通信の多様化に向けた生物の環境適応性に基づくWebサイトへのぜい弱性スキャン検知

久世 尚美† 石倉 秀† 八木 毅‡ 千葉 大紀‡ 村田 正幸†

†大阪大学 大学院情報科学研究科
565-0871 大阪府吹田市山田丘 1-5
{n-kuze, s-ishikura, murata}@ist.osaka-u.ac.jp

‡NTT セキュアプラットフォーム研究所
180-8585 東京都武蔵野市緑町 3-9-11
{yagi.takeshi, chiba.daiki}@lab.ntt.co.jp

あらまし Web サイトへの攻撃が急増する一方，Web サービスが多様化しており，攻撃を収集・解析して未知の攻撃への対策に活用することが重要となっている．Web サイトへの攻撃を収集する方法として Web サーバ型ハニーポットを用いる方法が知られているが，クローラなどの正常なパケットを含む多量なパケットを収集するため，悪質なパケットを自動的に識別する機構が求められる．特に攻撃の準備動作として Web サイトのぜい弱性を確認するぜい弱性スキャンの特徴はクローラと類似しているため，後者を高精度に識別する必要がある．本稿では，多量かつ多様なデータの扱いに長けた生物の仕組みに着想を得たクラスタリング手法 AntTree をクローラ識別へと適用した．実験結果より，AntTree が高い精度で識別が可能であることを示した．

Website vulnerability scanning detection inspired by biological adaptation toward diversifying communication services

Naomi Kuze† Shu Ishikura† Takeshi Yagi‡ Daiki Chiba‡
Masayuki Murata†

†Graduate School of Information Science and Technology, Osaka University
1-5 Yamadaoka, Suita, Osaka 565-0871, JAPAN
{n-kuze, s-ishikura, murata}@ist.osaka-u.ac.jp

‡NTT Secure Platform Laboratories
3-9-11 Midori-cho, Musashino-shi, Tokyo 180-8585, JAPAN
{yagi.takeshi, chiba.daiki}@lab.ntt.co.jp

Abstract Attacks against websites are increasing rapidly with the expansion of web services. More and more diversified web services make it difficult to prevent such attacks due to many known vulnerabilities in websites. To overcome this problem, it is necessary to collect latest attacks using decoy web honeypots and to implement countermeasures against malicious threats. Web honeypots collect not only malicious accesses by attackers but also benign accesses by web search crawlers. Thus, it is essential to develop a means of identifying malicious accesses automatically from mixed collected data including both malicious and benign accesses. In this study, we have focused on detection of crawlers whose accesses has been increasing rapidly. We have adapted AntTree, a bio-inspired clustering scheme that has high scalability and adaptability, for crawler detection. Through our evaluations, we show that AntTree can detect crawlers more precisely than a conventional scheme.

1 はじめに

インターネットのインフラ化，Web サービスの浸透に伴い，それらを提供する Web サイトに対する攻撃が急増している．その一方で，Web サービスの形態は多様化し，全てのサービスの脆弱性を特定し，Web サイトへの攻撃を防ぐことは困難になっている．そのため，既知の脆弱性に基づいて攻撃への対策を行うのみではなく，攻撃を収集，解析して，未知の攻撃への対策に活用ことが必須課題となっている．

Web サイトへの攻撃を収集する方法としては，Web サイトへの攻撃を収集するシステム（Web サーバ型ハニーポット）を設置する方法が知られている [1, 2]．ハニーポットは，攻撃ベクトルに応じて構築されるおとりのシステムであり，Web サーバ型のハニーポットは，Web サイトへの攻撃を収集する際に使用される．ハニーポットは一般的にエミュレータである低対話型と，実際の OS やアプリケーションを用いる高対話型に大別される [3]．特に Web サイトへの攻撃は，おとりであることが攻撃者に感知されにくく，多くの情報を収集できる高対話型が使用される [1, 2]．本稿では，高対話型の Web サーバ型ハニーポットを，ハニーポットと呼ぶ．

