情報処理学会論文誌 Vol.61 No.9 1388–1396 (Sep. 2020)

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

新井 悠[1,2,a)] 吉岡 克成[3,4] 松本 勉[3,4]

受付日 2019年11月25日, 採録日 2020年6月1日

概要：近年，様々な違法物品ならびにサービスが，ダークウェブ上に構築された仮想取引所などで取引されている．これを利用することにより誰でもそれらの違法物品を手に入れることが可能になってきている．研究者らがダークウェブをクローリングすることで，こうした違法物品取扱サイトの状況などを確認する試みも行われてきている．他方で，ダークウェブ内の違法取引所の自動検出に焦点を置いた研究は少ない．本研究ではダークウェブ上に構築されているこれらの秘匿サービスのクローリングを行い，データを収集した．そのうえで，隠語の変化などに左右されない，HTTP ヘッダを特徴量にする手法で，かかる違法物品取扱サイトを自動検出する手法を案出した．

キーワード：ダークウェブ，クローリング，Tor，機械学習，自動検出，HTTP ヘッダ

Automatic Illegal-goods-handling Site Detection by Using HTTP Headers

Yu Arai[1,2,a)] Katsunari Yoshioka[3,4] Tsutomu Matsumoto[3,4]

Received: November 25, 2019, Accepted: June 1, 2020

Abstract: In recent years, various illegal goods and services are traded on virtual exchanges built on the dark web. By using this, anyone can get these illegal goods. In addition, researchers have been trying to check the status of illegal goods handling sites by crawling the dark web. On the other hand, few studies focus on the automatic detection of illegal exchanges in the dark web. In this study, we crawled these secret services built on the dark web and collected data. In addition, we devised a method for automatically detecting such illegal article handling sites by using HTTP headers as features, regardless of changes in slang.

Keywords: dark web, crawling, Tor, machine learning, automatic detection, HTTP header

1. はじめに

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 [1]，Freenet [2]，I2P [3] 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，そ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 [4]．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，アクセスするごと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

1 株式会社 NTT データ
NTT DATA Corporation, Koto, Tokyo 135–8671, Japan
2 横浜国立大学大学院環境情報学府
Graduate School of Environment and Information Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
3 横浜国立大学先端科学高等研究院
Institute of Advanced Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
4 横浜国立大学大学院環境情報研究院
Faculty of Environment and Information Sciences, Yokohama National University, Yokohama, Kanagawa 240–8501, Japan
a) yuu.arai@gmail.com

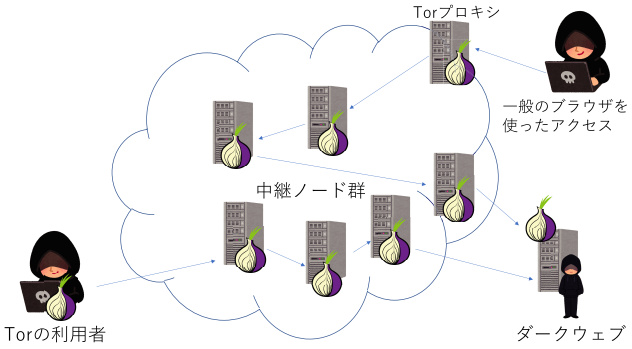

図 1 ダークウェブの模式図
Fig. 1 Schematic diagram of the dark web.

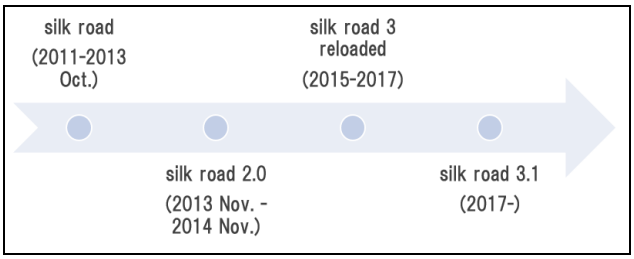

図 2 silk road の時系列整理
Fig. 2 Timeline of the silk road.

