘッダの値の長さと行数を特徴量に変換した例

この特徴量を使用し，前記の HTTP ヘッダの有無を特徴量に使用した場合と同じ条件で実験を行った．その際の結果を表 5 に示す．

表 5: HTTP ヘッダの値の長さと，同行数を特徴量とした実験結果

	正解率	適合率	再現率
ランダム木	85.8%	83.3%	80.7%
LightGBM	80.3%	79.7%	68.0%

ランダム木を使用した分類において，適合率は前記の実験結果より微小な低下がみられたが，再現率が 80%を超え，かつ正解率も 85.8%へ向上し，より見逃しの少ないモデルに改善されたといえる．さらに，汎化性能を評価するために 5 分割交差検定により正解率を測定したところ，その平均値は 83.1%であった．

ただし，本手法ではランダム木を用いているにも関わらず，特徴量選択がヒューリスティックになされているため，客観的に特徴量の重要度を計測することが肝要である．このため，選択された特徴量が妥当であることを確認するために，ジニ不純度をもとにした特徴量の重要度を計測した．図 10 にその結果のうち，上位 10 件のヒストグラムを示す．特徴量として Server ヘッダとヘッダの行数が相対値として 0.1 を超えているが，そのほかの特徴量も一様に寄与しており，何らかの突出した特徴量が存在しているわけではないことが明らかになった．

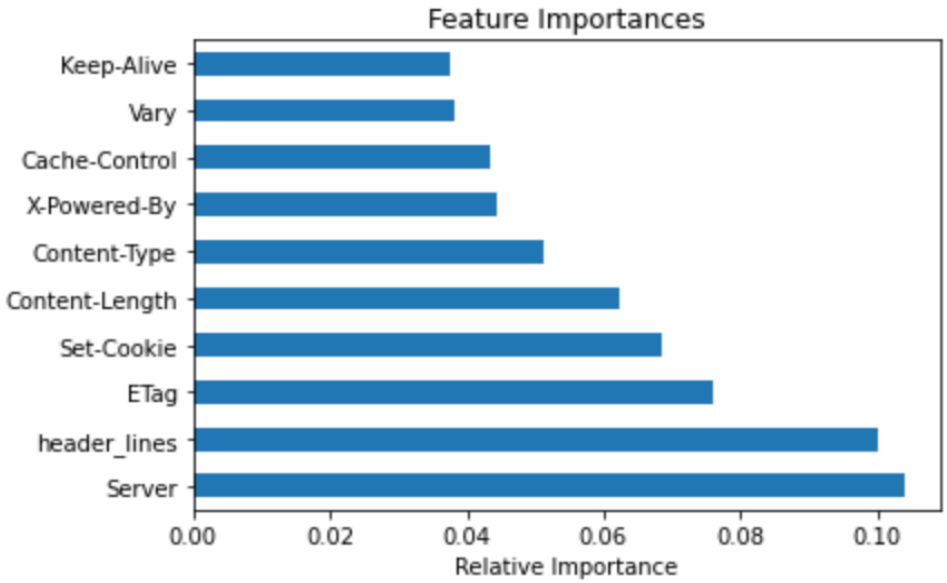

図 10: 特徴量の重要度の上位 10 件のヒストグラム

2.7 まとめと今後の課題

本研究では，まずダークウェブの違法物品取扱サイトを中心にクローリングを行った．クローリングにより蓄積したデータにアノテーションを行った上で，同データの分析を行い，特徴量を設計して分類器を開発した．その結果，再現率が 80%を超え，かつ正解率も 85.8%を得られる分類器を開発できた．また，従来型のセキュリティ対策の領域だけではなく，違法物品取扱サイトの検出のような新しい領域に関しても，機械学習の有効性を確認することができた．したがって，こうした新規の領域に対してこうしたアプローチを進めていくことがよりよい社会に貢献する手段に思える．しかしながらそれは本当だろうか．この疑問を解き明かすために，次章の研究に取り組むことにした．

第3章　次世代型ウイルス対策ソフトとハイブリッド検出を実装するウイルス対策ソフトに対する回避攻撃

3.1 次世代型ウイルス対策ソフトとは

近年，次世代型ウイルス対策ソフト（Next Generation Anti-Virus:NGAV）という名称で，従来型のウイルス対策ソフトでは対応が困難であった，未知のマルウェアまでをも検出が可能であることを長所として広報している，商用の製品が市場に投入されている [26]．それら製品の特徴のひとつとして，AI 技術・機械学習を使用することで「マルウェアらしさ」を数値化し，検知・駆除の対象であることを明らかにすることで予測防御を実現することができるという．そのために，数十万種類のマルウェアからなるデータセットで学習・訓練させた機械学習ベースの分類器を使用することで，未知の脅威も予測し，かつ高確率で検知・駆除が可能であるという．

機械学習の領域では，分析対象のデータに加工することによって標的の AI 技術を使用した製品やサービスに誤分類を生じさせる攻撃方法は回避攻撃と呼ばれる [27]．本章においては，NGAV への回避攻撃について検討をしたうえで実験を行った．さらに，回避率についてハイブリッド型のウイルス対策ソフトとの比較を行った．

3.2 関連研究

機械学習ベースの回避攻撃手法を検証する上で重要となるのが環境制約条件である．これは制約条件の有無によって，回避攻撃の成功の可否・難易度が大きく異なってくるためである．図 1 に，環境制約条件と難易度について整理した図表を示す．

基本的に，標的としているウイルス対策ソフトが使用しているモデルに関する攻撃者の知識量の多寡によって，回避攻撃の難易度は変化する．まず，回避攻撃におけるホワイトボックステストとは，アルゴリズムや，各特徴量の重みや偏りといった標的のウイルス対策ソフトの使用しているモデルと，そして判定時の「マルウェアらしさ」の数値（スコア）に関する情報を攻撃者側が把握している，ということである．このため，特定の特徴量に該当する箇所のデータをより多く，もしくは少なくすることで誤検知や見逃しといった事象を発生させやすくすることが可能となりうる．ホワイトボックステストによる回避攻撃の関連研究として Papernot ら [28]による研究がある．

図 11: 機械学習ベースのウイルス対策ソフトに対する回避攻撃の環境制約条件

続いて，グレーボックステストでは，特徴量の重みなどは不明だが，スコアを特定する事ができる．したがって，ホワイトボックステストよりも攻撃の難易度は高まる．ただし，様々な回避攻撃の手法を試みることで，スコアの上下を確認することによってその手段の有効性を確認することができる．グレーボックステストの関連研究として Xu ら [29]の研究がある．

そしてブラックボックステストでは，前期のような情報は全く得られず，単純にマルウェアと判断されたか，そうでないかの真偽値の判定結果のみが得られる．このためブラックボックステストは回避攻撃の中で最も攻撃の難易度が高いものとなる．ブラックボックステストの関連研究としては Hu ら [30]の研究がある．

なお，これら 3 つの環境制約に関係なく，適応型攻撃(Adaptive Attack)と呼ばれる，攻撃者が標的の反応や結果に基づいて戦略を適応させる種類の攻撃手段も案出されているが，主たる対象は画像の分類エンジンであり，ウイルス対策ソフトに対する同種の攻撃手法は先行研究では確認されていないため，本論文では対象外とする．

また，一般に販売されている機械学習ベースのウイルス対策ソフトに対する回避攻撃に関連した先行研究が存在している．Ashkenazy ら [31]は，商用の NGAV である Cylance 社の製品に対してリバースエンジニアリングを行い，当該製品が内部で使用している「マルウェアらしさ」の数値を検体のスキャン毎に取り出せるようにし，グレーボックステストを可能にさせた．さらに，当該製品が内部でホワイトリストとして使用していると思料される，ゲームソフトなどの開発元企業名やゲームの名称などの文字列を特定した．その上で，既知のマルウェア検体のファイル末尾にこれらのゲームソフトなどの正規のプログラムで使用されている文字列を大量に埋め込んで，「マルウェアらしさ」の数値の変化を確認した．すると，こうした文字列を大量に含むように修正した検体の「マルウェアらしさ」の数値は大幅に低下し，その結果当該製品は既知のマルウェアを見逃すようになった．しかしこの方法は，単純に大量の文字列を検体に追加で埋め込むことになり，プログラムとしてメモリ上に展開・実行される際に使用されるセクションデータとの