ハニーポットへの通信は，悪意のある通信のみでなく，検索エンジンデータベース作成のためのクローラなどによる正常な通信も含まれているため，収集した情報の中から目的となる攻撃情報を分類する必要がある．しかし，通信料の増加に伴い，これらの通信ログの判別は困難になりつつある．特に Web サーバへの攻撃の準備動作として Web サイトのぜい弱性を確認するぜい弱性スキャンは，様々なプログラムに対して様々な入力値を試行するクローラと特徴が類似しているため，両者の識別は非常に困難である．文献 [1] においては，ハニーポットで収集した多量の通信ログから悪意のあるものを判別するために，まずクローラによる通信の識別を行い，それ以外の通信を攻撃と判定する手法が提案されている．この手法においては，まず Google などの有名クローラを公開されている UserAgent と IP アドレスから判別し，有名クローラと似た挙動を示す通信をクローラと判定している．しかしながら，Web サービスの多様化に伴い，攻撃者からの通信の形態はもちろんクローラなどからの正常な通信の形態も非常に多種多様となっており，通信の多様化にも適応可能な判別手法が必要となる．

そこで，本研究では，生物の仕組みに着想を得たクラスタリング手法をクローラ識別へと適用する．生物は，全体の情報を用いることなく，個々の個体が知覚可能な情報のみに基づいて動作を決定し，結果として全体としての秩序や機能を創発するため，特定の特徴を持ったデータの判別のみでなく，データ全体をそれぞれ異なる特徴を持ったクラスタへと分類することが可能となる．そのため，多様なデータ群の分類を自動で行うことが可能となる．また，生物は局所情報のみを用い単純なルールに基づいて動作を決定するため，高い拡張性を有しており，多量なデータを取り扱うケースへの適用にも向いている．このような特徴から，生物の仕組みのクラスタリングへの応用に関しては様々な研究が行われており [4]，本稿では，代表的な社会的昆虫で，古くからそのの挙動に着想を得た工学技術について盛んに行われているアリの集団的行動に着想を得た AntTree [5, 6] をクローラ識別へと適用した．AntTree は特にアリが互いに連結して木構造を構築する集団的行動に着想を得て，アリに見立てたデータ同士が連結して木構造を構築していくことでクラスタを形成する手法である．今回 AntTree を採用している理由としては，生物の特徴である拡張性，適応性を有しつつ，結果が木構造として得られるためにクラスタの解釈，および各クラスタの解析が容易であり，工学的な応用に優位であることが挙げられる．

本稿の貢献は以下の通りである．

- 生物の環境適応性を Web サイトへの攻撃検知に応用する手法を提案した．
- Web サーバ型ハニーポットに対する 1 年 4ヶ月間の攻撃を解析することで，提案手法では，従来の攻撃判定手法と比較して，正確にクローラを識別して攻撃を抽出できることを明らかにした．

2 関連研究

本章では，本稿で扱っている攻撃，および生物の仕組みに着想を得たクラスタリング手法に

ついて，既存の研究を紹介する．

2.1 Web サイトへの攻撃検知

Web サイトへの攻撃検知には，ソフトウェアの脆弱性を解析して生成されたシグネチャを用いる場合と，攻撃を収集して解析することで対策を講じる場合がある．近年の Web アプリケーションの爆発的な普及を考慮すると，前者を網羅的に実施することは困難なため，後者が必須である．攻撃を収集して解析する手法には，ユーザ PC やサーバでトラヒックをキャプチャして解析する手法 [7, 8] と，本稿が着目しているような，ハニーポットを配置して攻撃を収集する手法 [9] がある．前者は高い検知精度が期待できる一方，プライバシの問題やトラヒックのキャプチャがサービス品質に与える影響などを考慮する必要があるため，実施できる環境には制限がある．このため，後者が重要となる．

ハニーポットにおける攻撃の収集においては，先述のとおり，クローラ識別が大きな問題となる．Google はクローラに適用する UserAgent や IP アドレスなどを公開しているため，Google からのクローラは識別することができる．Google からのクローラのアクセスは，様々なプログラムに対して様々な入力値を試行する．具体的には，`example.com` に対応するホスト対して `wget -r -np -l 0 http://example.com/<directory_name>/` と同様のアクセスが発生する．一方，攻撃は，ぜい弱性があるプログラムに対して入力を試行する．特にぜい弱性スキャンでは，ぜい弱性の有無を確認するために，ぜい弱性があるプログラムに対して無害な値を入力することが多い．このため，ぜい弱性スキャンとクローラの類似性は非常に高く，Google 以外のクローラの識別が困難となっている．