みを持っている．

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なくても，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，仮想資産など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 [5]．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 [6]，2019 年 11 月にも同様の容疑で別の被疑者を検挙している [7]．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 [8]．

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**）．

「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あげていた．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，いったんその運営者など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，Van Wegberg ら [9] 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 カ月であった．また，Soska ら [10] 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，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肝要であると思料される．

本研究では，かかる監視方法とその評価を達成するために，次のような手順を用いて，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した．

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 ページなどから収集し，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．

本論文では，上述の手順により，まず初期巡回先リストとして 5,763 サイト分の .onion ドメインを持つ URL を収集した．次に，クローラを使用して初期巡回先リストの URL を巡回し，うち 4,340 サイトから HTTP レスポンスデータを得ることができた．

結果の得られた 4,340 サイトの HTML データを確認し，同時に，可能であれば当該サイトに実際に接続して目視による確認を行う，アノテーションを行った．アノテーションを行ううえで，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．

- 違法薬物の取扱
- サイバー攻撃サービスの提供
- 重火器の販売
- 偽造クレジットカードや偽造身分証明書の販売

- 暗号資産ミキサ [11]
- 詐欺サイト
- 児童ポルノ
- 著作権侵害コンテンツの取扱
- その他，犯罪活動に結び付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などの販売サイトなども含まれる．

本論文の実態調査および研究により，以下のような実態と成果が明らかになった．

- クローリング結果の 41%にあたる 1,799 サイトが違法物品取扱サイトであった．
- データの収集結果の HTTP ヘッダの傾向として，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，違法物品取扱サイトが当該ヘッダをともなう割合は 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．
- 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ともなう割合は 14.29%であった．
- X-powered-by ヘッダをともな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ともなう割合は 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．
- 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．
- 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．
- こうした HTTP ヘッダには傾向があると仮定し，ヘッダの出現傾向から，HTTP ヘッダの有無を特徴に変換し，分類を行った．ランダム木と LightGBM を使用した分類器の正解率が 78%から 82%と良い結果を示したが，再現率が 67%程度と，見逃しが多いモデルとなった．
- 特徴量を見直し，HTTP ヘッダの値の長さと，HTTP ヘッダの行数を特徴量とし，前記と同じ条件で実験を行った．すると，ランダム木による分類において，正解率 85%，再現率も 80%を超える，見逃しのより少ないモデルを開発することに成功した．

2. 関連研究

2.1 ダークウェブに対するクローリング

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 [12] や，収集した HTML データの分類を目的としたもの [13] 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 ヘッダの特徴などについては示されていない．

2.2 ダークウェブの分類

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 ら [14] は，Bag-of-Words [15] 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．

3. 調査手法

3.1 クローリングのための初期巡回先リストの作成

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 [16]，およびダークウェブ内に設けられた Wiki ページなどから収集し，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 5,763 サイト分の.onion ドメインを持つ URL を収集した．そのうえで，クローラに初期巡回先リストを使用して，クローリングを実施した．

