# 辞書に基づくDGAボットにより生成された悪性ドメインの判別

佐藤 彰洋[1,a)]　福田 豊[1,b)]　和田 数字郎[1,c)]　中村 豊[1,d)]

**概要**：高度な DGA ボットの出現により，コンピュータネットワークは深刻な脅威に直面している．この DGA ボットは，自身の辞書から単語を連結することで，人為的に生成したものと判別が困難な悪性ドメインを機械的に生成する．本稿では，ドメインの文字列を構成する単語間の関係性を考慮することで，辞書に基づく DGA ボットにより生成された悪性ドメインの判別を試みる．また実験を通じて，提案手法が 0.9977 の再現率と 0.9869 の適合率で悪性ドメインを判別可能であることを確認した．この結果から，ネットワークに内在する多様なボットへの迅速な対処が可能となるため，ネットワークの運用において安全性の向上が期待できる．

**キーワード**：DGA ボット，辞書に基づくドメイン生成，ネットワークセキュリティ，機械学習

# An Approach for Identifying Malicious Domain Names Generated by Dictionary-based DGA bots

AKIHIRO SATOH[1,a)]　YUTAKA FUKUDA[1,b)]　SUJIRO WADA[1,c)]　YUTAKA NAKAMURA[1,d)]

***Abstract:*** Computer networks are facing serious threats by the emergence of sophisticated new DGA bots. These DGA bots have their own dictionary, from which they concatenate words to dynamically generate malicious domain names that are difficult to distinguish from human-generated domain names. In this paper, we propose an approach for identifying malicious domain names based on relations among the words that constitute the character string of each domain name. Our evaluation demonstrates that this approach has high identification ability, with a recall of 0.9977 and a precision of 0.9869. By enabling one to swiftly address various bots, our approach contributes to dramatically improving network security.

***Keywords:*** DGA bot, dictionary-based domain generation, network security, machine learning.

## 1. はじめに

ボットネットはインターネットにおける深刻な脅威のひとつである．ボットネットとは，所有者の意図に反して遠隔操作が可能な機器であるボットと，それら多数のボットに対して一括して命令を発行する C&C (Command-and-Control Server) で構成される．サイバー犯罪者は，C&C を介してボット群を操作することにより，マルウェアの拡散，クリック詐欺，サービス妨害攻撃などの悪意ある活動を試みる．ボットネットの規模は時として数十万台にも達するため，それにより生じる被害も甚大なものとなる [1].

ボットネットによる被害の抑止のため，管理者には自身のネットワークに内在するボットに対して迅速に対処することが求められる．一方，多くのボットには，検出を回避するための機能として DGA (Domain Generation Algorithm) が実装されている [2]．DGA とは，C&C のドメインを頻繁に変更することで，ボットから C&C へ向けた通信であ

[1] 九州工業大学，〒 804–8550 北九州市戸畑区仙水町 1–1
Kyushu Institute of Technology, 1–1 Sensuicho, Tobata, Kitakyushu, 804–8550, Japan
[a)] satoh@isc.kyutech.ac.jp
[b)] fukuda@isc.kyutech.ac.jp
[c)] swada@isc.kyutech.ac.jp
[d)] yutaka-n@isc.kyutech.ac.jp

ⓒ 2019 Information Processing Society of Japan

るコールバックを隠蔽するための仕組みである．具体的には，ボットは DGA に基づいて機械的にドメインを生成し，それらドメインに対して名前解決を試みる．その名前解決の結果，正しい応答を返したドメインを C&C と見做す．

幾つかの研究では，良性と悪性のドメイン文字列の差異から DGA ボットのコールバック通信を検出している [3], [4], [5]．これは，登録済みのドメインとの衝突を避けるため，悪性ドメインが無意味な文字列から成ることに起因する．それに対して，これまでの検出を無効化する高度な DGA ボットが出現している．この DGA ボットは，自身の辞書から単語を連結することで，人為的に生成したものと判別が困難なドメインを機械的に生成する．

本稿では，辞書に基づく DGA ボットの検出のため，DNS に対する膨大な数の名前解決要求から機械的に生成された悪性ドメインの判別を試みる．DNS に着目した理由は，ボットによる通信に先んじて必ず名前解決が生じること，暗号化による通信内容の隠蔽が困難であることに起因する．本稿の構成は次の通りである．先ず，2 章で既存研究とその問題点を整理する．3 章でドメイン文字列を構成する単語の関係性に基づく悪性ドメイン判別手法を提案した後，4 章で提案手法の有効性を議論する．最後に 5 章で本研究の貢献と課題を纏める．