関係性も大きく損なうことになる．このため，マルウェアとしてのプログラムとしての動作不全，すなわち感染動作に支障をきたす可能性があり，回避攻撃は成功させても感染動作の不全により脅威としては低くなる．

また，Ceschin ら [32]は，別の回避攻撃によって機械学習ベースのウイルス対策ソフトを回避している．この手法では，オリジナルのマルウェア検体のバイナリセクションを PE ファイルのリソースセクション[33]に移動させる．そのうえで，実行時にはこのリソースセクションにあるオリジナルのバイナリデータをファイルに書き出し，実行するようなコードに書き換える．こうした手法をとることによって，ブラックボックステストにおいて最大で 99.99%の確率で機械学習ベースのウイルス対策ソフトを回避できたという．しかし，この手法はパッカー [34]などを使用してマルウェアを圧縮・難読化する手法に酷似しており，マルウェアそのものを大幅に書き換えるため，従来型のパターンマッチングを主に使用しているウイルス対策ソフトに対しても回避攻撃を実現できるとしている．しかし，同研究で採用されている手法はオリジナルの既知のマルウェア検体をファイルに書き出すため，この時点で NGAV ないしは従来型のウイルス対策ソフトの両方で検知される蓋然性が高い．このため，実世界での脅威としては低くなると思料される．

3.3 本研究の目的

前項のような実態を背景として，本研究では次の三点を目的としている．

① セクションデータなどとの不整合を生じさせず，マルウェアとしての感染動作には影響を与えない手法による回避攻撃を実現する
② 本研究で使用した回避攻撃は，機械学習を検出に使用しているゆえ生じる課題であることを明らかにする
③ 商用の NGAV に対するリバースエンジニアリング行為への倫理的ならびに適法性への配慮によりブラックボックステストを採用する

本研究と関連研究との関係性を表 6 に示す．

表 6: 関連研究と本章の実験との比較表

	対象	環境制約	攻撃手法	検知の比較
Ashkenazy らの研究	商用のAV製品	グレーボックス	文字列の追加	なし
Ceschinらの研究	商用のAV製品	ブラックボックス	セクションの移動	なし
本研究	**商用のAV製品**	**ブラックボックス**	**証明書による署名を通じた文字列の追加**	**あり**

まず，本研究では他の先行研究と同じく，商用の機械学習ベースのウイルス対策ソフトを回避攻撃の対象とし，環境制約は最も厳しいブラックボックステストとした．なおブラックボックステストとした理由は，グレーボックステストとした場合に付随的に発生するリバースエンジニアリング行為の適法性を考慮外にできる，という倫理的・法的配慮がある．

回避攻撃の手法には，感染動作そのものへ支障をもたらす影響を避けると同時に，機械学習ベースのウイルス対策ソフトの検出に影響を与えると仮説立て，証明書による署名を採用した．このような仮説を立てた理由は２つある．まず一点目は，既存のマルウェアに対して追加しやすい手段であり，セクションやメタデータといった箇所には変更を加えないため，マルウェアの感染活動そのものには影響を与えないことがある．

二点目は，機械学習ベースのウイルス対策ソフトのモデルの訓練に使用されているデータセットには，文字列を使った特徴量が含まれている点である．例えば Schultz ら [35]の研究はデータセットの特徴量としてバイナリに含まれるインポートアドレステーブルの API，文字列，バイト列が使用されている．ほかにも Ahmadi ら [36]は文字列の長さの分布を特徴量に使用したデータセットを作成して研究に用いた．このほか Islam ら [37]も文字列を特徴量としたデータセットを使用しておるほか，「EMBER」と呼ばれる機械学習ベースのウイルス対策ソフトの研究において最も著名なデータセット [38]でも，文字列の長さの分布のほかに，印字可能な文字列の数，長さの平均，エントロピー，URL の数と文字列由来の特徴量が採用されている．このように，機械学習ベースのウイルス対策ソフトが検出モデルの訓練用に使用しているデータセットの特徴量には，文字列ならびに文字列に由来したデータが採用されている蓋然性が高い．このため，こうした文字列に関連した特徴量を推定し，操作を行うことで回避攻撃を実行できる可能性を高められると思われる．

かかる目的で，マルウェアの感染動作などの本体部分には無影響で文字列を埋め込むための領域として，適した箇所は前記の通りデジタル証明書によるコード署名に割り当てられる領域である．なぜならば，一般にプログラムのコンパイル後にコード署名は行われるため，実行時に使用されるセクションデータとは別の領域に署名データが保存される．このため，既知のマルウェア検体に署名のためのツールなどを使用してコード署名を行っても，感染動作自体には支障がない．

さらに，オリジナルの既知の検体と，デジタル証明書による署名で文字列を追加した既知の検体とを，それぞれ NGAV とハイブリッド型のウイルス対策ソフトの両方で検出率を確認する．そしてこの検証は，先行研究にはなく新規性が特に得られる蓋然性がある．

3.4 実験手順

本研究では次の手順で実験を行った．

① 機械学習エンジン採用製品 X とハイブリッド型ウイルス対策ソフト A をインストールした 2 つの Windows 環境を準備した．
② Windows 上で動作する検体を VirusShare [39]から 3,000 件入手し，両方の環境でマルウ

ェアとして検出できた 1,065 検体を抽出した.

③ コンセプトドリフトの問題 [40]を回避するため，使用検体は 2022 年 3 月に同サイトに登録されたものを採用した．その理由は，製品 X のエンジンのアップデート履歴を確認すると，本実験で使用したバージョンのリリース日は 2022 年 3 月であった．このため，同製品が同年同月に共有された検体データを学習に使用している可能性があり，製品 X の検知率の向上に寄与する蓋然性が高いと考え，このような選択をした.

④ 自己署名のコード署名用証明書を作成したうえで，前記で抽出したマルウェアに Authenticode 形式でコード署名した.

⑤ 再度製品 X と A とで前記の手順で署名した既知のマルウェアをスキャンした.

製品 A を比較対象とした理由は，市場での占有性が高いと同時に，開発元のホームページ等に機械学習エンジンを同製品にハイブリッドで組み込んでいるといった記載が確認できたためである．なお，母集団となった 3,000 検体に対する製品 A の検出率は 37.1%であり，製品 X の検出率は 65.1%であった．本実験に使用した，両製品で検出できた 1,065 検体のバイトサイズの分布を図 12 に示す.