3.2 クローリングの実施

先の手順で作成した初期巡回先リストを使用し，2019 年 6

```
{
  "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
Fig. 3 Example JSON output of crawling results.

表 1 JSON 要素の詳細
Table 1 Details of the JSON elements.

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

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 [17] 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 [18] を使用し，HTTP リクエストを Tor Browser Bundle の User-Agent を設定して送信した．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** のとおりである．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．

クローリングを実施した結果，4,340 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，Lewis の報告 [19] 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 4,400 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 [20] 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 1,490 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 [21] 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 4,584 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十分であるという蓋然性は高いと考えられる．

3.3 アノテーションの実施

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 データをサイトごと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 4,340 サイトのうち，41%にあたる 1,799 サイトが違法物品取扱サイトと確認できた．

なお，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．

- 個人のブログサイト
- 反政府的な主張を掲載しているサイト
- 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
- 自作の詩や小説，パロディなどの紹介
- ダークウェブのリンクサイト
- 一般的な掲示板サイト
- 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

表 2 代表的な HTTP レスポンスヘッダの例
Table 2 An example of a typical HTTP response header.

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

3.4 調査結果

3.4.1 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．本研究においては，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 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．このため，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．

3.4.2 HTTP レスポンスヘッダの傾向

まず全体的な傾向を把握するため，クローリングの結果

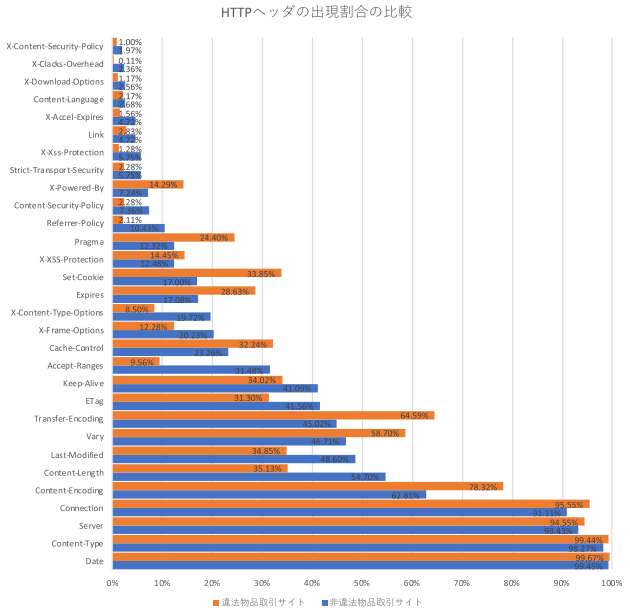

図 4 非違法物品取扱サイトでの HTTP ヘッダの出現割合のうち，上位 30 種類を対象に違法物品取扱サイトのものと比較した結果

Fig. 4 Results of comparison between the top 30 HTTP headers in the non-illegal sites and in the illegal sites.

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．

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ともなう割合は 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ともなう割合は 14.29%であった．当該のヘッダをともな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ともなう割合は 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．このような差がなぜ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．

3.5 運用を考慮した目的の検討

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．

- 実務的な有用性を考慮し，見逃しの少ないモデルの構築
- 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，サイバー犯罪など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．

3.6 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．

3.6.1 ランダム木

ランダム木は Breiman によって 2001 年に提案された [22] 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．

3.6.2 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 [23] の手法の 1 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，そのまとめた単位ごとに学習を行うことで計算量を減らすことができる．

3.7 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．