## 2. 関連技術

本章では，DGA ボットを中心とした関連技術について述べる．2.1 節で DGA ボットの詳細について説明した後，2.2 節で既存研究とその問題点を整理する．

### 2.1 DGA ボット

図 1 に DGA ボットによるコールバック通信の概要を示す．ここで，図中の Q で示す通信は再帰 DNS に対する名前解決を，R はその応答を意味する．先ず，ボットは DGA に基づいて機械的に複数のドメインを生成し，それらドメインを自身の属するネットワーク内の再帰 DNS に問い合わせる．再帰 DNS は，ドメインが登録済みであった場合，そのドメインに対応付けられたアドレスを，ドメインが未登録であった場合，エラーメッセージとして NXDOMAIN を応答する．最終的に，ボットは正しい応答があったドメインを C&C と見做し，そのドメインに対してコールバック通信を試みる．

DGA の目的は，ボットと C&C の間に可用性の高い通信経路を確立することにある．具体的には，C&C のドメインを変更することで，ブラックリストに基づく通信の遮断を容易に回避することが可能となる．加えて，ネットワーク内から外へ向けた通信は宛先が多岐に渡るためコールバックの発見が困難となること，アドレス変換やファイアウォールにより通信を制限されないことが挙げられる．

Conficker や GameOver Zeus, Torpig など，これまでに世界

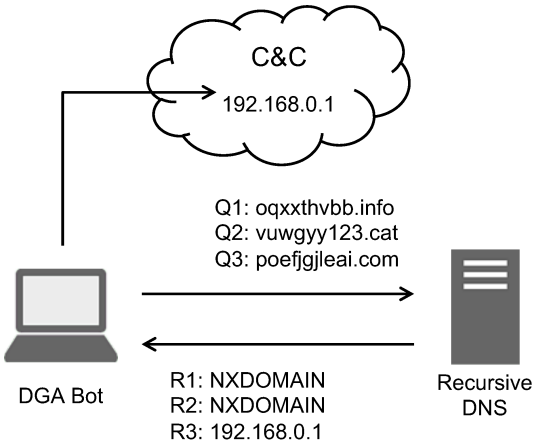



**図 1** DGA ボットによるコールバック通信

で深刻な被害を齎したボットは, その機能の一部として DGA が実装されている．GameOver Zeus は [6]，登録済みのドメインとの衝突を避けるため, 15 から 30 文字のランダムな英数字から成る文字列を悪性ドメインとして生成する．そのドメインの例は，`1ygx14u1vnf8hb1twhv8619h8ygr.net` や `cipu0wdgsnq9u8st8m1lym0hq.com` であり，良性ドメインの文字列との明確な差異が見て取れる．一方，辞書に基づく DGA ボットは，自身の辞書から単語を連結することで，人為的に生成したものと判別が困難なドメインを機械的に生成する．

これまでに，ボットネットを容易に構築できるツールキットの公開や [7]，サイバー犯罪者に向けたサービスとしてのボットネットの提供などが確認されている [8]．また，米国 Symantec 社によると，1 万台のボットが 15 ドルほどの価格で取引されているとの報告がある [9]．故に，大規模なボットネットを利用したサイバー犯罪は増加する傾向にあるのが現状である．

### 2.2 既存研究と問題点

ブラックリストの高度化については頻繁な研究が行われており，現在もネットワークにおける脅威防御戦略の中核を成している．Soldo らは，複数の参加者から提供される過去の攻撃ログに基づいて，新たにブラックリストを生成する Collaborative Blacklisting を提案した [10]．また，Freudiger らは，P2P の技術を応用することで，機密性を担保した攻撃ログの共有を実現している [11]．それに対して，DGA ボットは，C&C のドメインを頻繁に変更することにより，ブラックリストに基づく通信の遮断を回避する機能を有している．

Rahbarinia らは，名前解決要求において既知の悪性ドメインと高確率で共起するドメインから未知の悪性ドメインを発見する Segugio を開発した [12]．Segugio は次の直感的知見，(1) 同じマルウェアファミリに感染した機器は，同じ悪性ドメイン群と通信する傾向にあること，(2) 未感染

