攻撃通信を持続的に検知する合成型機械学習手法の検討

小久保 博崇 † 満保 雅浩 ‡ 岡本 栄司 †

† 筑波大学大学院システム情報工学研究科
305-8573 茨城県つくば市 天王台 1-1-1
s1020718@u.tsukuba.ac.jp
okamoto@risk.tsukuba.ac.jp

‡ 金沢大学理工研究域
920-1192 金沢市角間町
mambo@ec.t.kanazawa-u.ac.jp

あらまし 近年, マルウェアの増加率が過去最大になっており, 未知のマルウェアが頻出している. そのため, 未知のマルウェアの侵入や活動を検出し, 被害を防ぐ必要がある. 本論文では CCC DATAset2011 の攻撃通信データを利用し, 通信プロトコルヘッダの特徴を, 性質の異なる複数の機械学習手法を組み合わせて学習することで未知攻撃を含む攻撃通信の持続的な検知を試みた.

A Combined Machine Learning Method for Sustainable Detection of Attacks

Hiroaka Kokubo† Masahiro Mambo‡ Eiji Okamoto†

†Graduate School of Systems and Information Engineering, University of Tsukuba
1-1-1 Tennodai, Tsukuba City, Ibaraki, 305-8573 JAPAN
s1020718@u.tsukuba.ac.jp
okamoto@risk.tsukuba.ac.jp

‡Institute of Science and Engineering, Kanazawa University
Kakuma, Kanazawa, Ishikawa, 920-1192, JAPAN
mambo@ec.t.kanazawa-u.ac.jp

Abstract Recently, the malware proliferation rate is highest ever and unknown malware appears frequently. Therefore, it is necessary to detect the invasion and the activity of unknown malware, and to prevent damage. In this paper, we combine multiple machine learning methods to achieve sustainable detection of attack communication including unknown attacks. We use the attack communication data of the CCCDATAset2011 for the analysis of the proposed method.

1 はじめに

近年, マルウェアの増加が過去最大になっており, 未知のマルウェアや攻撃も頻出している状況となっている [1]. 現在のマルウェアは, 過去の愉快犯的な物とは異なり, 社会に実害を及ぼす大きなリスクとなっている. マルウェアの侵入や攻撃を検出する手法としては, シグネチャとマッチングを行い検出する方式（ミスユース型）が現在主流となっているが, 一種類のマルウェアに対して膨大な量の亜種・変異種が存在するケースも多いため, マルウェアのエミュレーション等を行わない限り必要なシグネチャの数も膨大となってしまう. 対して, アノマリ型の検知方式はパターンマッチングや機械学習の技術を用いて未知の攻撃通信やマルウェアも検知することが可能である. しかし, シグネチャ型と比べて誤検知が多いといった欠点もあるため,

克服すべき課題は存在している．本論文はサイバークリーンセンター (CCC) で収集された研究用データセット CCC DATAset 2011[2] に含まれる攻撃通信データを利用し, 調査実験を行う．複数の機械学習手法を連結することで精度向上を図り, また各機械学習の学習タイミングを工夫することで未知攻撃が頻出する状況でも常に精度を保つことができないかを考察する．

2　関連研究

柿本ら [3] は機械学習を異常・侵入検知に自己組織化マップ [4] を使用することを提案している．マルウェアが実行時に呼び出すシステムコール列を自己組織化マップの入力列として与え, 異常検知を行おうとしている．自己組織化マップを使用する利点として, 未知のデータの性質を近接する既知のデータから類推することができることなどを挙げている．山田 [5] は, アノマリ型の侵入検知の教師情報としてミスユース型の検知結果を用いる方式を考案している．対象通信データとして DARPA Intrusion Detection Data Sets[6] を用いており, シグネチャで見逃した攻撃通信の検知に成功している．

3　実験・手法

決定木と二次元自己組織化マップを実装・連結し, 攻撃通信と正常通信の分類を試みる．逐次学習が可能な自己組織化マップと, 一定量のまとまったデータで定期的に学習する決定木を組み合わせることで徐々に変化する攻撃通信を検知し続けることを目標とする．