2.2 生物由来のクラスタリング手法

自然界において群れをなしている個体が，それぞれ知覚可能な情報のみで簡単なルールの基で行動を決定し，結果として全体の秩序が生まれていく仕組みは，その高い拡張性，適応性，耐故障性，柔軟性から，工学，経済学などの様々な分野に応用されている．クラスタリングに関しても，大量のデータを低コストで扱うことや容易な実装が求められる点が生物の仕組みとの親和性が高く，様々な研究が行われている．代表的なものとして，鳥や魚など群れにおいて，ある個体が食料を発見すると周囲の個体がそれに倣っていき，やがて群れ全体に伝搬していく様子をモデル化した Particle Swarm Optimization (PSO) を応用したもの [10] やアリが食料を見つけると巣までの帰路に揮発性のフェロモンを残して他のアリに食料の在り処を示すことですばやく食料までの最適経路を発見，維持する様子をモデル化した Ant Colony Optimization (ACO) を応用したもの [11] が挙げられる．こうした生物由来のクラスタリング手法は，以下の要件を満たす必要がある Web サイトへの攻撃の検知においても有効に働くと考えられる．

1. 膨大な通信から識別を行うため，大量のデータを扱う必要がある．
2. 悪意のある通信などの識別対象の特徴をあらかじめ全て特定することが困難であり，特定の特徴の通信を識別するだけでなく，未知の特徴から識別を行う必要がある．
3. 常に変化し続ける通信の傾向に適応可能である．

3 AntTree

ここでは，本研究でクローラ識別へと適用を行うアントベースのクラスタリング手法 AntTree [5, 6] について説明を行う．AntTree は，アルゼンチンアリ（*Linepithema humile ant*），ハタオリアリ（*Oecophylla longinoda ant*）が，互いに連結して木構造を構築する集団的行動に着想を受けたクラスタリング手法である．文献 [12] では，ハタオリアリが木構造を構築する集団的行動について論じている．ハタオリアリは木の間を移動する，あるいは木に生えた葉を一箇所に集めて巣を作る際に，葉や枝を基点として木構造を構築する．

3.1 概要

AntTree は，アリを模したデータが互いの類似度に基づいてツリー構造を構築することでデータのクラスタリングを行う手法である．AntTree では，データの一つ一つを ant と呼ばれるモバイルエージェントとみなしており，また ant が木構造を構成するノードとして互いに連結していく（図 1）．初期状態では，全ての ant は support と呼ばれる根に存在し，support から ant が一つずつ移動を開始する．Support を出発した ant は，

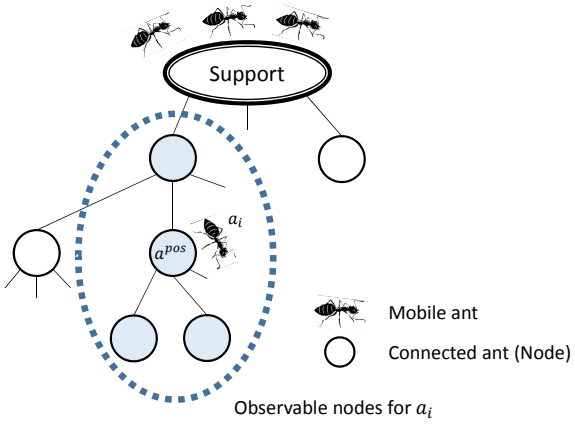

図 1: AntTree

後述の類似度に基づいて近隣のノード（現在存在するノード，およびその親ノードと子ノード）と自身の比較を行いながらツリー上に連結されたノード（ant）の間を移動する．移動中の ant は，自身と類似した特徴を持つノードに到達し，かつ近隣により類似したノードが存在しない場合には，ant はそのノードの子ノードとなる．子ノードとなった ant は移動を停止し，support に存在する ant の一つが新たに移動を開始する．

本手法では，ant a_i，a_j（$i, j \in [1, \cdots, N]$）との類似度を $Sim(a_i, a_j) \in [0, 1]$ で表す．また，各 ant a_i は他の ant との類似性を測る閾値 $T_{Sim}(a_i)$，および非類似性を測る閾値 $T_{Dissim}(a_i)$ を保持しており，ant a_i が自身と ant a_j を比較した際に，$Sim(a_i, a_j) \geq T_{Sim}(a_i)$ であれば ant a_j が自身と類似した特徴を持っていると判定，一方で $Sim(a_i, a_j) < T_{Dissim}(a_i)$ であれば ant a_j が自身と異なる特徴を持っていると判定する．二つの閾値 $T_{Sim}(a_i)$，$T_{Dissim}(a_i)$ は ant a_i がツリー構造上を移動を行う過程で更新を行い，適切な値を学習する．