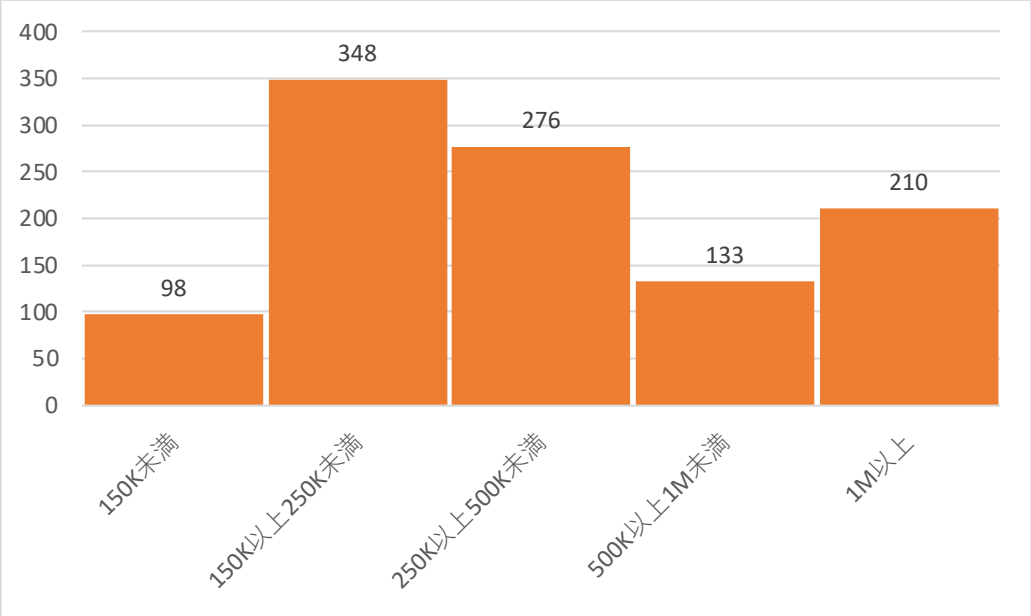

図 12: 実験に使用した 1,065 検体のバイトサイズの分布

また，同検体データの製品 A による検出名称について VirusTotal [41]を使用して取得し，検出名称をカウントしたところ 263 種が確認できた．VirusTotal で検出名称を特定できなかった検体が 25 検体あった．同カウント結果の上位 20 種の検出名称をに示す．なお，製品 X は製品の仕様上，検出名称を示さないため，製品 A による検出名称を使用した.

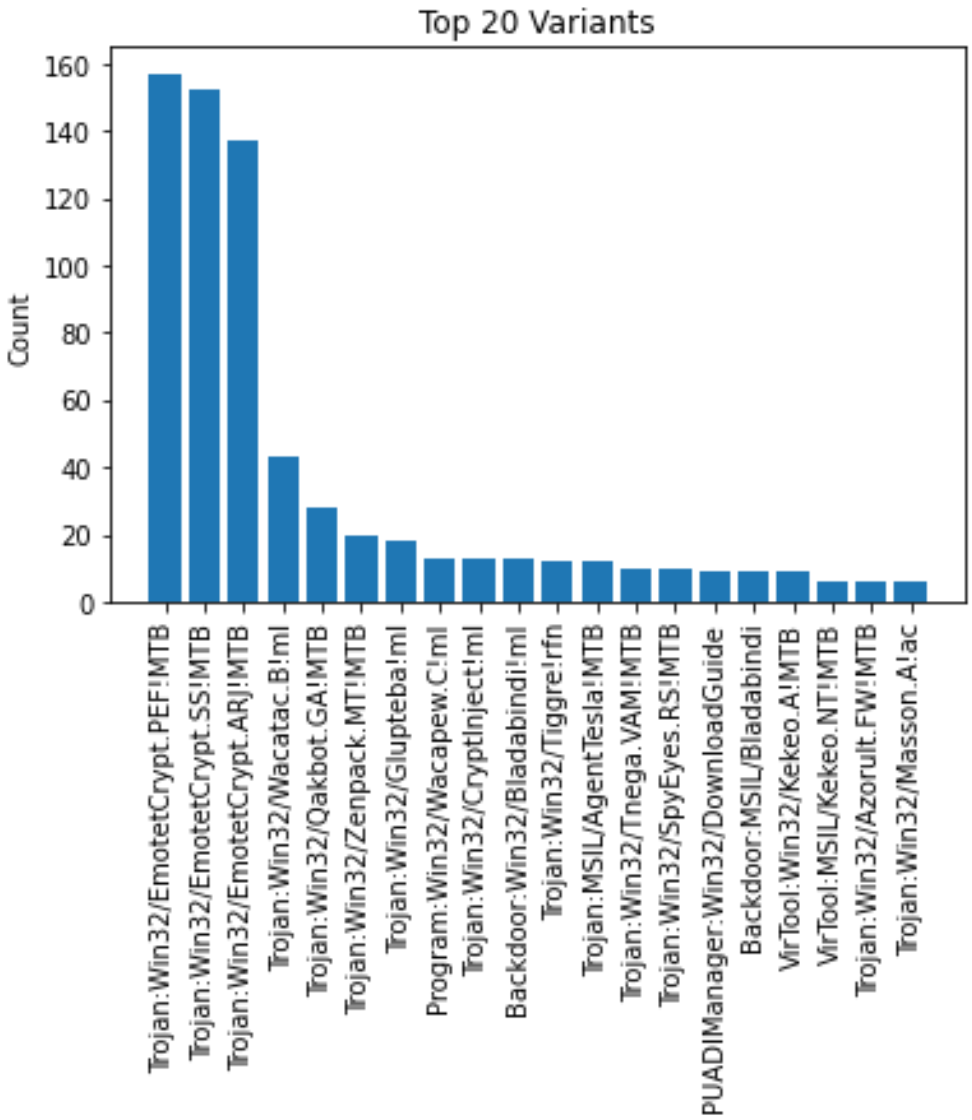

図 13: 本実験で使用した検体の製品 A による検出名称をカウントした結果上位 20 種

次にデジタル証明書による検体への署名の詳細な手順は，以下の通りである.

① Windows 上で PowerShell スクリプトを使用し，自己署名証明書を新規で作成した.
② 別環境の Linux マシン上に同証明書をコピーした.
③ Openssl ベースの Windows バイナリ用の署名ツールである osslsigncode に同証明書を使用して，前記の手法で収集した既知のマルウェアに Authenticode 形式でデジタル署名し，その際に署名者の名前欄や URL 欄に特定の文字列を大量に加えた.
④ 文字列を加えた検体を，各製品がインストールされた Windows 環境にコピーした上で，各ウイルス対策ソフトでスキャンした.

Authenticode は Windows バイナリに対して図 14 のような形式で署名される．本実験では，このオプションで設定可能な署名者のデータ領域(署名者の名称と URL を書き込むことが可能)に対して，上記③の手順において特定の文字列を加えることで回避攻撃を発生させる．