の機器は，悪性ドメインと通信することがないことに基づいている．DGA ボットにおいては，(1) コールバック通信に生存時間が極端に短い一時的な悪性ドメインを用いること，(2) 一時的な悪性ドメインと共起するドメインは存在し得ないことが挙げられる．故に，このシステムは DGA ボットの通信に対して効果を成し得ない．

Bilge らは，DNS トラフィックから受動的に計測可能な特徴量と機械学習を用いてドメインを評価する Exposure を開発した [13]．特徴量の例は，ドメインの有効期間，ドメインに割り当てられたアドレスの数，ドメインの TTL，ドメインの文字長などである．このシステムが採用する教師付き機械学習において，その精度は一般的に学習用データセットの数と質に依存する．しかしながら，DGA ボットにおける多くの悪性ドメインは NXDOMAIN 応答を返すため特徴量が得られないこと，C&C に対応付けられた悪性ドメインは生存時間が極端に短いことから，十分な量の学習用データセットを確保することが困難である．

これまでに，ボットが生成するドメインの解析結果や [14], [15]，その解析を自動化する仕組みなどが報告されている [16], [17]．その結果を踏まえ，幾つかの研究では文字列の特徴のみを用いたドメインの判別が試みられている．Truong らは，ドメイン文字列のみから良性・悪性を判別する手法を提案した [4]．この手法は，教師付き機械学習とバイグラムモデリングによりドメインにおける頻出文字パターンを学習する．Anderson らは，深層学習を用いた文字レベルのモデリングにより，その手法を拡張した [5]．これらの手法は，悪性ドメインを生成するためのルールに識別可能な偏りが存在することに基づいている．しかしながら，単語を考慮しない文字レベルのモデリングでは，辞書に基づく DGA ボットの検出において十分な精度が期待できない．Pereira らは，ドメイン文字列のみから辞書に基づく DGA ボットを検出する手法を提案した [18]．この手法は，ドメインの生成に用いる辞書をボットのコールバック先から推定することに主眼を置いている．一方，良性と悪性の判別は非常に単純で，ドメイン文字列を成す単語が辞書に閾値以上含まれるか否かに基づいている．故に，多岐に渡る DGA に対して精度を維持するためには，その判別の仕組みの高度化が求められる．

**表 1** 辞書に基づく DGA ボットにより生成された悪性ドメインの例

| | |
|---|---|
| Banjori | earnestnessbiophysicalohax.com |
| | pbmnestnessbiophysicalohax.com |
| Pizd | actionwelcome.net |
| | brokenforget.net |
| Rovnix | toourgovernmentscorrespondence.com |
| | ofhistoryandwithoutindependent.com |
| Suppobox | windowtherefore.net |
| | severadifference.net |

## 3. 提案

本稿では，辞書に基づく DGA ボットの検出のため，DNS に対する膨大な数の名前解決要求から機械的に生成された悪性ドメインの判別を試みる．表 1 に，辞書に基づく DGA ボットとその悪性ドメインの例を示す．機械的に生成したドメインは特定の辞書の単語から構成されるため，人為的に生成したドメインと比較して使用される単語に大きな偏りが生じると予想される．そのドメイン文字列で頻出する単語や共起する単語に明確な差異が現れるという仮定を踏まえ，我々はドメインの文字列を構成する単語の関係性に基づく悪性ドメイン判別手法を提案する．本手法の特徴は，(1) 文字列のみからドメインの良性と悪性を判別すること，(2) ドメイン文字列を成す単語群の関係を一般的なグラフ理論における重み付き無向グラフで表現すること，(3) 中心性によりグラフにおける各頂点の重要性，すなわち各単語の重要性を測ることである．最終的に，その指標に基づく特徴ベクトルに機械学習アルゴリズムを適用することで，ドメインの良性と悪性を判別する．

文献 [19] では，DGA によるコールバック先の変更に伴い NXDOMAIN が発生することを利用して，良性・悪性を判別するドメインの数を大きく絞り込んでいる．ここで，NXDOMAIN はドメインが存在せず名前解決に失敗したことを意味する．その知見を踏まえ，提案手法では DNS に対する名前解決の結果が NXDOMAIN であったドメインのみに着目する．また，その名前解決による通信は発生しないため，提案手法による良性と悪性の判別には厳格な実時間性を要求されないことを留意されたい．