3.2 アルゴリズム

この節では，ant の挙動について詳しく説明を行う．

初期状態では，ツリー構造は support のみで構成され，全ての ant が support に移動可能な状態で存在する．最初に移動を開始した ant は，support の子ノードとなり移動を停止し，後続の ant が移動を開始する．

後続の ant a_i は，まず自身と support の子ノードとの比較を行う．Support の子ノードに a_i と類似したノードが存在する場合，つまり $Sim(a_i, a_j) \geq T_{Sim}(a_i)$ を満たすノード a_j が存在する場合，a_i は最も類似度の高いノードへと移動する．一方で，support の子ノードが全て a_i と異なる特徴を持つ場合，つまり support の子ノード a_j 全てについて $Sim(a_i, a_j) < T_{Dissim}(a_i)$ となる場合，a_i は support の新たな子ノードとなって移動を停止し，後続の ant が移動を開始する．このとき，support，および全てのノードにそれぞれ連結可能な子ノードの最大数 l を超えて子ノードとなることはできず，support が既に l 個の子ノードと連結している場合には，a_i は自身の類似性閾値 $T_{Sim}(a_i)$ を後述の (1) に基づいて減少させた後，support の子ノードの内 a_i との類似度が最も高いノードへと移動する．上記のどちらにも当てはまらない場合には，ant a_i は自身の類似性閾値 $T_{Sim}(a_i)$，非類似性閾値 $T_{Dissim}(a_i)$ を (1)，(2) に基づいて更新し，support の子ノードの内最も類似度の高いノードへと移動する．

$$T_{Sim}(a_i) \leftarrow T_{Sim}(a_i) \times \alpha_1 \quad (1)$$

$$T_{Dissim}(a_i) \leftarrow T_{Dissim}(a_i) + \alpha_2 \quad (2)$$

Ant 同士が比較を行う際，類似性の閾値 $T_{Sim}(a_i)$ が高く，非類似性の閾値 $T_{Dissim}(a_i)$ が高い場合には，ant 同士が異なる特徴を持つと判定されて別々のクラスタへと分類されやすく，反対に類似性の閾値 $T_{Sim}(a_i)$ が低く，非類似性の閾値 $T_{Dissim}(a_i)$ が低い場合には，ant 同士が類似した特徴を持つと判定されて同じクラスタへと分類されやすくなる．Ant が support から移動を開始する時点では，これらの閾値は $T_{Sim}(a_i) = 1$，$T_{Dissim}(a_i) = 0$ に設定されており，ツリー上を移動する過程で更新されていき，適切な値を学習していく．α_1，α_2 は，それぞれ閾値更新の際の類似性の閾値の減少量，非類似性の閾値の増加量を決定するパラメータである．これらの値が高いほど閾値の変動が大きくなり，類似，あるいは非類似ノードの発見が早くなり，クラスタの形成速度が向上するが，一方で誤検知率が高くなる．

Ant a_i が support 以外のノード a^{pos} に到達したとき，a_i はまず自身と a^{pos} との比較を行う．a_i が a^{pos} を類似した特徴を持つと判定した場合，つまり $Sim(a_i, a^{pos}) \geq T_{Sim}(a_i)$ である場合，さ

らに a_i は自身と a^{pos} の親ノード，子ノードとの比較を行う．a_i が a^{pos} の親ノード，子ノードいずれとも異なる特徴を持つと判定した場合，a_i は a^{pos} の新たな子ノードとなる．a^{pos} が既に l 個の子ノードと連結している場合には，a_i は a^{pos} の親ノード，子ノードからランダムにノードを選択し，移動を行う．一方で，a_i が a^{pos} の親ノード，子ノード全てと異なる特徴を持つと判定できない場合，ant a_i は (1)，(2) に基づいて自身の類似性閾値 $T_{Sim}(a_i)$，非類似性閾値 $T_{Dissim}(a_i)$ を更新して，a^{pos} の隣接ノードからランダムに一つ選択して移動を行う．

Ant a_i が a^{pos} に到達したとき，a_i と a^{pos} とが類似した特徴を持つと判定できない場合には，a_i は a^{pos} の隣接ノードからランダムに一つ選択し，移動を行う．