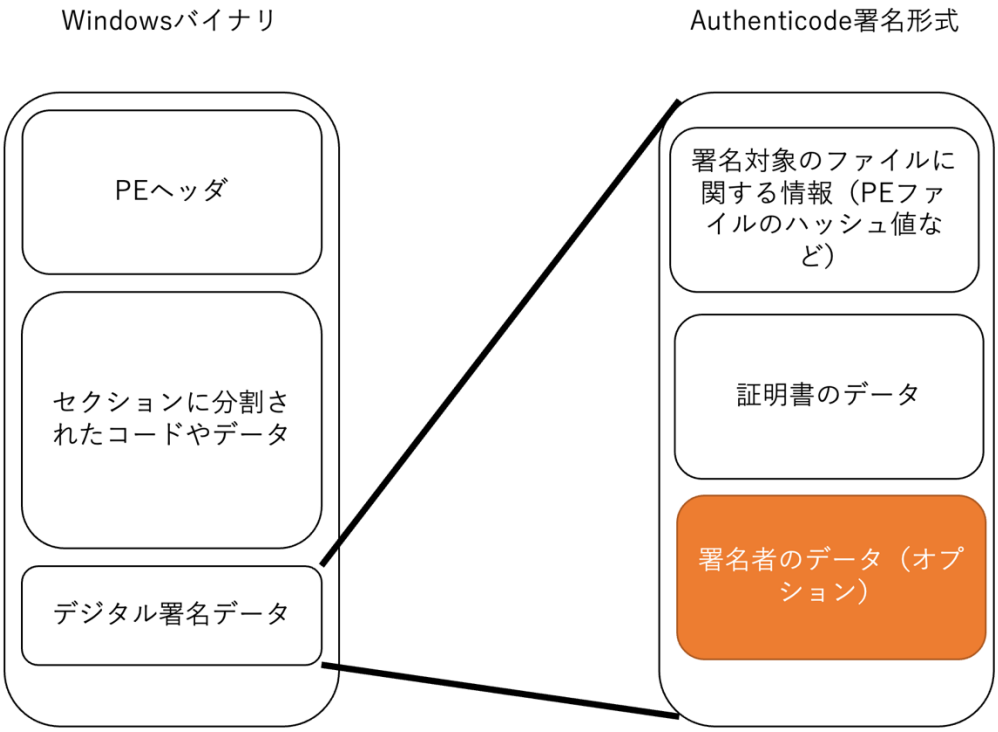

図 14: Authenticode 形式の署名の概要

3.5 ビックテックの企業名を文字列として加えた実験

回避攻撃を実現するために，デジタル署名に含ませる文字列として，いくつかの検討を行った．その際のアイディアのひとつとして，ビッグテック企業名を含ませることが回避攻撃として有効性が高いのではないか，と思料された．なぜならば，こうしたビッグテックは様々なソフトウェアやサービスを広く提供しており，かつ，それらの世界的な市場占有性が高い．このため，機械学習ベースのウイルス対策ソフトが学習を行う際のデータセットに含まれる，正規ファイルにはこうした企業の名称が含まれている可能性が高い．それゆえ，仮に既知のマルウェアがそれらを大量に含む場合には見逃しが生じうる蓋然性があるのではないかと考えた．したがって，こうしたビッグテックの企業名をデジタル署名に大量に含む既知のマルウェアを作成することで，回避攻撃を実現できる可能性があるのではないかと考えた．

そこで本仮説を検証するために，前記の 1,065 件のマルウェア検体ひとつひとつに対して”Apple”を署名者欄に，”apple.com”を URL 欄に合計で 5 キロバイト追加したデータセットを用意した．加えて”Google Microsoft Apple” を署名者欄に，”google.com microsoft.com apple.com”を

URL 欄に合計で 120 キロバイトをデジタル署名を使用して追加したデータセットを作成した．この 2 つのデータセットを製品 X と製品 A を使用してファイルスキャンを実施した．その結果を表 7 に示す．

表 7: ビッグテックの企業名を文字列として指定量を含ませた検体の検出結果

	本手法適用前の検知率	“Apple”を5K追加した検体の検知率	“Google Apple Microsoft” を120K追加した検体の検知率
機械学習エンジン採用製品X	100%	72.4%	61.2%
ハイブリッド型ウイルス対策ソフトA	100%	85.5%	89.6%

験結果を確認する限りでは，Apple と apple.com を合計で 5 キロバイト追加したデータセットで製品 X の検知率は約 28%低下した．”Google Microsoft Apple”を 120 キロバイト追加したデータセットでは製品 X の検知率は約 39%低下した．製品 X の検知率が 39%低下した際，検知できなくなった検体の検出名称上位 20 種を図 15 に，同検体のデータサイズの分布を図 16 にそれぞれ示す．検出できなくなった検体の種類は 59 種あり，9 件は VirusTotal に登録がなかった．また 150 キロバイト以上 250 キロバイト未満の検体に最も効果があり，他のデータサイズにおいても一定の効果が認められる．