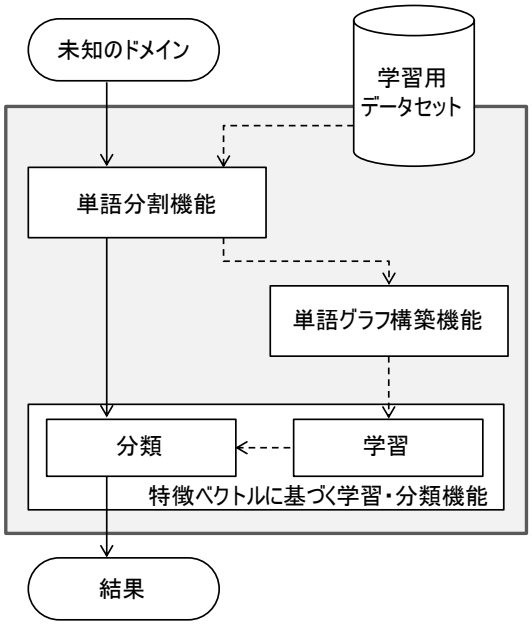



**図 2** ドメインの文字列を構成する単語間の関係性に基づく悪性ドメイン判別手法の概要

ⓒ 2019 Information Processing Society of Japan

図 2 に提案手法の概要を示す．本手法は，(1) 単語分割機能，(2) 単語グラフ構築機能，(3) 特徴ベクトルに基づく学習・分類機能により構成される．3.1 節で学習用データセットについて，それ以降の節で各機能の詳細について述べる．

### 3.1 学習用データセット

同種のドメイン文字列から成るデータセットを $x \in X_i, i \in 0 \cdots n$ で記述する．ここで，$X_0$ は良性データセット，$X_1$ から $X_n$ は $n$ 種のボットにより生成された悪性データセットである．留意すべきは，ドメインをプライマリドメインまで短縮する点である．プライマリドメインは登録可能な最高レベルのサブドメインである [20]．具体的には，`www.ipsj.or.jp` と `smtp.kyutech.ac.jp` のプライマリドメインは，それぞれ `ipsj.or.jp` と `kyutech.ac.jp` となる．

### 3.2 単語分割機能

本機能は，先ずホワイトリストと合致したドメインを良性と判別する．ホワイトリストに含まれるのは，学習用データセットにおいて $th_\alpha$ 以上の頻度で発生するドメイン群である．これは，DGA の特性上，同一ドメインの名前解決を幾度も繰り返すことがないことに起因する．それに加え，DNS の仕様 [21] に反する文字列から成るドメインを，入力ミスや設定ミスにより生じたノイズとして除外する．

次いで，辞書 $\mathbb{D}$ に基づいてドメイン $x$ のプライマリレベルの文字列を単語群 $w$ に分割する．辞書 $\mathbb{D}$ は，クローリングで作成したコーパスと英語辞書である．各単語の文字長が最大且つ単語数が最小になること，極端な選択率の差により辞書に含まれる単語を優先することの 2 点を踏まえ，その分割を次式で示す．

$$\mathcal{F}(x) = \underset{w \in \mathbb{W}(x)}{\arg\max} \frac{1}{m} \prod_{j=1}^{m} \mathcal{P}(w_j)$$

$$\mathcal{P}(w_j) = \begin{cases} 1 & (w_j \in \mathbb{D}) \\ 1/|\mathbb{D}|^{|w_j|} & (w_j \notin \mathbb{D}) \end{cases}$$

ここで，$\mathbb{W}(x)$ はドメイン $x$ のプライマリレベルの文字列における全分割候補の集合，$w$ は単語 $w_1, \cdots, w_j, \cdots, w_m$ から成る候補の単語群，$|w_j|$ は単語 $w_j$ の文字長，$|\mathbb{D}|$ は辞書 $\mathbb{D}$ の総単語数をそれぞれ意味する．また，$\mathcal{P}(w_j)$ は，単語 $w_j$ が辞書 $\mathbb{D}$ に含まれるか否かに基づいて，単語 $w_j$ の選択率を導出する関数である．以降，文字列 *kyutechlocaldomain* が *kyutech*，*local*，*domain* の 3 単語に分割される場合，それを{*kyutech, local, domain*}と表記する．最終的に，本機能は学習用データセット $X_i$ におけるドメインから，そのプライマリレベルの文字列を成す単語群の集合 $W_i$ を出力する．