4　AntTree のクローラ識別適用

本章では，AntTree のクローラ識別への適用について説明を行う．

4.1　類似性

AntTree において，ant a_i と ant a_j との類似度 $Sim(a_i, a_j)$ は 3 章で示したように，移動中の ant がツリー上の類似した ant を探索し，ツリーを構築していく際の指標として用いられる．本研究では，文献 [5] に基づき，類似度 $Sim(a_i, a_j)$ を ant a_i，a_j の特徴ベクトル空間上のユークリッド距離 $d(a_i, a_j)$ を用いて，下式で定義する．

$$Sim(a_i, a_j) = 1 - d(a_i, a_j) \tag{3}$$

類似度 $Sim(a_i, a_j)$ が高い値を持つほど，ant 同士が類似した特徴を持つ．Ant a_i が M 個の特徴量 $\{v_{i_1}, \cdots, v_{i_M}\}$ を持つとき，$d(a_i, a_j)$ は以下で表すことができる．

$$d(a_i, a_j) = \sqrt{\frac{1}{M}\sum_{k=1}^{M}(v_{i_k} - v_{j_k})^2} \tag{4}$$

本稿で用いる特徴量に関しては，5.3 節で詳しく説明を行う．

4.2　クラスタの解釈

本研究では，構築されたツリーにおいて深さ h のノードを根とした部分木を一つのクラスタとみなす．図 2 において，h を 2 とした場合のクラスタ解釈例を示す．

本稿では，クローラ識別の精度を従来手法 [1] と比較するため，各クラスタをそのクラスタに属する最多のノード種別（クローラ，非クローラ）へと分類するものとしている．クローラ，非クローラが同数存在する場合には，クラスタはその根のノード種別へと分類する．

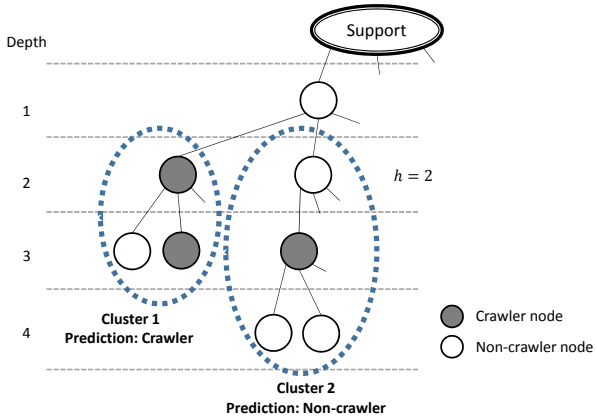

図 2: AntTree におけるクラスタの解釈 ($h = 2$)

5　提案手法の評価

提案手法について，実ネットワークで収集した通信ログを用いて実験を行い，有効性を示す．

5.1　概要

AntTree を用いたクローラ識別について，実ネットワークで収集した通信ログを用いて評価を行う．収集された通信ログの内，Google による通信ログは Google がクローラ情報を公開しているため，容易に判別することが可能となっている．そのため，本実験では初めに Google による通信ログを除外し，Google 以外のクローラとクローラ以外の通信との識別を行う．本実験で用いる実験用データセット，および特徴ベクトルに関しては，それぞれ 5.2，5.3 節において詳しく説明する．

本実験では，有名クローラの特徴からその他のクローラの識別を行っている文献 [1] のクローラ識別手法との比較を行う形で評価を行う．有名クローラとしては，公開情報から容易に識別可能な Google を用いており，Google との通信とクローラ以外の通信を訓練用データセットとして学習を行って分類モデルを作成し，Google 以外のクローラとの通信とクローラ以外の通信との識別に用いる．この手法では，学習アルゴリズムとして RandomForest[13] を用いている．

なお，本実験において，AntTree の評価には C++で実装したプログラムを，従来手法の評価

表 1: ハニーポットにより収集された通信ログ

ラベル	数
Google	8,276,246
Crawler	1,502,254
Non-crawler	11,547,739
Other	710,708
合計	22,036,947