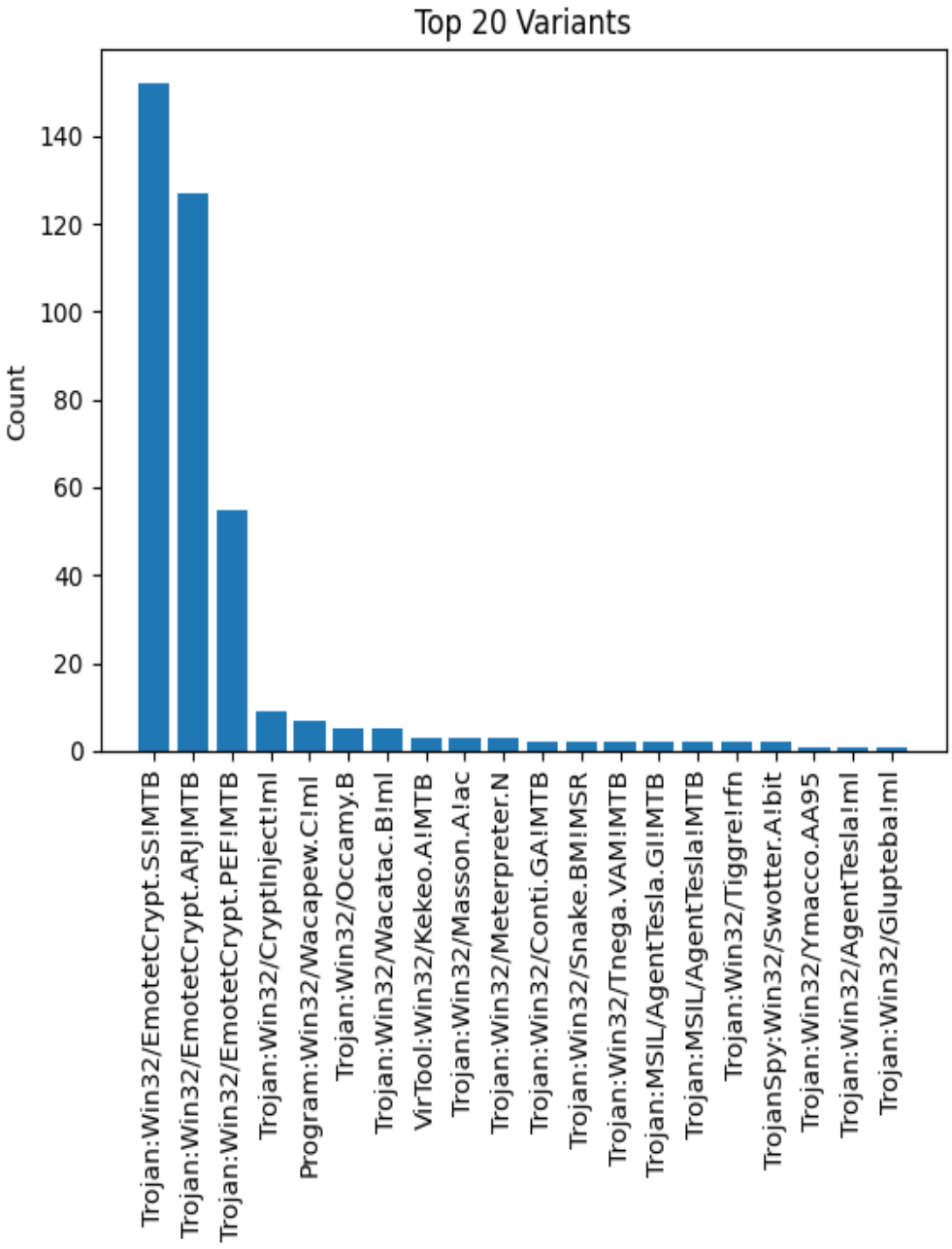

図 15: 製品 X の検知率が約 39%低下した際に見逃した検体の上位 20 種

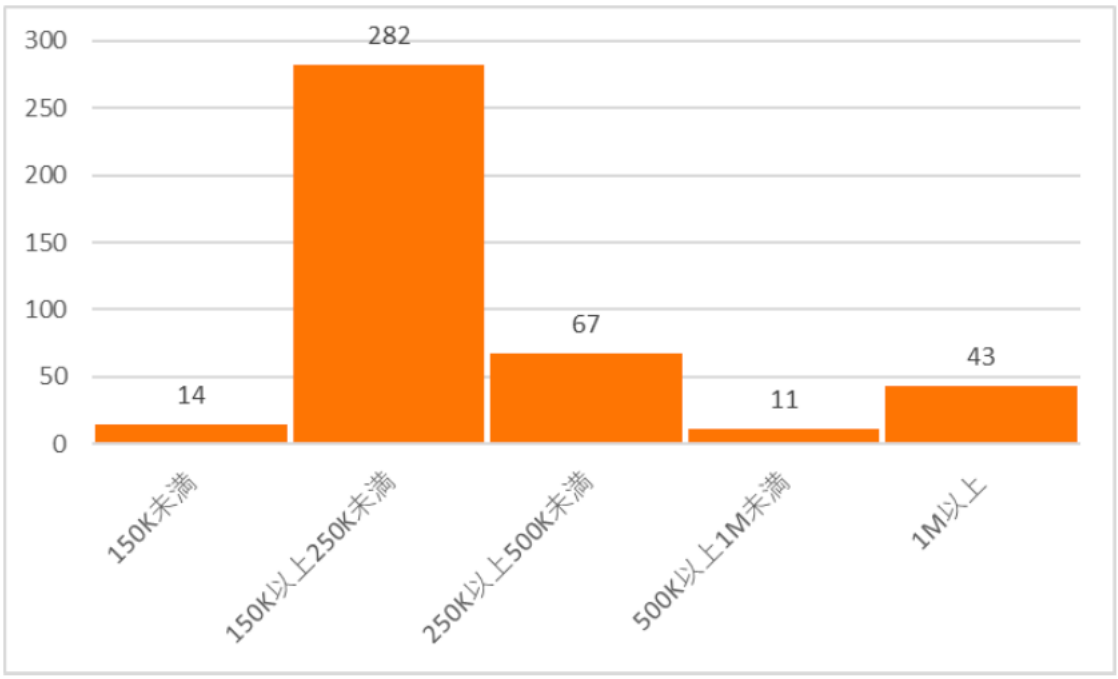

図 16: 製品 X の検知率が 39%低下した際に検知できなくなった検体のデータサイズの分布

一方で製品Aにおいても Apple を 5 キロバイト追加したデータセットで約 14%，”Google Microsoft Apple”を 120 キロバイト追加したデータセットでは約 10%低下した．ハイブリッド型のウイルス対策ソフトにおいても検知率の低下がみられた事由を考察するために，検出できなかった検体の検出名称を調査した．図 7 に Apple を 5 キロバイト追加したデータセットについて，製品 A で検出できなかった検体の名称をカウントしたヒストグラムの上位 20 種を示す．

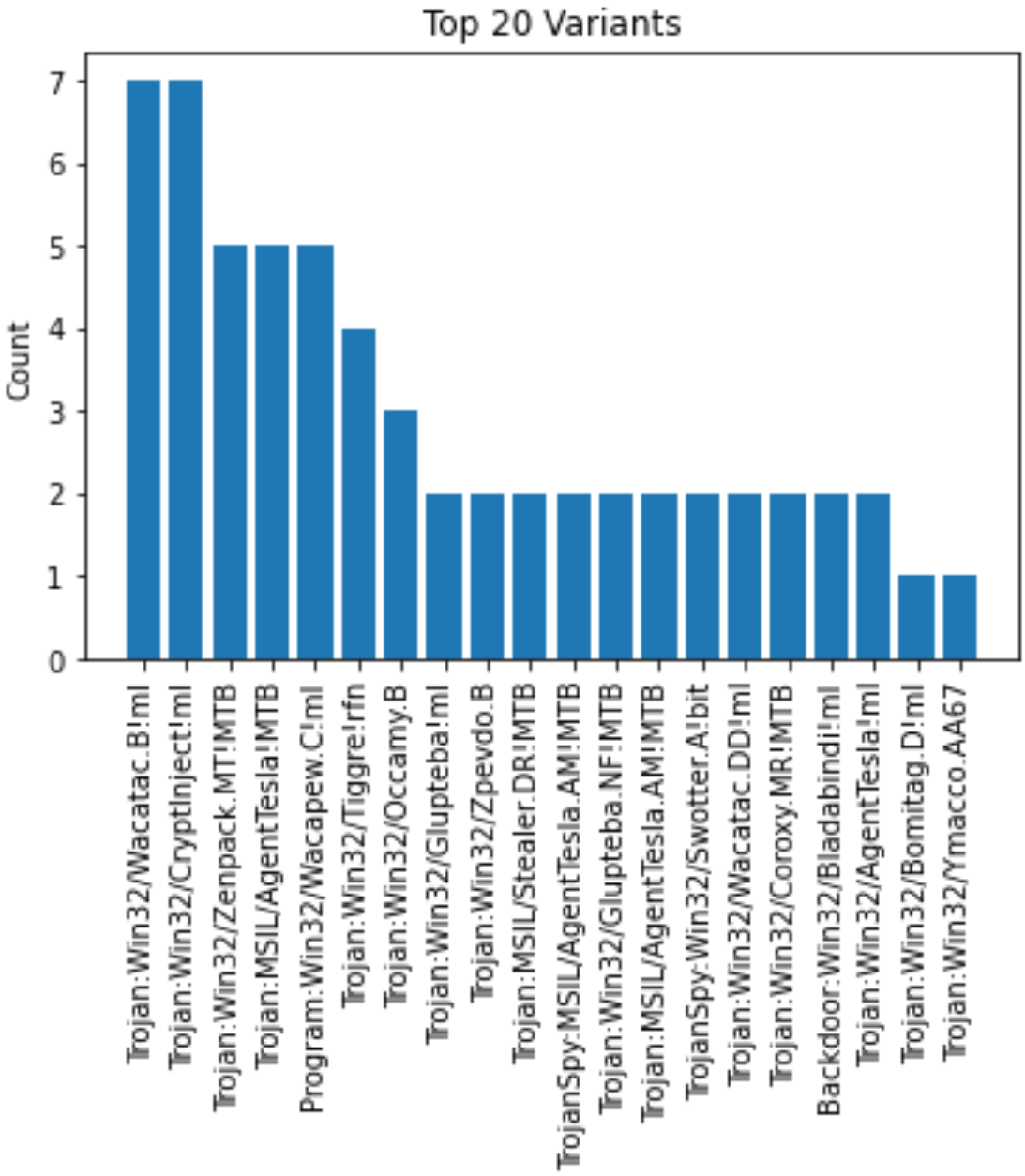

図 17: Apple を 5 キロバイト追加したデータセットで製品 A が検出できなかった検体の上位 20 種

このように，製品 A が検出できなくなった検体の検出名称の接尾辞(suffix)として”!ml”を含むものが上位 20 種中 7 種確認できた．一つの推論として，これは”Machine Learning”による検出を示す接尾辞であり，製品 A は機械学習エンジンも実装してハイブリッド検出に使用していることから，同エンジンに対しても本手法による回避攻撃を成功させている結果が生じているものと考えられる．さらにシンプルな文字列のほうがより検知率の低減に成功している点については，製品 A が実装している機械学習エンジンの使用する特徴量の偏りとして，文字列 Apple ないし apple.com に重みづけが他の 2 つより

高く与えられていることが作用している可能性がある.

