Computer Security Symposium 2017
23 - 25 October 2017

侵入検知システムへのカーネル法を用いたオンライン学習手法の適用

高畑 孝輝[1,]　新美 礼彦[2,]

概要：ネットワーク機器を通過するパケットを監視し侵入検知が行われている．本研究では，カーネル法を用いたオンライン学習手法を提案する．提案手法は，学習のための事例データを入手した際に，事例データをこれまでに学習されたモデルに追加入力する形で学習を行う．また，学習手法にカーネル法を用いた学習を用いることにより，線形オンライン学習以上の分類精度が期待できる．実験では，線形オンライン学習によるモデルと，提案手法によるモデルの精度を比較することによって提案手法の有効性を示した．

キーワード：MWS，侵入検知，オンライン学習，カーネル法，ネットワークパケット

Applied Online Learning with Kernel for Intrusion Detection System

Kouki Takahata[1,]　Ayahiko Niimi[2,]

Abstract: Intrusion detection is performed by examining the packets that pass through network devices. Herein, we propose to apply a kernel-based online machine-learning method to intrusion detection. The proposed method uses additional packets when some additional packets passed after previous model is learned. We expect the proposed method, which uses kernel method, to yield more accurate results than the linear online learning method. In this study, we compared the accuracy of both the normal linear model and the proposed method, and discussed the effectiveness of the proposed method.

Keywords: Intrusion Detection, Online Learning, Kernel Method, Network Packet

1. はじめに

情報セキュリティの分野で，外部から企業や大学などの内部ネットワークへの不正アクセスが発生が問題視されている．そこで，ネットワークの通信から不正アクセスがされているかを検知する必要があるため，侵入検知システム（Intrusion Detection System: IDS）と呼ばれるネットワーク機器が導入されている [1][2]．侵入検知の実現方法は，過去に不正アクセスされた通信のブラックリストからチェックするマッチング手法や，普段のアクセス状況から極端に変化している状態を検知する異常検知手法など，複数提案されている．機械学習による侵入検知は過去の不正アクセスのログから汎化された不正アクセスのモデルを作成することで，過去の不正アクセスと違っていても，似たような動作をするアクセスを検知できるという利点がある．

侵入検知手法では機械学習の中のバッチ学習と呼ばれるデータ入力をまとめて行い学習する手法が用いられている．この手法では，新たな事例データを入手した際に，過去の事例データを含めて，全データを用いて学習をやり直さなければいけない．

新たな事例データを入手した際に，事例データをこれまでに学習されたモデルに追加入力する形で学習させることができれば，過去のデータを含めて学習する負担を軽減することができる．

本研究ではオンライン学習と呼ばれる手法を侵入検知

[1] 公立はこだて未来大学大学院　システム情報科学研究科
Graduate Scool of Systems Information Science, Future University Hakodate
[2] 公立はこだて未来大学 システム情報科学部
School of Systems Information Science, Future University Hakodate

に適用する．オンライン学習とは事例データを 1 つ 1 つを順番に入力し，分類モデルを事例データが 1 つ入力されるごとに更新していく学習手法の概念である．オンライン学習を用いることで逐次的に学習させることができる．本研究では，オンライン学習についての研究の中で高い分類精度が得られた SCW(Soft-Confidence-Weighted learning)[3] を用いる．ただし，SCW では線形モデルが使われている．線形モデルでの複雑な構造のデータでは線形分離できないため，精度が低くなる．精度を上げるためには，線形モデルではなく，非線形にデータ分離する必要がある．そこで本研究では，ネットワークパケットのようなデータにさまざまな種類を持つ複雑なデータ構造に対応するために，非線形モデルに対応したカーネル法と呼ばれる手法を用い，複雑な構造のデータを学習できるようにする．カーネル法とはデータの特徴量を高次元化し，その内積計算をカーネル関数に置き換えることにより，特徴量を増やしながらも計算速度を抑えるテクニックである．カーネル法により，もともと分離できなかったデータを高次元にすることで高次元空間上で分離可能になるため，精度の向上が期待できる．