には RapidMiner 5 で提供される RandomForest スキームをそれぞれ用いている．

5.2 データセット

本実験では，実験用データセットとして，多くの攻撃情報を収集できるハニーポット [14] 37 台により，2013 年 8 月 29 日 ~2014 年 1 月 14 日の期間に収集した HTTP 通信を解析した際のログを用いる．それぞれの通信ログは，ハニーポットが外部からリクエストパケットを受信し，受信したリクエストパケットに対してレスポンスパケットを送信する（あるいはリクエストパケットを送信せず通信を中断させる）までの一連の情報を含んでいる．上記の期間中に収集されたログ約 2,203 万個に対し，以下のラベル付けを行う．

- *Google*（約 827 万個）: Google が使用するクローラによる通信ログ．判別には，Google が公開している UserAgent 名，および送信元 IP アドレスの AS 事業社名を用いる．
- *Crawler*（約 150 万個）: Google 以外が運用しているクローラによる通信ログ．研究者，技術者が収集した通信ログを手動で解析し，クローラと特定した UserAgent，AS 事業社名から判別を行う．
- *Non-crawler*（約 1,154 万個）: クローラ以外による通信ログ．ぜい弱性スキャンをはじめとした各種攻撃で構成される悪意のある通信ログが含まれる．
- *Other*（約 71 万個）: 上記三つに分類されない通信ログ．主に解析に十分な情報が収集されなかったログが含まれる．今回の評価では用いていない．

上記 3 種類のラベル付けがされた通信ログの内，Google による通信ログは Google がクローラ情報を公開しているため，容易に判別することが可能となっている．本実験では，まず Non-crawler ログから Crawler ログと同数の 1,502,254 個のログをサンプリングし，それを 1,502,254 個の Crawler ログとマージし，計 3,004,508 個のログを実験用データセットとしている．この実験用データセットから Crawler ログを正しく識別できるかについて評価を行う．なお，時間帯，曜日あるいは年などに由来する時間的なデータの偏りに依存せずに識別精度の評価を行うため，ログのサンプリングはランダムに行っている．

表 2: 特徴量の数

特徴量の種類	数
Request	89
Response	37
合計	126

従来手法 [1] では有名クローラを送信元 IP アドレスや UserAgent から識別し，有名クローラと類似した特徴を持つパケットをクローラと判定していた．今回は，提案手法と文献 [1] の定量評価を実施するために，以下の手順でクローラの識別を行った．

1. 有名クローラとして判別が容易な Google クローラを用い，Google とラベル付けされたログから 1,502,254 個ランダムにサンプリングしたものと前述の同数の Non-crawler ログを訓練用データセットとする．
2. 訓練用データセットに対して，5.3 節で説明する特徴量ベクトルを用いて RandomForest で学習を行い，識別モデルを作成する．
3. 前述の実験用データセットをテストセットとして，作成した識別モデルを用いて Google ログの特徴から他の Crawler の識別が正しく行えるかについて評価を行う．

5.3 特徴ベクトル

ここでは，AntTree によりクラスタリングを行う際に類似度の評価に用いる特徴量ベクトルについて説明する．本実験では，ハニーポットで収集可能な情報から，通信ログの識別に有用であると考えられるものの選別・加工を行い，特徴量ベクトルの設計を行った．具体的には，クローラ識別に用いる特徴量はリクエストパケットの情報に関するもの，リクエストパケットに対するレスポンスの情報に関するものの 2 つに分けられる．

- リクエストパケット:

ハニーポットが外部から受信した HTTP リクエストパケットに含まれる情報，およびそれらを解析したもの．具体的には以下の情報を用いている．

- リクエスト部: リクエスト URL，通信メソッド
- ヘッダ部: UserAgent，referer，送信・受信側ポート番号，通信プロトコル
- ボディ部: ボディ部の長さ

- リクエストに対する応答:
  ハニーポットが，HTTP リクエストパケットを受信した際に，そのパケットに対して行った動作に関する情報，およびそれらを解析したもの．具体的には，リクエストパケット受信後のハニーポットの挙動（レスポンスパケット送信の有無等），レスポンスパケットを送信した際には，それに付加される StatusCode，コンテンツの種類が挙げられる．

これらの情報は，情報間の重みを均等にするため $[0,1]$ の値をとる実数へと変換し，特徴量として識別に用いている．データ i（$i \in [1,\cdots,N]$）の特徴量 v_{i_j}（$j \in [1,\cdots,M]$）は下式に基づいて正規化される．

$$\frac{v_{i_j} - \min_{n\in[1,\cdots,N]} v_{n_j}}{\max_{n\in[1,\cdots,N]} v_{n_j} - \min_{n\in[1,\cdots,N]} v_{n_j}}$$