3.6 ビデオゲーム開発元の企業名と製品名を文字列として加えた実験

前節の実験結果を踏まえて，さらなる回避率向上のために別のアプローチ手段が必要であるという認識に至り，他の手法について検討を行った．まず，現在のサイバーセキュリティの脅威の情勢を調査したところ，人気の高いゲームタイトルを偽装して拡散し，それらゲームの利用者の個人情報等の重要データを窃取しようとする攻撃が増加傾向にある [41]という報道を確認した．また，こうしたゲーム利用者間で盛んに利用されているチャットツールである Discord を悪用するマルウェアも増加傾向にある [42]といった情勢も確認した．係る情勢をもとにした推論として，マルウェアはゲームプラットフォームである Steam や，ゲーム利用者向けチャットツールである Discord を標的にしているため，同ソフトウェアを偽装している可能性がある．その一方で，こうした企業から正規にリリースされているソフトウェアについては誤ってウイルス対策ソフトが検知しないよう，ウイルス対策ソフトベンダーとしては自らの製品の真陰性を高めていかなければならない．さらに，こうしたソフトウェアのアップデートは頻繁に発生することが確認 [43] [44]されており，またゲームタイトルも数万本に及ぶことから，誤検知のリスクが多発する可能性も確認されている [45]．このようなジレンマが機械学習ベースのウイルス対策ソフトには生じるのではないかと推定し，これらのソフトウェア名称並びに開発元の企業名称を既知のマルウェア検体に含ませる文字列として選択した．前記の 1,065 件のマルウェア検体ひとつひとつに対して”Steam”を署名者欄に，”steam.com”を URL 欄に合計で 5 キロバイト追加したデータセットを用意した．加えて区切り文字としてスペースを使用していたのを省き”SteamDiscordValve”を署名者欄に，”valve.com”を URL 欄に合計 120 キロバイト，デジタル署名を使用して追加したデータセットを作成した．この 2 つのデータセットを製品 X と製品 A，さらに本手法の有効性の影響範囲を追加調査するために製品 B と製品 C を追加し，ファイルスキャンを実施した．製品 B は開発元のサイトに「機械学習も使用するハイブリットタイプに強化している」と述べられていた．また製品 C も同様にハイブリッドタイプであると開発元サイトに記載されていた．また両製品共に市場占有性が世界トップ 5 に入っているため，本手法の効果を計測する対象として適切であると考えた．この結果を表 8 に示す.

このように，製品 X のみならず，すべての製品に対して本手法による検出率の低下が認められ，特に製品 C においては 59.2%の低下が認められた．製品 X の検知率が 44%低下した際に，検知できなくなった検体のデータサイズの分布を図 15 に示す．検出できなくなった検体の種類は 77 種で 3.3 節の実験より多くなり，9 検体は VirusTotal に登録がなかった．また 3.3 節での実験同様に 150 キロバイト以上 250 キロバイト未満の検体に最も効果が認められると共に，他のデータサイズにおいて回避件数が向上したことが認められる.

表 8: 四製品に対して特定の企業名と製品名を文字列として加えた検体をスキャンさせた結果

	本手法適用前の検知率	”Steam”を5K追加	”SteamDiscord Valve” を120K追加
機械学習エンジン採用製品X	100%	72.4%	55.9%
ハイブリッド型ウイルス対策ソフトA	100%	85.5%	89.0%
ハイブリッド型ウイルス対策ソフトB	98.2%	76.7%	77.6%
ハイブリッド型ウイルス対策ソフトC	99.4%	43.0%	40.2%

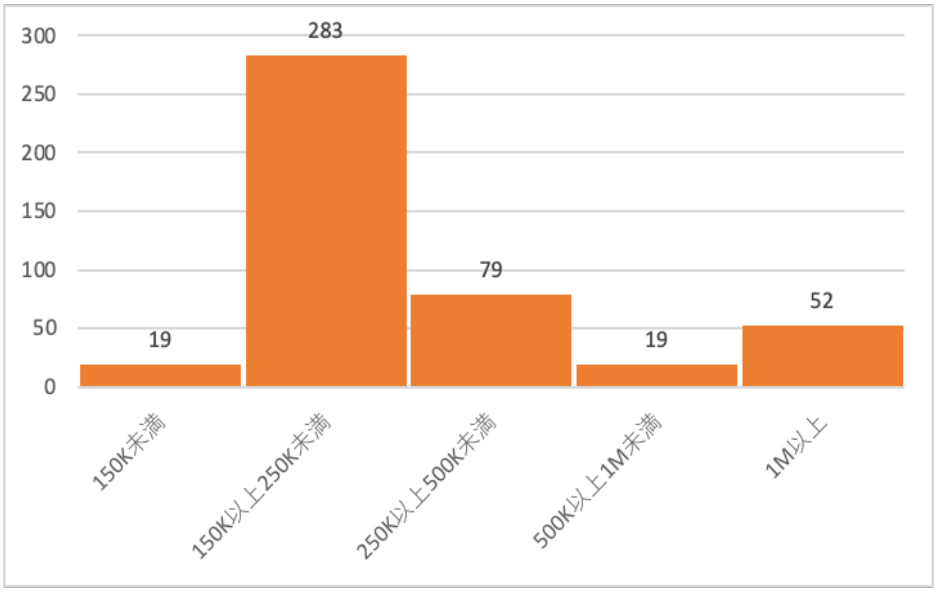

図 18: 製品 X の検知率が 44%低下した際に検知できなくなった検体のデータサイズの分布

次に製品 B と製品 C で検出できなかった検体の検出名称の特定について VirusTotal を使用して試みた．製品 B についてはこの製品固有のヒューリスティック検出名称が付与された検出名称が，検出不能になった検体の 70%に付与されていた．この固有の検出名称を明らかにすることは製品の特定に即つながるため言及を避けるが，ヒューリスティック検知の手段の一つに機械学習エンジンが使用されている

ためにこのような結果になった可能性も考えられる．一方で製品 C について同様の試みを行ったところ，検出不能になった検体のうち 94%の検出名称に”ML”という文字を含む検出名称が付与されていた．これは製品 A と同様に”Machine Learning”による検出を示す証左であると考えられる．そして回避率が大きくなった主要因として，製品 C は製品 A や製品 B よりも機械学習エンジンによる検出に頼るところが大きくなっており，パッケージそのものはハイブリッド型ウイルス対策ソフトの製品名であるが，その実はほぼ NGAV に変更されているためにこのような結果が示されたと考えられる．
さらに製品 C に対して Steam のみを 5 キロバイト追加した際に検出できなかった検体のサイズ分布を図 19 に示す．5 キロバイトと追加した量が比較的少量にもかかわらず，広範な分布に対して回避を成立させている．

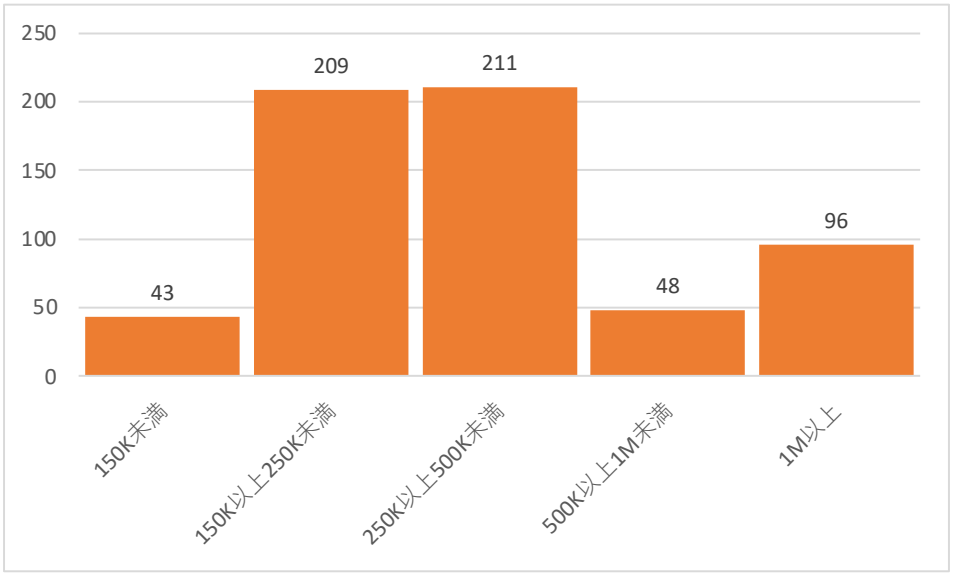

図 19: 製品 C の検知率が 57%低下した際に検知できなくなった検体のデータサイズの分布

3.7 研究倫理的考察

本研究が次世代型ウイルス対策ソフトに対する回避攻撃という重要課題の認知に貢献することを期待し，本論文を公表するものであるが，本研究で検討した回避攻撃が実製品の回避に悪用されることを防ぐため，検証対象の製品ベンダに対して事前に実験結果に関する通知を行った．また，製品名を匿名化し，回避手法についても技術的な主旨を保ちつつ，使用したパラメータ等の詳細な記述を避けた．