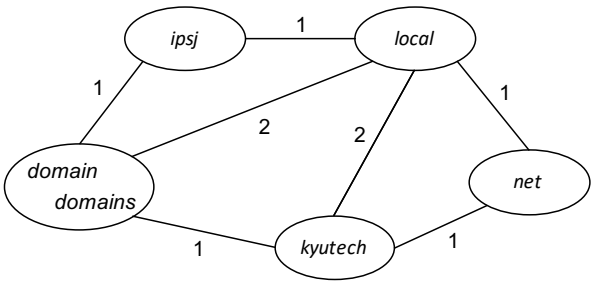



**図 3** ドメイン文字列を成す単語群を用いた単語グラフの構築例

### 3.3 単語グラフ構築機能

本機能は，単語グラフを用いることで，プライマリレベルの文字列を成す単語群の集合 $W_i$ の関係性を示す．単語グラフ $G_i$ は，頂点と辺の集合から成る重み付き無向グラフである．頂点は集合 $W_i$ における単語，辺の重みは単一ドメインにおいて 2 つの単語が共起する頻度を意味する．ここで留意すべきは，単語の活用を考慮するため，文字列が類似した単語を同一頂点に集約する点である．その類似性の導出には編集距離比を，集約には閾値 $th_\beta$ の重心法に基づく階層型クラスタリングを採用した [22]．図 3 に，{*kyutech, local, domain*}，{*local, kyutech, net*}，{*ipsj, domains, local*} の単語群を用いた単語グラフの構築例を示す．図における *domain* と *domains* は，その文字列の類似性から同一頂点に集約されることになる．

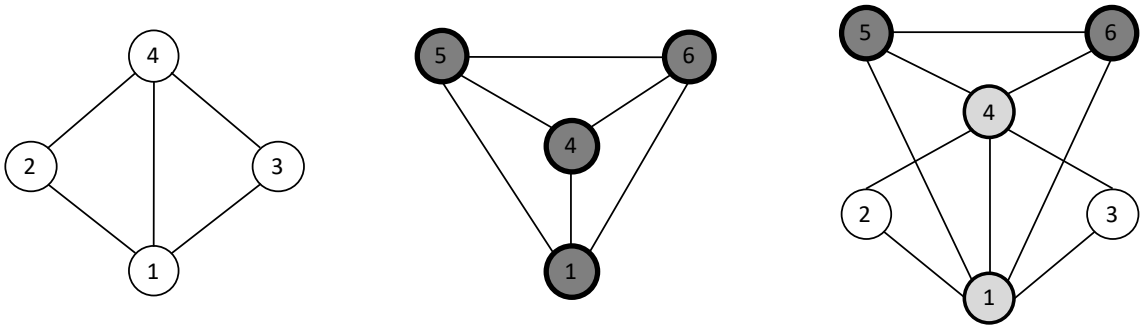
次に，2 つの単語グラフ間の処理としてグラフの結合を定義する．結合は，単語グラフ $G_i$ と $G_j$ を構築するために用いた単語群の集合 $W_i$ と $W_j$ の和から，単語グラフ $G_{i:j}$ を再構築する処理である．その具体例を図 4 に示す．図 4(a) と 4(b) の単語グラフ $G_i$ と $G_j$ の結合の結果は図 4(c) となる．ここで，白で示される頂点は単語グラフ $G_i$ のみ含まれる単語，黒で示される頂点は単語グラフ $G_j$ のみ含まれる単語，灰色で示される頂点はその両方に共通する単語を意味する．

機械的に生成した悪性ドメインと人為的に生成した良性ドメインでは，その文字列で頻出する単語や共起する単語に明確な差異が現れる．加えて，それら良性と悪性の判別に効果的な単語は単語グラフにおいて中心的な役割を担うこととなる．それらの観点を踏まえ，任意の単語群 $w$ の重要性を次式で示す．

$$\mathcal{S}_i(w) = \sum\nolimits_{w_j \in w} |w_j| \big( \mathcal{C}_{0:i}(w_j) - \mathcal{C}_0(w_j) \big)$$

ここで，$|w_j|$ は単語 $w_j$ の文字長を意味する．また，$\mathcal{C}_0(w_j)$ と $\mathcal{C}_{0:i}(w_j)$ は，単語グラフ $G_0$ と $G_{0:i}$ における単語 $w_j$ の中心性を導出する関数である．すなわち，良性データセットから構築した単語グラフ $G_0$ を基準として，それと悪性データセットから構築した単語グラフ $G_i$ の結合による中心性の変化量を単語 $w_j$ の重要性とする．その中心性の導出には，単語グラフが無向且つ非連結になることを勘案し

ⓒ 2019 Information Processing Society of Japan




(a) 単語グラフ $G_i$　(b) 単語グラフ $G_j$　(c) 単語グラフ $G_{i:j}$

**図 4**　単語グラフ $G_i$ と $G_j$ の結合例

て PageRank を採用した [23].