通信ログの情報から特徴量への変換の一例として，リクエストパケットに含まれるリクエスト URL をどのように特徴量として用いるかについて以下で述べる．

1. リクエスト URL に含まれる文字数を一つの特徴量とする．
2. リクエスト URL をリクエストの対象となるリソースの場所を指すパス部とパラメータを指定するクエリ部に分割し，次の情報を特徴量とする．(a) パス部でリクエストされたパスの階層数，および階層ごとの文字列長の平均．(b) クエリ部で指定されたパラメータの数，およびパラメータごとの文字列長の平均．
3. リクエスト URL の文字列の特徴を分類，クラスタリング手法で一般的に扱いやすい数値データとして用いるために，< 文字列の種類（string 型，integer 型，hex 型など）; 文字列長 > の形で正規表現化 [15] した後，得られた正規表現に含まれる文字列の種類とその割合をパス部，クエリ部それぞれに対して求め，特徴量とする．

本実験で用いている特徴量の数は表 2 に示している．

表 3: AntTree におけるパラメータ設定

パラメータ	値
l	5
h	3
α_1	0.95
α_2	0.2

5.4 評価結果

ここでは，AntTree によるクローラ識別の精度について，従来手法と比較を行う形で評価を行う．AntTree におけるパラメータ設定は表 3 に従う．本実験では，識別精度を評価する指標として再現率，適合率を用いる．再現率は同一ラベルを付加されたデータの内，正しく分類されたデータの割合，適合率は同一のカテゴリへ分類されたデータの内，正しく分類されているデータの割合で定義される．

5.2 節で示した実験用データセットを用い，従来手法，AntTree によりクローラの識別を行った結果を表 4，5 にそれぞれ示す．これらの表において，ラベルは 5.2 で示した基準に基づいて各ログにあらかじめ付加されたラベル，予測は従来手法，あるいは AntTree により識別した結果を示す．Crawler，Non-crawler のラベルが付加されたログがそれぞれ正しく識別された割合（再現率），Crawler，Non-crawler それぞれに識別されたログの識別結果が正しい割合（適合率）いずれに着目した場合も，AntTree を用いた場合の方がより高い精度を示していることが確認できる．つまり，AntTree を用いることで，クローラログの誤検知率，見逃し率が低下している．これは，クローラの多様化により，クローラ同士が必ずしも類似した特徴を持つわけではなく，従来手法のように既知のクローラ（今回は Google）の特徴から他のクローラの識別を行うことが困難であるためであると考えられる．従来手法に対して，AntTree は個々の通信ログの特徴の比較し，類似するもの同士を同一のクラスタへと分類していくため，多様な通信の分類，

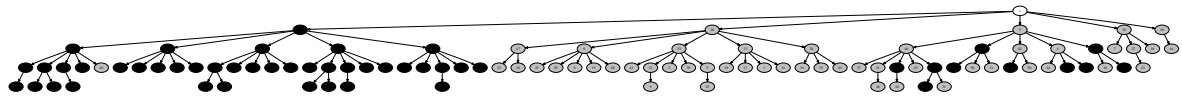

図 3: AntTree による識別の例．Crawler ログ，Non-crawler ログをそれぞれランダムに 50 個ずつサンプリングしたものを実験用データセットとし，パラメータ設定は表 3 に従う．白ノードが support，黒ノードが Crawler ログ，灰ノードが Non-crawler ログを表している．

表 4: 従来手法

		予測		再現率
		Crawler	Non-crawler	
ラベル	Crawler	1,241,437	260,817	82.64%
	Non-crawler	105,952	1,396,302	92.95%
適合率		92.14%	84.26%	

表 5: AntTree

		予測		再現率
		Crawler	Non-crawler	
ラベル	Crawler	1,259,976	242,278	83.87%
	Non-crawler	76,417	1,425,837	94.91%
適合率		94.28%	85.48%	

識別に有用である．

また，AntTree による分類結果の特徴として，データ全体としては比較的サイズの小さなクラスタの分類が正しく行われていることが挙げられる（図 3）．これは，AntTree においては，各データがツリー上を移動しながら局所的な情報を用いて類似するデータの探索を行うため，データ全体から見たそれぞれの通信の特徴の重要度にかかわらず分類が可能であるためである．これは，新規の通信傾向についても少ないサンプルからの分類を可能にし，常に変化し続ける通信傾向にも適応していくことができると考えており，引き続き検討を進める．