実装・実験環境を表 1 に示す．

表 1: 実装・実験環境

OS	Windows7 64bit
言語	Visual C#.NET
プロセッサ	Intel Core i7 2.2GHz
RAM	8GB
pcap 操作ライブラリ	自作物

3.1　使用するデータと特徴量

異常通信データとして CCC DATAset2011 の攻撃通信データを使用する．また, 著者の研究室・自宅で WEB サイト閲覧・動画閲覧などを行って発生させた日常的な通信を tcpdump によりキャプチャしたデータを正常通信データとして使用する．機械学習は, 入力としてデータの特徴量を受け取り, 学習または分類結果の出力を行う．特徴量は機械学習の精度に大きく影響する．今回はチェックサム等の明らかに関係の薄い項目を除いて各データから特徴量を抽出し実験を行う．使用する特徴量の項目は次の通りである．

- キャプチャ時獲得データ
  - パケット全体のサイズ
- データリンク層
  - ネットワーク層で使われているプロトコル番号
- ネットワーク層
  - IP
    - ヘッダ長
    - データグラム長
    - Flag
    - TTL
    - トランスポート層プロトコル
  - ICMP
    - メッセージタイプ
    - コード
  - IGMP
    - タイプ
- トランスポート層
  - TCP
    - 送信元ポート
    - 宛先ポート

    - ヘッダ長
    - URG〜FIN の各種フラグ
- UDP
    - 送信元ポート
    - 宛先ポート
    - データ長

宛先・送信元 IP アドレスや MAC アドレスは実運用上は非常に重要な特徴量だと考えられるが, 正常通信データと攻撃通信データの採取条件が異なるため, それらの特徴量を採用してしまうと不当な過学習を起こしてしまうと考えるため, 今回は不採用とした.

3.2 決定木

多数の if 文で構成された木を根から辿ることで入力データの判別を行う一般的な手法である. 決定木のメリットとして, 教師有り学習のため高め正答率が期待できること・完成した決定木は if 分の列なので動作が軽いことが挙げられる. 決定木を構成するためには教師データが必要となる. 本論文では正常通信・攻撃通信の一部から教師データを生成し, 決定木の構成にあてた. 教師データを一番正確に分割する条件式を総当たりで探索し, 教師データがある程度分離するまでそれを繰り返すため教師データ数や特徴量の項目・取りうる値が多い場合, 決定木の構成は非常に時間がかかる.

3.3 二次元自己組織化マップ

自己組織化マップとは T. コホネンが提唱した教師なしニューラルネットワークアルゴリズムである [4]. 多次元の入力データを, 近い性質のデータが近い座標に配置されるように二次元平面上に写像を行う. 自己組織化マップのメリットとして, 位置関係から未知の通信の性質が類推できること・判別を行いながら学習ができることが挙げられる. 本論文では二次元自己組織化マップ (以下,SOM と呼ぶ) を用いた分類プログラムを作成した. 概要は次の通りである. $i = 40, j = 40$ のインデックスを持つ二次元平面上にユニット $u_{i,j}$ を並べる. ユニットはそれぞれ参照ベクトル $rv_{i,j} = [e_{i,j,0}, e_{i,j,1}, ..., e_{i,j,n}]$ を持っている. $e_{i,j,n}$ はそのユニットが正常通信と判断された回数と攻撃通信と判断された回数の両方を保持する要素である. このユニット集合を SOM と呼ぶ. SOM に対して行える操作は学習と判別である.

学習:入力ベクトル $iv = [e_0, e_1, ..., e_n]$ を与える. 入力ベクトル iv と最も似ている参照ベクトル $rv_{i,j}$ を持ったユニット $u_{i,j}$ を選び, 勝者ユニットとする. 勝者ユニットとその周囲にあるユニットの参照ベクトルを, 次の式で更新する. $rv_{i,j} + ((iv - rv_{i,j}) * c(distance, time)$ (但し,c は距離 distance と時間 time によって変化する係数列).