評価実験では，オンライン学習手法である SCW と，既存のバッチ学習手法である SVM(Support Vector Machine) で，線形モデル，カーネルトリックを用いたモデルを用いて，侵入検知データを学習，分類を行う．SVM の精度を比較し，精度が保たれることを確認し，有用であることを確かめる。

1.1 章構成

2 章では関連研究について述べ，3 章では提案する手法と適用するアルゴリズムについて述べる．4 章では実験目的，実験対象，評価手法について述べ，5 章では実験結果を示す．結果を受けた考察と今後の展望について 6 章，7 章で記述する．

2. 関連研究

機械学習手法を利用した侵入検知とオンライン学習についての関連研究を示す．

2.1 侵入検知手法

機械学習手法の侵入検知への応用について記述する．

機械学習による侵入検知では，機械学習手法，扱うデータの種類，前処理などの工夫によりさまざまな研究がされている．機械学習手法にはランダムフォレスト，ナイーブベイズ，SVM などのさまざまな手法が用いられている [4][5][6]．また，パケットから得られる特徴量としては IP アドレスやポート番号，フラグなどのパケットが持つプロトコルごとのヘッダ情報やペイロードを用いることや，単方向の同一の IP アドレス，ポート番号をまとめてフローと定義し，フローごとの統計情報を特徴量として用いられる [7]．

SVM やランダムフォレストはデータをまとめて入力する形であるため，モデルを更新するためには過去のデータを再び入力し直すコストがかかる問題がある．また，ナイーブベイズでは逐次的にデータ更新をしやすい特徴があるが，複雑なモデルの対して十分な精度を得られていない [5]．

その中で，山内らはいくつかの種類のプロトコルで行われる C&C 通信の検出のために，TCP セッションの送受信のパケット数，バイト数，セッション時間を特徴量として 3 種類のバッチ学習による機械学習（SVM, ロジスティック回帰，ナイーブベイズ）によって分類精度の確認を行っている [8]．本研究では，学習アルゴリズムについての研究を行い，これらセキュリティの機械学習応用に関しての論文で使用される特徴量の中から山内ら [8] の提案する特徴量を使用して，SVM 手法と精度の面で比較を行う．

2.2 オンライン学習手法

オンライン学習とは過去の予測と新たに入力されたを用いて予測器を連続的に作る手法の枠組みである [9]。

オンライン学習についての研究は，データを 1 つずつしか利用できないという制約の元で，精度の向上，計算速度の向上，モデルの収束速度の向上という目標でモデルのパラメータ更新への工夫などの方法で改良がされてきた．Crammer による PA(Passive-Aggressive) は，データ入力前のモデルと入力後のモデル間のユークリッド距離を最小化する最適化問題を解くことにより精度の向上を行っている [10]．さらに，Dredze らの提案する CW(Confidence-Weighted learning) はモデルのパラメータの正規分布を導入し，PA と同様のアイデアで，学習前後の正規分布間のカルバックライブラー距離を最小化している．分散の値を小さくし，分散の大きさに対応してモデルの更新速度を調整することで，効率の良い学習と収束速度を実現している [11]．しかし，これら手法は，間違えた正解ラベルにノイズが有る場合に大幅にモデルを間違えた方向に更新してしまう問題がある．そこで，Wang らの提案する SCW(Soft-Confidence Weighted learning) では，更新する速度に制限を加えることによって，CW に大幅な更新が起きないように改良を行っている [3]．

それ以外に ALMA[12]，NHERD[13]，AROW[14] と呼ばれるアルゴリズムも提案されている．このように，オンライン学習手法の研究としてさまざまな提案がされているが，Hoi らはそれらアルゴリズムを実装，比較を行い [15]，他手法と比較した際に，SCW を用いた場合，より高い精度を得ているため，本研究では，SCW を用いた分類を行う．