### 3.4　特徴ベクトルに基づく学習・分類機能

本機能は，先ず学習用データセットのドメインから特徴ベクトルを導出する．単語グラフにおける重要性に基づいて，単語群 $w$ から成るドメイン文字列の特徴ベクトルを次式で示す．

$$\vec{w} = \big(\mathcal{S}_1(w), \cdots \mathcal{S}_i(w), \cdots \mathcal{S}_n(w)\big)$$

次いで，それら特徴ベクトルに機械学習アルゴリズムを適用することで学習モデルを構築する．機械学習アルゴリズムには，その汎化性能と判別性能を勘案して SVM (Support Vector Machine) を採用した [24]．最終的に，未知のドメインはドメイン文字列の分割と特徴ベクトルの導出を経て，その学習モデルに基づくことにより良性と悪性を判別する．

## 4.　評価

本章では，実験を通じて提案手法を評価することで，DNS に対する膨大な数の名前解決要求から機械的に生成された悪性ドメインを判別できること，その結果によりネットワークに内在する辞書に基づく DGA ボットを高精度で検出できることを示す．4.1 節で実験の諸元について述べた後，4.2 節で結果について議論する．

### 4.1　諸元

表 2 に実験に用いたデータセットを示す．悪性ドメインは，実際に 7 種のボットにより生成されたドメイン群である [25]．ここで，Banjori と Sisron は単語にランダムな文字列を結合することでドメインを生成するボット，その他は自身が有す辞書の単語を結合することでドメインを生成するボットである．また，良性ドメインは，キャンパスネットワークにおける名前解決に対して NXDOMAIN 応答を返したドメイン群である．その観測結果にボットによる通信が含まれないことは，複数のセキュリティ製品と文献 [15] の解析結果に基づく目視により確認済みである．それら良性と悪性ドメインに対して 5 分割交差検証を適用することで，全体の 20% を検証用データセット，残りを学習用データセットとした．

提案手法との比較のため，文献 [5], [18] を参考にドメインの文字列のみから悪性ドメインを判別する 2 種類の手法を実装した．第一の実装は，ドメイン文字列に対して LSTM モデルを適用することで判別する手法であり，第二の実装は，ボットのコールバック先から構築した辞書の単語がドメイン文字列に閾値以上含まれるか否かで判別する手法である．

提案手法における閾値を経験的に $th_\alpha = 25$，$th_\beta = 0.85$ とした．また，SVM のカーネルとして RBF を採用した．これらの最適化は今後の課題とする．

### 4.2　議論

各手法における悪意ドメインの判別性能を定量的に評価するために，一般的な 2 つの指標を用いた．再現率は，悪性ドメインの総数に対する悪性と判別されたドメインの数の比率であり，適合率は，悪性と判別されたドメインの総数に対する真に悪性であるドメインの数の比率である．

実験結果を表 3 に示す．ここで，各値は交差検証における 5 回の試行の平均を意味する．この結果から，提案手法は 0.9977 の再現率と 0.9869 の適合率を達成しており，2 つの実装よりも高い精度を示すことが見て取れる．2 つの実装の精度が低下した理由は次の通りである．先ず，単純な文字の並びのみからドメインの良性と悪性を判別することの限界である．特筆すべきは，8 文字以上のアルファベットのみから成るドメインを悪性と判別する傾向が見られた点である．次いで，ドメイン文字列における単語の重要性を画一的に測るため，良性ドメインと悪性ドメインの一部で誤判が多発したことが原因である．提案手法は，良性と悪性のドメイン文字列を成す単語の差異を考慮すること，単語グラフにおいて中心的な役割を担う単語に着目することで，これらの要因を除外できたと考えられる．

各データセットにおいて誤判したドメインの数を表 4 に示す．ここで，各値は交差検証における 5 回の試行の合計を意味する．提案手法は，Banjori と Sisron が生成したド

ⓒ 2019 Information Processing Society of Japan

**表 2** 各データセットにおけるドメインの数

| $D0$ | $D1$ | $D2$ | $D3$ | $D4$ | $D5$ | $D6$ | $D7$ |
|---|---|---|---|---|---|---|---|
| Benign | Banjori | Gozi | Matsnu | Pizd | Rovnix | Sisron | Suppobox |
| 3021124 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 | 30000 |