判別：入力ベクトル $iv = [e_0, e_1, ..., e_n]$ を与える. 入力ベクトル iv と最も似ている参照ベクトル $rv_{i,j}$ を持ったユニット $u_{i,j}$ を選び, 勝者ユニットとする. 勝者ユニットの参照ベクトル中の判別結果が書かれている要素 $e_{i,j,n}$ を参照する. $e_{i,j,n}$ を見て, 正常通信と判別された回数と攻撃通信と判別された回数どちらか大きい方を判別結果として返す.

以上が学習と判別の操作である. 本論文では, 精度向上のため次の方法で SOM の初期化を行った.

初期化：正常通信・攻撃通信データから 1 日分を抜き出し教師データとする. インデックス i, j を二つの領域に分割し, 片側を正常通信から抽出した入力ベクトルのみ, もう片側を攻撃通信から抽出した入力ベクトルのみで学習させる. (ただし, 一度も勝者ユニットとなっていないユニットを発見した場合, そのユニットの参照ベクトルを入力ベクトルで置き換える.)

3.4 決定木と自己組織化マップの連結

決定木と自己組織化マップを図 1 のように連結させ一つのシステムとする. このシステムは次のような手順で動作する.

1. 通信データから特徴量を抽出し入力ベクトルを生成
2. 入力ベクトルを決定木に入れ, 結果を得る

3. 2 で得た結果を入力ベクトルの末尾に加え, SOM に入力する

4. SOM から結果を得て, システムの判定結果とする.

5. システム j の判定結果を定期的に決定木の教師データとしてフィードバックする.

決定木での判別は高速に処理されるため, 連結させても SOM 単体とほぼ変わらない速度で判別を行うことができる. 決定木は新しい教師データで再構築しない限り, 徐々に環境の変化から検出精度が落ちていくと考えられる. しかし,SOM 部分は最新のデータを分類しつつ学習を行うため, 次の決定木の再構成までに生じる精度低下の穴を埋めることができると期待される. 同時に,SOM 部分は決定木の結果を十分に反映しつつ補正を行うことで, 高い判定率を目指す. 一方, システムの判別結果を定期的に人間がレビューし, 決定木の教師データとしてフィードバックすることで持続的に安定した検知が行えることを期待している. これに対する検証は今後の課題とする。

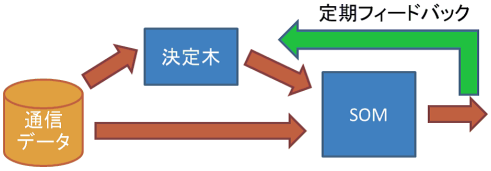

図 1: 決定木と自己組織化マップの連結

4 システム評価

4.1 実験結果

表 2 にシステムによる分類結果を示す. 教師データや初期化に使われた通信データはテストデータに含んでいない. 平均正答率はシステムが出した解答と, 入力データが元々含まれていた通信種別を比べ, 一致していた場合を正答として扱っている. パケット平均分類時間は C#の System.Diagnostics.Stopwatch クラスにより入力データ全てに対して解答が出るまでの時間を計測し, それを入力データ数で割ることにより算出している.

表 2: 分類結果

種別	平均正答率	パケット平均分類時間
攻撃	98.352 %	3.374ms
正常	97.280 %	3.015ms

また, 決定木と SOM を連結しても SOM 単体と動作速度がさほど変わらないかどうかを確かめるため, 決定木抜きで測定を行った. その結果, 連結時と見分けがつかない速度が出た. 実行環境のコンディションによっては決定木抜きのほうが遅くなることもあったため, 誤差の範囲といえるだろう. なお, 決定木単体で 1 万パケット分類させ, 1 パケット当たりの処理速度を求めたところ平均 0.0007412ms であった.