また，これら手法はモデルとして線形モデルを想定しているため，線形分離できないような複雑なデータに対して有効でない．そこで本研究では SCW を元にカーネル法を

適用することで，モデルの複雑化による精度向上を行う．

オンライン学習の非線形化についての研究はすでに行われている．Kivinen らは単純パーセプトロンのオンライン学習に対してカーネルトリックを行うことを試みている [16]．この手法ではカーネル法を使用するために内積計算を用いたモデルである $\sum_i^N \alpha K(\boldsymbol{x}_i \cdot \boldsymbol{x})$ をパーセプトロンにそのまま適用した手法となっており，マージンを大きくするなどの工夫がされておらず，精度が低くなることと考えられる．本研究ではこの手法をもとに SCW に対してこのモデルを学習させることにより，マージンを大きくすることなどの工夫を行うことによって精度向上させることを考える．

3. 提案手法

本研究では侵入検知問題にカーネル法を適用したオンライン学習手法を用いたシステムの提案を行う．提案手法の概要と，学習アルゴリズムについて記述する．

3.1 提案手法の概要

オンライン学習を用いたシステムの全体像を図 1 に示す．

入力されるデータは学習のためのパケットデータと，未知のパケットデータに分けられる．学習のためのパケットデータにはは，パターンマッチングで検出されたパケットデータや別ネットワークで得られた攻撃データを想定し，正常データには侵入検知を行うネットワーク内のパケットデータを用いることを想定している．未知のデータは検知システムを使用するイントラネットで入力されたパケットデータを用いる．実用上ではモデルの学習（図 1 の図中①）でモデルを更新しながら，未知データの識別（図 1 の図中②）を行うという流れで運用を行う．

実験上の手法の流れを述べる．学習の段階，識別の段階を分けると図 2，図 3 のようになる．学習データ，未知データそれぞれのパケットに関して，パケットから特徴量の抽出を行う．学習データと未知データの両方において特徴量を抽出する処理を行う．

まず，学習データを用いて学習を行う（図中①）．２クラス分類を行うために攻撃パケットと正常パケットにラベルを付記し，取り出した特徴量とともに 1 セッションずつ学習を行う．

次に未知データを与える（図中②）．未知データに対して学習フェーズで学習したモデルを用いて攻撃データか正常データかを識別する．識別することによりネットワークに攻撃が来たかどうかを判断する．

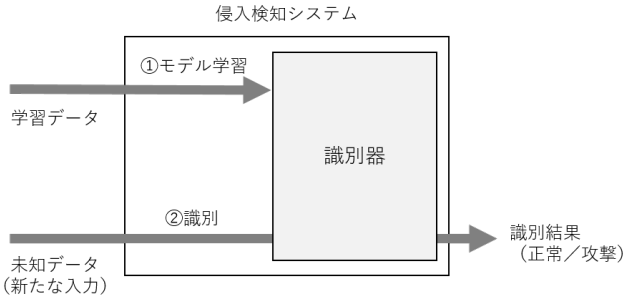

図 1 提案システムの全体像

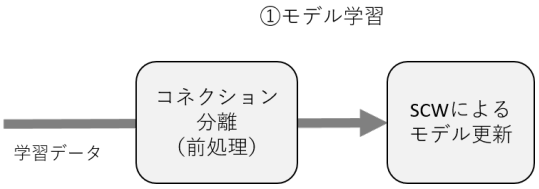

図 2 提案手法の流れ（モデルの学習）

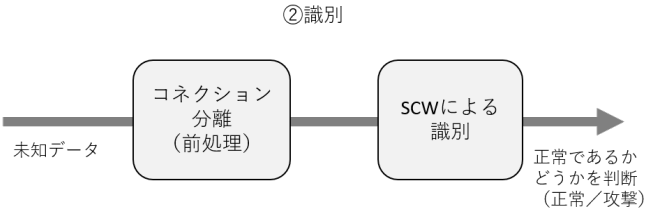

図 3 提案手法の流れ（未知データの識別）

3.2 学習アルゴリズム

提案手法の学習にはオンライン学習の一手法である SCW を用いて識別を行う．SCW では線形モデルである $y = \sum_i^N \boldsymbol{w}^T \boldsymbol{x}$ のパラメータ $\boldsymbol{w}$ が正規分布に従っていると考え，モデルは重みの平均 $\boldsymbol{\mu}$ を利用して，