**表 4** 各データセットにおけるドメインの誤判数

| | $D0$ | $D1$ | $D2$ | $D3$ | $D4$ | $D5$ | $D6$ | $D7$ |
|---|---|---|---|---|---|---|---|---|
| Anderson et al. [5] | 15631 | 0 | 154 | 0 | 62 | 27 | 0 | 220 |
| Pereira et al. [18] | 105706 | 0 | 285 | 26 | 23286 | 28 | 0 | 36 |
| Our work | 2772 | 0 | 295 | 31 | 29 | 32 | 0 | 82 |

**表 3** 実験結果

| | Recall | Precision |
|---|---|---|
| Anderson et al. [5] | 0.9977 | 0.9305 |
| Pereira et al. [18] | 0.8873 | 0.6380 |
| Our work | 0.9977 | 0.9869 |

メインに対して非常に優秀な判別を実現した．その一方，良性を含む他のデータセットにおいては少数の誤判が発生している．その誤判の多くは次の 4 種，(a) 単一の単語のみから成るドメイン，(b) 学習用データセットにおいて，出現頻度が極端に少ない単語から成るドメイン，(c) 一般的な単語を多数含むドメイン，(d) 辞書に含まれない単語から成るドメインであった．提案手法は，ドメイン文字列における単語の関係性に着目しており，良性と悪性の判別には教師付き機械学習アルゴリズムを利用している．それらの特性上，(a) と (b) のドメインの正確な判別は困難となる．(c) のドメイン文字列を調査したところ，*domain*，*network*，*host*，*local* など，良性・悪性を問わず頻出する単語を多く含んでいた．そのため，自然言語処理におけるストップワードを参考に [26]，ドメインにおける一般的な単語を除外することで改善が可能である．(d) の誤判は，辞書における語彙数の不足により，ドメインを無意味な短い文字列に分割することが原因である．その例としては，ランダムな文字列から成るドメインや非アルファベットをアルファベット表記したドメイン，固有名詞から成るドメインなどである．

文献 [17] では，マルウェアの通信に対する逆行解析を補助する方法，文献 [18] では，マルウェアの通信からドメインの生成に用いる辞書を推定する方法について言及している．一方，提案手法では，ドメイン文字列の分割に用いる辞書としてクローリングで作成したコーパスと英語辞書を採用している．辞書の語彙と範囲は良性・悪性ドメインの判別に大きく影響することから，これらの成果を踏まえることで更なる精度の向上が期待できる．

以上の議論より，幾つかの課題があるにしても提案手法が 0.9977 の再現率と 0.9869 の適合率で悪性ドメインを判別できることを確認した．この結果は，これら悪性ドメインの名前解決を予兆として，ネットワークに内在する辞書に基づく DGA ボットを高精度で検出できることを示唆している．

## 5. おわりに

本稿では，辞書に基づく DGA ボットの検出のため，DNS に対する膨大な数の名前解決要求から機械的に生成された悪性ドメインの判別を試みた．また実験を通じて，提案手法が 0.9977 の再現率と 0.9869 の適合率で悪性ドメインを判別可能であることを確認した．その結果から，ネットワークに内在するボットへの迅速な対処が可能になるため，ネットワークの運用において安全性の向上が期待できる．

今後の予定は，大規模なネットワークで観測した通信を対象に，判別精度と所要時間を評価する．それに加え，単位時間当りに判別可能なドメイン数を調査することで，ネットワークの規模と計算機性能の関係性を明らかにする．また，文献 [17], [18] を参考に，DGA ボットのコールバック通信から，そのボットが使用する辞書を再構築する方法を検討する．