3.8 まとめと今後の課題

本研究においては，この NGAV に生じうる回避攻撃を実現するため，既存の検体に対して証明書を使用した文字列の追加という手法を採用した．しかるのち，追加する文字列としてビッグテックの企業名と，ゲーム会社の企業名などを使用した．その上で，回避率についてハイブリッド型のウイルス対策ソフト

と NGAV とを比較することで，1,065 検体を使用して NGAV に最大で約 44%の見逃しを発生させるという結果を確認した．また，パターンファイルと機械学習エンジンの併用をするハイブリッド型のウイルス対策ソフトにおいても 15%から最大で 59%の見逃しを発生させることも確認した．ハイブリッド検出の採用により，未知のマルウェア検知に対する効能が強調される一方で，本提案手法のような回避手法が存在することが認知され，さらなる対策の発展につながることを期待したい．

ただし，環境によっては Windows SmartScreen のようなレピュテーションベースの対策機能が有効に働き，本研究で採用した自己署名の証明書によるコード署名が検知される可能性はあると思料される．その一方，こうしたデータセットに含まれていることが高い文字列をもとにした特徴量が機械学習ベースのウイルス対策ソフトで使用されていることは，先行研究などからも標準的な手法となってきていることが判明している．したがって，他の機械学習ベースのウイルス対策ソフトにおいても同様の手法が回避攻撃として適用可能な蓋然性が高い．さらには，こうした特徴量の推定とノイズの追加による回避攻撃はウイルス対策ソフトのみならず他の機械学習を取り込んだセキュリティ対策にも影響を与える可能性が高いと考える．

第4章 結論

本研究において，サイバーセキュリティの領域における既存の対策だけではなく新しい領域に関しても機械学習の有効性を試すべく，ダークウェブにおける違法物品取引サイトの検知に関する実験を行った．クローリングにより蓄積したデータにアノテーションを行った上で，同データの分析を行い，特徴量を設計してデータセットを開発し，同データセットを学習に使用した分類器を開発した．その結果，再現率が80%を超え，かつ正解率も 85.8%を得られる分類器を開発できた．また同分類器は，隠語の変化の影響を受けにくいという特性を備えることもできた．したがって，サイバーセキュリティの領域における既存の対策だけではなく新しい領域に関しても機械学習の有効性を確認することができた．

その上で，こうした機械学習の社会実装が進んだことが逆に引き起こす盲点や，課題について検討を行い，具体的な事例として，商用の機械学習ベースのウイルス対策ソフトに対する回避攻撃を行った．本研究においては同攻撃を成功させ，その回避率についてハイブリッド型のウイルス対策ソフトと NGAV とを比較することで，1,065 検体を使用して NGAV に最大で約 44%の見逃しを発生させるという結果を確認した．また，パターンファイルと機械学習エンジンの併用をするハイブリッド型のウイルス対策ソフトにおいても 15%から最大で 59%の見逃しを発生させることも確認した．

なお，本研究の手法の詳細，ならびに論文の公表については，実験で使用したウイルス対策ソフト製品の開発元ベンダーに通知済みである．本論文の公表については同開発元ベンダーらから特段のコメントはなかったので，課題の認識としてなされているものと考慮される．本研究の中で明らかになった回避手法が，現実世界のサイバー犯罪者に積極的に採用されている情勢は，幸いにも今のところ確認されていない．それゆえに，サイバー犯罪者より先んじてセキュリティ対策製品等の開発者が本研究の成果を取り込む可能性によってもたらされる社会的利益がもたらされる蓋然性が高いと考える．さらには，今後生成 AI の社会実装が進展するにつれ，こうした生成 AI の課題を特定する際にも本研究が参照され，貢献されていくことを期待している．

本研究において確認できたことのひとつに，サイバーセキュリティ対策としてやはり銀の弾丸は手に入れることはできておらず，単一の手段に頼ってしまうことは間違いの起こる可能性を高めてしまう，ということの再認識が挙げられる．特にウイルス対策ソフトに対する回避攻撃では，従来型のパターンファイル検出とのハイブリッドを謳いつつも，実際は開発元の省力化やコスト削減のために機械学習に頼る余地がどんどん大きくなってきているとの証左ではないかと思料される．一般的な期待としてこうした機械学習への期待の高まりがあるが，機械学習を採用しているがゆえの問題をしっかりと把握し，適切な使用することが本質的なセキュリティ対策に繋がる．

謝辞

本研究を進めるにあたり，ご支援を賜りました全ての皆様に深く感謝致します．

特に，多大なご指導とご助言を頂きました，横浜国立大学大学院環境情報研究院松本勉教授，吉岡克成教授に深く感謝致します．

また，研究活動に際し多大なご援助を頂きました成松美央秘書，石舘知子技術補佐員，高山宏明研究員に深く感謝致します．

さらには本研究を進める上で多大なアドバイスを賜りました，トレンドマイクロ株式会社の東結花氏にも感謝申し上げます．

参考文献一覧

[1] 国立がん研究センター, “AIを活用したリアルタイム内視鏡診断サポートシステム開発 大腸内視鏡検査での見逃し回避を目指す,” [オンライン]. [アクセス日: 3 10 2023].

[2] 一般社団法人行政情報システム研究所, “2020 年 06 月号トピックス 公共分野のデジタル化―AI 技術を活用した水道管路劣化状況のオンライン診断,” [オンライン]. Available: https://www.iais.or.jp/articles/articlesa/20200610/202006_07/. [アクセス日: 3 10 2023].

[3] 情報処理推進機構, “セキュリティ関係者のための AI ハンドブック,” [オンライン]. Available: https://www.ipa.go.jp/jinzai/ics/core_human_resource/final_project/2022/ngi93u0000002jj0-att/000099871.pdf. [アクセス日: 3 10 2023].

[4] Tor Project, “Tor Project,” [オンライン]. Available: https://www.torproject.org/. [アクセス日: 22 11 2019].

[5] Freenet, “Freenet,” [オンライン]. Available: https://freenetproject.org/. [アクセス日: 22 11 2019].

[6] I2P 匿名ネットワーク, “I2P 匿名ネットワーク,” [オンライン]. Available: https://geti2p.net/ja/. [アクセス日: 22 11 2019].

[7] Tor2web, “Tor2web: Browse the Tor Onion Services,” [オンライン]. Available: https://www.tor2web.org/. [アクセス日: 22 11 2019].

[8] Europol, “Operation Onymous,” [オンライン]. Available: https://www.europol.europa.eu/activities-services/europol-in-action/operations/operation-onymous. [アクセス日: 22 11 2019].

[9] 産経新聞, “匿名化ソフト「Ｔｏｒ」使い児童ポルノ公開疑い 京都府警が初摘発,” [オンライン]. Available: https://www.sankei.com/west/news/180605/wst1806050108-n1.html. [アクセス日: 22 11 2019].

[10] ITmedia, “Tor 経由で児童ポルノを公開した疑い 元漫画家を逮捕,” [オンライン]. Available: https://www.itmedia.co.jp/news/articles/1911/15/news040.html. [アクセス日: 22 11 2019].

[11] 現代ビジネス, “経産省 20 代キャリア官僚「覚せい剤密輸」にちらつくダークウェブの影,” [オンライン]. Available: https://gendai.ismedia.jp/articles/-/64579. [アクセス日: 22 11 2019].

[12] R. V. Wegberg, “ Plug and prey? Measuring the commoditization of cybercrime via online anonymous markets, ” 27th USENIX Security Symposium, Baltimore, MD, USA, 2018.