また, 決定木の性能でシステム全体の正答率がどれだけ変わるかを確かめるため, 決定木部分を, 任意確率で正答させることのできるダミープログラムに入れ替えて全体の正答率の計測を行った. (任意の確率で正答する決定木を作ることは困難なためダミーを使用している.) その結果が図 2 である.

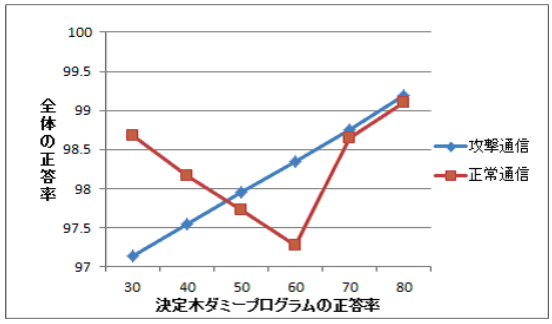

図 2: 決定木の精度向上のシステム全体への影響

4.2 考察

システム全体として 90 %後半の正答率を出しているが, 検知率 (攻撃通信に対する正答率) の他の論文 [5] などでも採用されている目標は 99 %, 誤検知 (正常通信を攻撃通信と判断してしまう確率) の目標は 0.01 %である [5]. そのた

め, どちらにもやや届いていない結果となっている.

図 2 の結果を見ると, 現時点での SOM と決定木のパラメータ設定の場合, 当初の予測とは異なり決定木が SOM の補助的な役割になってしまっていると言える. しかし, ほぼ無視できる分類時間の増加で数%成功率を引き上げることができうるため, 組み合わせる価値はあると考える.

性能を改善する以下のような幾つかの手法が考えられる。教師データ量を増やし決定木の正答率を上げることで全体の正答率を目標値に近づけることができると考えられる. また,SOM のパラメータ設定を見直すことも向上に繋がると期待される. 更に, SOM を逐次学習で最新の状態を学習させつつ, ある程度通信データが貯まったタイミングで決定木を再構成することで未知攻撃に対して持続的に検知性能を発揮できると考える.

5 おわりに

未知攻撃を含む攻撃通信の持続的な検知を目指した. 決定木と二次元自己組織化マップという複数の機械学習手法を組み合わせた検出方式を提案し, その性能を評価した. その結果,97 %以上の確率で攻撃通信と正常通信を検知できることがわかり, 検知率目標値に迫ることができた.

持続的な検知ができているかを確かめるために, CCC DATAset2010 等に含まれる過去の攻撃通信を教師データや SOM 初期化データとしシステムの初期設定を行い, そこから SOM の逐次学習と決定木の定期更新を行っていくことで 1 年後の CCC DATAset の通信も正確に分類できるようになるかを実験する必要がある. また, 検知率 99 %, 誤検知率 0.01 %を達成するために, 決定木や SOM のパラメータ等を改良していく. 加えて分類時間の短縮のために, 分類時間が増える大きな要因である SOM のユニット数を正答率を落とさずにどこまで減らすことができるかを実験する必要がある.

参考文献

[1] McAfee Labs, “2011 年第 1 四半期脅威レポート”, http://www.mcafee.com/japan/media/mcafeeb2b/international/japan/pdf/threatreport/threatreport11q1.pdf.

[2] 畑田充弘, 他 : “マルウェア対策のための研究用データセット ～MWS 2011 Datasets～”, MWS2011,（2011 年 10 月）.

[3] 柿本圭介, 田中英彦, “自己組織化マップを用いた異常検知についての一検討”, 情報科学技術フォーラム一般講演論文集 6(4), 79-80, 2007.

[4] T. コホネン, “自己組織化マップ”, シュプリガーフェアラーク東京, 2005.

[5] 山田 明, “ネットワーク侵入検知システムの高度化に関する研究”, 東北大学大学院情報科学研究科博士学位論文, 2009.

[6] Lincoln Laboratory, Massachusetts Institute of Technology, “ DARPA Intrusion Detection Data Sets ”, http://www.ll.mit.edu/mission/communications/ist/corpora/ ideval/data/.