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．

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

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

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

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．

3.8 HTTP ヘッダを特徴量にしたモデルの案出

前記の 2 つの目的を達成するために，HTTP ヘッダを特徴量に使うことにした．この方法であれば，隠語の変化に左右されない．実際に特徴量に使用した HTTP ヘッダを**リスト 1** に示す．

3.8.1 HTTP ヘッダの出力有無を特徴量に使用した場合

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うえで，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．

適合率と再現率の差があり，適合率のほうが高いという

表 3 分類結果の正誤評価
Table 3 Evaluation of correctness of classification results.

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

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

リスト 1 特徴量に使用した HTTP ヘッダ一覧
List 1 List of HTTP headers used for the features.

	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 5 HTTP ヘッダの有無を特徴量に変換した例
Fig. 5 An example of converting the presence of HTTP headers to features.

表 4 HTTP ヘッダの出力有無を特徴量に使用した実験結果
Table 4 Experimental results using the presence or absence of HTTP header output as a feature.

	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%

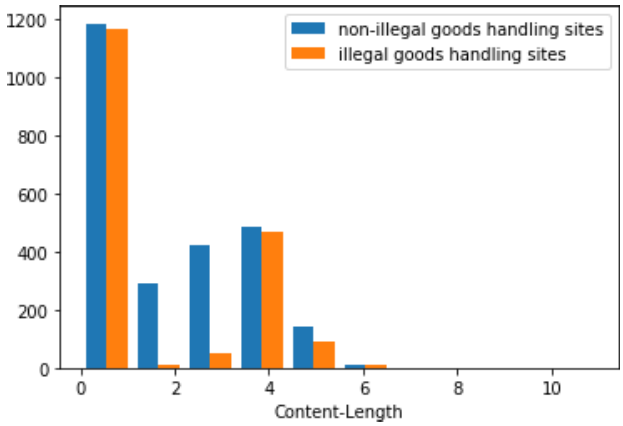

図 6 Content-Length ヘッダの値の長さのヒストグラム
Fig. 6 Histogram of the length of the Content-Length header value.

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．

3.8.2 HTTP ヘッダの値の長さを特徴量に使用した場合

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分かった．**図 6** 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．

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分かった．**図 7** 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 8** に示す．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．

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かかわらず，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 9** 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．

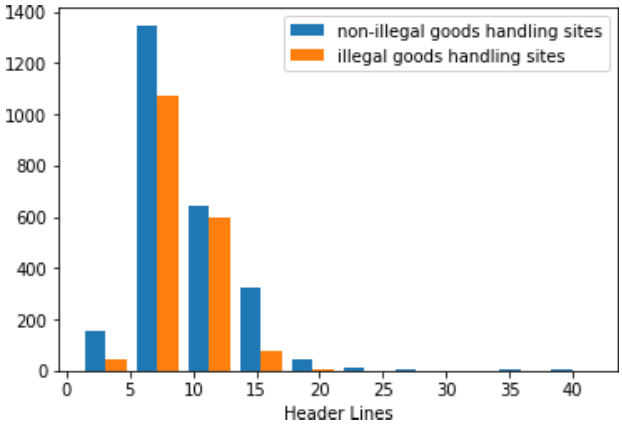

図 7 HTTP ヘッダの行数のヒストグラム
Fig. 7 Histogram of the number of lines in the HTTP header.

	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 8 HTTP ヘッダの値の長さと行数を特徴量に変換した例
Fig. 8 An example of converting the length and number of lines of an HTTP header value to features.

表 5 HTTP ヘッダの値の長さと，HTTP ヘッダの行数を特徴量とした実験結果
Table 5 Experimental results using the length of the HTTP header and the number of lines of the HTTP header as features.

	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%

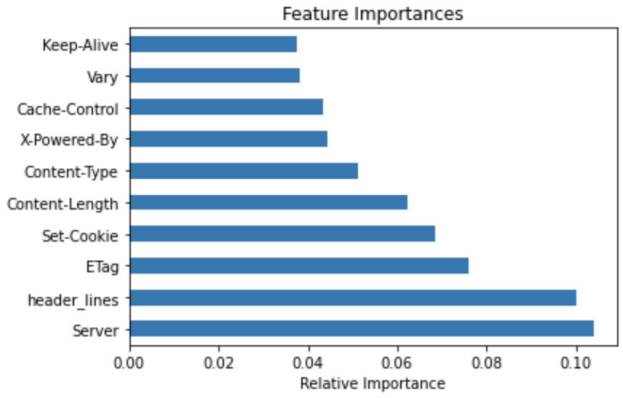

図 9 特徴量の重要度の上位 10 件のヒストグラム
Fig. 9 Histogram of the top 10 most important features.

4. 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うえで，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．一方で，今回の研究はダークウェブの HTTP レスポンスヘッダのみに着目した研究であり，クローリング結果の HTML データの調査分析には着手していない．今後はこの HTML データにも調査をすすめ，社会問題化しているダークウェブに対する，より効果的な対策となりうるような手段の案出について研究を続けたい．

なお，本研究成果の一部は，国立研究開発法人情報通信研究機構（NICT）の委託研究「Web 媒介型攻撃対策技術の実用化に向けた研究開発」の支援により得られた．

参考文献

[1] Tor Project, available from ⟨https://www.torproject.org/⟩ (accessed 2019-11-22).
[2] Freenet, available from ⟨https://freenetproject.org/⟩ (accessed 2019-11-22).
[3] I2P 匿名ネットワーク，入手先 ⟨https://geti2p.net/ja/⟩（参照 2019-11-22）.