[13] K. Soska, “Measuring the longitudinal evolution of the online anonymous marketplace ecosystem,” 24th USENIX Security Symposium, Washington, DC, 2015.

[14] E. N. e. al., “Darknet and deepnet mining for proactive cybersecurity threat intelligence, ” Intelligence and Security Informatics (ISI) 2016 IEEE Conference, 2016.

[15] D. M. e. al., “Cryptopolitik and the darknet,” Survival, 2006.

[16] S. Ghosh, “ATOL: A Framework for Automated Analysis and Categorization of the Darkweb Ecosystem,” AAAI-17 Workshop on Artificial Intelligence for Cyber Security, 2017.

[17] “Bag of Words（単語の袋）& TF-IDF | Skymind,” [オンライン]. Available: https://skymind.ai/japan/wiki/bagofwords-tf-idf. [アクセス日: 22 11 2019].

[18] AHMIA, “AHMIA,” [オンライン]. Available: https://ahmia.fi/. [アクセス日: 22 11 2019].

[19] JSON の紹介, “ JSON の紹介, ” [オンライン]. Available: https://www.json.org/json-ja.html. [アクセス日: 22 11 2019].

[20] “Requests: 人間のための HTTP,” [オンライン]. Available: https://requests-docs-ja.readthedocs.io/en/latest/. [アクセス日: 22 11 2019].

[21] Sarah Jamie Lewis, “OnionScan Report: Freedom Hosting II, A New Map and a New Direction., ” [オンライン]. Available: https://mascherari.press/onionscan-report-fhii-a-new-map-and-the-futfut/. [アクセス日: 22 11 2019].

[22] F. Onions, “ Fresh Onions, ” [オンライン]. Available: https://github.com/dirtyfilthy/freshonions-torscraper. [アクセス日: 22 11 2019].

[23] Hunchly, “Hunchly,” [オンライン]. Available: https://www.hunch.ly/. [アクセス日: 22 11 2019].

[24] 仮想通貨 watch, “仮想通貨ミキシングサービスの 3 番手，マネーロンダリングの助長により検挙 , ” [オンライン]. Available: https://crypto.watch.impress.co.jp/docs/news/1186927.html. [アクセス日: 22 11 2019].

[25] L. Breiman, “Random forests,” Machine learning, 2001.

[26] J.H.Friedman, “Greedy function approximation: a gradient boosting machine,” Annals of statistics, 2001.

[27] ESET, “NGAV（次世代型アンチウイルス）とは？EDRとの違いと製品比較のポイント,” [オンライン]. Available: https://www.eset.com/jp/topics-business/next-gen-antivirus/. [アクセス日: 30 11 2022].

[28] 三井物産セキュアディレクション, “用語集,” [オンライン]. Available: https://www.mbsd.jp/aisec_portal/term.html#evasion_attack. [アクセス日: 30 11 2022].

[29] N. Papernot, “Practical black-box attacks against machine learning,” 2017 ACM on Asia conference on computer and communications security, 2017.

[30] W. Y. D. Xu, “Automatically evading classifiers,” 2016 network and distributed systems symposium, 2016.

[31] W. Y. T. Hu, “Generating adversarial malware examples for black-box attacks based on GAN,” arXiv, 2017.

[32] S. Z. Adi Ashkenazy, “Cylance, I Kill You!,” [オンライン]. Available: https://skylightcyber.com/2019/07/18/cylance-i-kill-you/. [アクセス日: 30 11 2022].

[33] F. CESCHIN, “Shallow security: On the creation of adversarial variants to evade machine learning-based malware detectors,” 3rd Reversing and Offensive-oriented Trends Symposium, 2019.

[34] Microsoft, “PE 形式,” [オンライン]. Available: https://learn.microsoft.com/ja-jp/windows/win32/debug/pe-format. [アクセス日: 30 11 2022].

[35] キヤノン IT ソリューションズ, “パッカー（Packer）,” [オンライン]. Available: https://eset-info.canon-its.jp/malware_info/term/detail/00084.html. [アクセス日: 30 11 2022].

[36] M. G. Schultz, “Data mining methods for detection of new malicious executables,” 2001 IEEE Symposium on Security and Privacy, 2001.

[37] M. Ahmadi, “Novel feature extraction, selection and fusion for effective malware family classification,” sixth ACM conference on data and application security and privacy, 2016.

[38] R. Islam, “Classification of malware based on integrated static and dynamic features,” Network and Computer Applications, 2013.

[39] H. S. P. R. Anderson, “Ember: an open dataset for training static pe malware

machine learning models,” arXiv, 2018.

[40] VirusShare, “VirusShare,” [オンライン]. Available: https://virusshare.com/. [アクセス日: 30 11 2022].

[41] ITmedia, “概念ドリフト（Concept drift）／データドリフト（Data drift）とは？,” [オンライン]. Available: https://atmarkit.itmedia.co.jp/ait/articles/2202/21/news033.html,. [アクセス日: 30 11 2022].

[42] “VirusTotal,” Google, [オンライン]. Available: https://www.virustotal.com/. [アクセス日: 30 11 2022].

[43] 株式会社カスペルスキー, “人気の高いゲームを装い，認証情報やクレジットカード情報を窃取するマルウェアが増加傾向に，” [オンライン]. Available: https://prtimes.jp/main/html/rd/p/000000319.000011471.html. [アクセス日: 30 11 2022].

[44] Sophos, “ゲーマー向けチャットツール「Discord」を悪用するマルウェアが増加,” [オンライン]. Available: https://news.sophos.com/ja-jp/2021/08/01/malware-increasingly-targets-discord-for-abuse-jp/. [アクセス日: 30 11 2022].

[45] Steam, “最近のアップデート，” [オンライン]. Available: https://store.steampowered.com/updated/all/. [アクセス日: 30 11 2022].

[46] Discord, “Discord Version History,” [オンライン]. Available: https://www.ipa4fun.com/history/101592/. [アクセス日: 30 11 2022].

[47] Steam, “ウイルス対策ソフトウェアから Steam ゲームが有害であると警告されました,” [オンライン]. Available: https://help.steampowered.com/ja/faqs/view/5f3d-1477-aff9-c4f3. [アクセス日: 30 11 2022].

公表論文リスト

学会論文誌論文

1. 新井悠，吉岡克成，松本勉．ダークウェブ内の違法物品取扱サイトの HTTP ヘッダ情報を特徴量にした同サイトの自動検出．情報処理学会論文誌,61(9),1388-1396（2020-09-15），1882-7764
2. 新井悠，吉岡克成，松本勉．次世代型ウイルス対策ソフトとハイブリッド検出を実装するウイルス対策ソフトに対する回避攻撃．情報処理学会論文誌,64(9),1287-1294（2023-09-15），1882-7764

本研究に関連する国際会議発表

1. Yu Arai, Katsunari Yoshioka, and Tsutomu Matsumoto. Evasion attacks against Next-Generation AntiVirus software and antivirus software implementing hybrid detection. AsiaJCIS 2023(Poster Session)

本研究に関連する研究会・シンポジウム等発表（査読なし）

1. 新井悠，吉岡克成，松本勉．ダークウェブ内の違法物品取扱サイトのミドルウェアの特徴に着目した実態調査．コンピュータセキュリティシンポジウム 2019 論文集,2019,482-487 (2019-10-14)
2. 新井悠，吉岡克成，松本勉．ダークウェブの経年的変化に関する考察．コンピュータセキュリティシンポジウム 2020 論文集,44-49 (2020-10-19)

その他

1. 新井悠．DDIR：ダークウェブの研究を目的としたオープンソースデータセット．Code Blue 2019
2. 新井悠，一瀬小夜，黒米祐馬．セキュリティエンジニアのための機械学習．オライリー・ジャパン．2019