結論として，生物の仕組みに着想を得た手法である AntTree を用いることにより，従来手法と比較してクローラの高い精度での識別が実現できており，通信の多様化にも適応可能なことを示した．

6 おわりに

通信の多様化により，膨大な通信の中から，従来のように既知の特徴のみから通信の識別を行うことは困難となっている．本稿では，通信の多様化にも適用可能な手法として，アリの仕組みに着想を得たクラスタリング手法である AntTree をクローラ識別に適用し，実験を通して従来手法と比較して高い精度での識別が可能であることを示した．

今後の課題として，AntTree について，通信の傾向の変化を考慮した解析，実験を行っていきたいと考えている．また，同一の IP アドレスや AS からの通信を一つの集合として，そこから得られる統計的な特徴を用いた識別に関しても検討を進めていきたいと考えている．

参考文献

[1] J. P. John, F. Yu, Y. Xie, A. Krishnamurthy, and M. Abadi, "Heat-seeking honeypots: Design and experience," in *Proc. of the 20th International Conf. on World Wide Web*, Mar. 2011, pp. 207–216.

[2] D. Canali and D. Balzarotti, "Behind the scenes of online attacks: an analysis of exploitation behaviors on the web," in *Proc. of 20th Annual Network & Distributed System Security Symposium (NDSS 2013)*, Feb. 2013.

[3] N. Provos and T. Holz, *Virtual honeypots: from botnet tracking to intrusion detection*. Addison-Wesley Professional, Jul. 2007.

[4] C. Grosan, A. Abraham, and M. Chis, *Swarm Intelligence in Data Mining*. Springer Berlin Heidelberg, 2006, vol. 34.

[5] H. Azzag, N. Monmarche, M. Slimane, and G. Venturini, "AntTree: a new model for clustering with artificial ants," in *Proc. of IEEE Congress on Evolutionary Computation (CEC 2003)*, vol. 4, Dec. 2003, pp. 2642–2647.

[6] H. Azzag, G. Venturini, A. Oliver, and C. Guinot, "A hierarchical ant based clustering algorithm and its use in three real-world applications," *European Journal of Operational Research*, vol. 179, no. 3, pp. 906–922, Jun. 2007.

[7] C. Kruegel, G. Vigna, and W. Robertson, "A multi-model approach to the detection of web-based attacks," *Computer Networks*, vol. 48, no. 5, pp. 717–738, Feb. 2005.

[8] T. H. Kim, K. Kim, J. Kim, and S. J. Hong, "Profile-based web application security system with positive model selection," in *Proc. of the 2nd Joint Workshop on Information Security*, Aug. 2007.

[9] T. Yagi, N. Tanimoto, and T. Hariu, "Design of provider-provisioned website protection scheme against malware distribution," *IEICE transactions on communications*, vol. 93, no. 5, pp. 1122–1130, May 2010.

[10] S. Alam, G. Dobbie, Y. S. Koh, P. Riddle, and S. U. Rehman, "Research on particle swarm optimization based clustering: A systematic review of literature and techniques," *Swarm and Evolutionary Computation*, vol. 17, pp. 1–13, Aug. 2014.

[11] O. M. Jafar and R. Sivakumar, "Ant-based clustering algorithms: A brief survey," *International journal of computer theory and engineering*, vol. 2, no. 5, pp. 1793–8201, Oct. 2010.

[12] A. Lioni, C. Sauwens, G. Theraulaz, and J.-L. Deneubourg, "Chain formation in oecophylla longinoda," *Journal of Insect Behavior*, vol. 14, no. 5, pp. 679–696, Sep. 2001.

[13] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R news*, vol. 2, no. 3, pp. 18–22, Dec. 2002.

[14] T. Yagi, N. Tanimoto, and T. Hariu, "Intelligent high-interaction web honeypots based on url conversion scheme," *IEICE transactions on communications*, vol. 94, no. 5, pp. 1339–1347, May 2011.

[15] T. Nelms, R. Perdisci, and M. Ahamad, "ExecScent: Mining for new C&C domains in live networks with adaptive control protocol templates." in *Proc. of 22nd USENIX Security Symposium*, Aug. 2013, pp. 589–604.