[4] Tor2web: Browse the Tor Onion Services, available from ⟨https://www.tor2web.org/⟩ (accessed 2019-11-21).
[5] Operation Onymous | Europol, available from ⟨https://www.europol.europa.eu/activities-services/europol-in-action/operations/operation-onymous⟩ (accessed 2019-11-18).
[6] 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，入手先 ⟨https://www.sankei.com/west/news/180605/wst1806050108-n1.html⟩（参照 2019-11-22）.
[7] ITmedia，Tor 経由で児童ポルノを公開した疑い 元漫画家を逮捕，入手先 ⟨https://www.itmedia.co.jp/news/articles/1911/15/news040.html⟩（参照 2019-11-22）.
[8] 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，入手先 ⟨https://gendai.ismedia.jp/articles/-/64579⟩（参照 2019-11-22）.
[9] Van Wegberg, R., Tajalizadehkhoob, S., Soska, K., Akyazi, U., Ganan, C.H., Klievink, B., Christin, N. and Van Eeten, M.: Plug and prey? Measuring the commoditization of cybercrime via online anonymous markets, *27th USENIX Security Symposium*, Baltimore, MD, USA: USENIX Association, pp.1009–1026 (2018).
[10] Soska, K. and Christin, N.: Measuring the longitudinal evolution of the online anonymous marketplace ecosystem, Proc. 24th USENIX Security Symposium (USENIX Security'15), Washington, DC, pp.33–48 (Aug. 2015).
[11] 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，入手先 ⟨https://crypto.watch.impress.co.jp/docs/news/1186927.html⟩（参照 2019-11-22）.
[12] Nunes, E. et al.: Darknet and deepnet mining for proactive cybersecurity threat intelligence, *IEEE Conference on Intelligence and Security Informatics* (*ISI 2016*), pp.7–12, IEEE (2016).
[13] Moore, D. et al.: Cryptopolitik and the darknet, *Survival*, Vol.58, No.1, pp.7–38 (2006).
[14] Ghosh, S. et al.: ATOL: A Framework for Automated Analysis and Categorization of the Darkweb Ecosystem, *Proc. AAAI-17 Workshop on Artificial Intelligence for Cyber Security*, San Fransisco, USA (Feb. 2017).
[15] Bag of Words（単語の袋）& TF-IDF | Skymind，入手先 ⟨https://skymind.ai/japan/wiki/bagofwords-tf-idf⟩（参照 2019-11-22）.
[16] AHMIA, available from ⟨https://ahmia.fi/⟩ (accessed 2019-11-22).
[17] JSON の紹介，入手先 ⟨https://www.json.org/json-ja.html⟩.
[18] Requests：人間のための HTTP，入手先 ⟨https://requests-docs-ja.readthedocs.io/en/latest/⟩（参照 2019-11-22）.
[19] OnionScan Report: Freedom Hosting II, A New Map and a New Direction, available from ⟨https://mascherari.press/onionscan-report-fhii-a-new-map-and-the-futfut/⟩ (accessed 2019-11-22).
[20] Fresh Onions, available from ⟨https://github.com/dirtyfilthy/freshonions-torscraper⟩.
[21] Hunchly, available from ⟨https://www.hunch.ly/⟩ (accessed 2019-11-22).
[22] Breiman, L.: Random forests, *Machine learning*, Vol.45, No.1, pp.5–32 (2001).
[23] Friedman, J.H.: Greedy function approximation: A gradient boosting machine, *Annals of statistics*, pp.1189–1232 (2001).

新井 悠

2019 年 4 月横浜国立大学大学院環境情報学府博士後期課程に進学．株式会社 NTT データにて CSIRT 業務に従事．

吉岡 克成 （正会員）

2005 年 3 月横浜国立大学大学院環境情報学府情報メディア環境学専攻博士課程後期修了，博士（工学）．同年 4 月情報通信研究機構研究員．2007 年 12 月横浜国立大学学際プロジェクト研究センター特任教員．2011 年 4 月同大学院環境情報研究院准教授．マルウェア解析やネットワーク攻撃観測・検知等のネットワークセキュリティの研究に従事．

松本 勉

横浜国立大学大学院環境情報研究院教授．1986 年東京大学大学院博士課程修了，同年より横浜国立大学勤務．2001 より同大学・環境情報研究院教授，同大学先端科学高等研究院情報・物理セキュリティ研究ユニット主任研究者．2018 年 11 月から産業技術総合研究所サイバーフィジカルセキュリティ研究センター長．