### 参考文献

[1] M. Antonakakis et al.: Understanding the Mirai Botnet, *Proceedings of the USENIX Conference on Security Symposium*, pp. 1093–1110 (2017).
[2] A. K. Sood et al.: A Taxonomy of Domain-Generation Algorithms, *IEEE Security & Privacy*, Vol. 14, No. 4, pp. 46–53 (2016).
[3] A. Satoh et al.: Estimating the Randomness of Domain Names for DGA Bot Callbacks, *IEEE Communications Letters*, Vol. 22, No. 7, pp. 1378–1381 (2018).
[4] D. Truong et al.: Detecting Domain-Flux Botnet based on DNS Traffic Features in Managed Network, *Security and Communication Networks*, Vol. 9, No. 14, pp. 2338–2347 (2016).
[5] H. S. Anderson et al.: DeepDGA: Adversarially-Tuned Domain Generation and Detection, *Proceedings of the ACM Workshop on Artificial Intelligence and Security*, pp. 13–21 (2016).
[6] D. Andriesse et al.: Highly Resilient Peer-to-Peer Botnets Are Here: An Analysis of GameOver Zeus, *Proceedings of the International Conference on Malicious and Unwanted Software*, pp. 116–123 (2013).
[7] H. Binsalleeh et al.: On the Analysis of the Zeus Botnet Crimeware Toolkit, *Proceedings of the Annual International Conference on Privacy, Security and Trust*, pp. 31–38 (2010).
[8] B. Hammi et al.: An Empirical Investigation of Botnet as a Service for Cyberattacks, *Transactions on Emerging Telecommunications Technologies*, Vol. 30, No. 3, pp. 1–11

ⓒ 2019 Information Processing Society of Japan

(2019).

[9] M. Fossi et al.: Symantec Internet Security Threat Report, https://www.symantec.com/security-center/archived-publications (2011).

[10] F. Soldo et al.: Blacklisting Recommendation System: Using Spatio-Temporal Patterns to Predict Future Attacks, *IEEE Journal on Selected Areas in Communications*, Vol. 29, No. 7, pp. 1423–1437 (2011).

[11] J. Freudiger et al.: Controlled Data Sharing for Collaborative Predictive Blacklisting, *Proceedings of International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pp. 327–349 (2015).

[12] B. Rahbarinia et al.: Efficient and Accurate Behavior-Based Tracking of Malware-Control Domains in Large ISP Networks, *ACM Transactions on Privacy and Security*, Vol. 19, No. 2, p. 4:1–4:31 (2016).

[13] L. Bilge et al.: Exposure: A Passive DNS Analysis Service to Detect and Report Malicious Domains, *ACM Transactions on Information and System Security*, Vol. 16, No. 4, pp. 14:1–14:28 (2014).

[14] J. Geffner: End-to-End Analysis of a Domain Generating Algorithm Malware Family, *Black Hat USA* (2013).

[15] D. Plohmann et al.: A Comprehensive Measurement Study of Domain Generating Malware, *Proceedings of the USENIX Conference on Security Symposium*, pp. 263–278 (2016).

[16] Z. Xu et al.: AutoProbe: Towards Automatic Active Malicious Server Probing Using Dynamic Binary Analysis, *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, pp. 179–190 (2014).

[17] G. Viglianisi et al.: SysTaint: Assisting Reversing of Malicious Network Communications, *Proceedings of the Software Security, Protection, and Reverse Engineering Workshop*, pp. 1–12 (2018).

[18] M. Pereira et al.: Dictionary Extraction and Detection of Algorithmically Generated Domain Names in Passive DNS Traffic, *Proceedings of the International Symposium on Research in Attacks, Intrusions, and Defenses*, pp. 295–314 (2018).

[19] T. S. Wang et al.: DBod: Clustering and Detecting DGA-based Botnets using DNS Traffic Analysis, *Computers & Security*, Vol. 64, pp. 1–15 (2017).

[20] D. Sahoo et al.: Malicious URL Detection using Machine Learning: A Survey, *arXiv:1701.07179*, pp. 1–21 (2017).

[21] Mockapetris, P.: Domain Names — Implementation and Specification, IETF Request for Comments 1035 (1987).

[22] D. Müllner: fastcluster: Fast Hierarchical, Agglomerative Clustering Routines for R and Python, *Journal of Statistical Software*, Vol. 53, No. 9, pp. 1–18 (2013).

[23] G. Csárdi et al.: The igraph Software Package for Complex Network Research, *InterJournal Complex Systems*, No. 1695 (2006).

[24] A. Karatzoglou et al.: Support Vector Machines in R, *Journal of Statistical Software*, Vol. 15, No. 9, pp. 1–28 (2006).

[25] GitHub: Some results of my DGA reversing efforts, https://github.com/baderj/domain_generation_algorithms.

[26] J. Nothman et al.: Stop Word Lists in Free Open-source Software Packages, *Proceedings of the Workshop for NLP Open Source Software*, pp. 7–12 (2018).

ⓒ 2019 Information Processing Society of Japan