$$y = \sum_i^N \boldsymbol{\mu}^T \boldsymbol{x} \tag{1}$$

とする．SCW では以下の最適化問題により学習を行う．

$$(\boldsymbol{\mu}^{(t+1)}, \Sigma^{(t+1)}) = \arg\min_{\boldsymbol{\mu},\boldsymbol{\Sigma}} D_{KL}(N(\boldsymbol{\mu},\boldsymbol{\Sigma}) \| N(\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)})) + Cl^{\phi}(\boldsymbol{\mu},\boldsymbol{\Sigma},\boldsymbol{x}^{(t)}, y^{(t)}) \tag{2}$$

Σ は，共分散行列で，D_{KL} はカルバックライブラー距離である．カルバックライブラー距離とは，確率分布の距離を測るために設計された距離である．また，第 2 項は制約項であり，入力された後が正しく分類されることを示している．この問題を解くと，平均，分散の更新式は

$$\boldsymbol{\mu}^{(t+1)} = \boldsymbol{\mu}^{(t)} + \alpha^{(t)} y^{(t)} \boldsymbol{\Sigma}^{(t)} \boldsymbol{x}^{(t)} \quad (3)$$

$$\boldsymbol{\Sigma}^{(t+1)} = \boldsymbol{\Sigma}^{(t)} - \beta^{(t)} \boldsymbol{\Sigma}^{(t)} \boldsymbol{x}^T \boldsymbol{x}^{(t)} \boldsymbol{\Sigma}^{(t)} \quad (4)$$

となる．このときの α，β は，

$$\alpha = \max\left(0, \frac{1}{v_t \zeta}\left(-m_t \psi + \sqrt{m_t^2 \frac{\phi^4}{4} + v_t \phi^2 \zeta}\right)\right) \quad (5)$$

$$\beta = \frac{\alpha_t \phi}{\sqrt{u_t} + v_t \alpha_t \phi} \quad (6)$$

となる．

次にカーネル法を適用する．カーネル法を使用するためには，内積計算によるモデルを作成する必要があるため，式 (1) のモデルを過去のデータの内積の総和の式に置き換える必要があるため，

$$y = \sum_{i=1}^{N} \alpha K(\boldsymbol{x}_i, \boldsymbol{x}) \quad (7)$$

をモデルとして SCW による学習を行う．この式を当てはめると，

$$\boldsymbol{\mu}_i = \sum_{p}^{i-1} \nu_p^{(i)} \boldsymbol{x}_p \quad (8)$$

$$\Sigma_i = \sum_{p,q=1}^{i-1} \pi_{p,q}^{(i)} \boldsymbol{x}_p \boldsymbol{x}_q^T + aI \quad (9)$$

ここで，ν, $\pi_{p,q}$ の更新式は，

$$\nu_i^{(i)} = 1 \quad (10)$$

$$\nu_p^{(i+1)} = \nu_p^{(i)} + \alpha_i y_i \sum_{q}^{(i-1)} \pi_{p,q}^{(i)} \boldsymbol{x}^T \boldsymbol{x}_i \quad (11)$$

$$\pi_{p,q}^{(i+1)} = -\beta_i \sum_{r,s} \pi_{p,r}^{(} i) \pi_{s,q}^{(i)} \boldsymbol{x}_r^T \boldsymbol{x}_s + \pi_{p,q}^{(i)} \quad (12)$$

$$\pi_{p,i}^{(i)} = \pi_{i,p}^{(i)} = -\beta_i \sum_{p,r}^{(i-1)} \pi_{p,r}^{(i)} (\boldsymbol{x}_r^T \boldsymbol{x}_i) \quad (13)$$

$$\pi_{i,i}^{(i+1)} = -\beta_i \quad (14)$$

ここで，π の更新式に着目すると，データを入力し更新を行う度に，$\pi_{p,q}$ は，データ数×データ数の計算を行っていることがわかる．これを全ての π に対して更新を行うので，更新のたびにデータ数の三乗計算しなければならず，データ数に応じて計算量が膨大になる．同様にパラメータに正規分布を導入した CW では，CW の近似として，共分散行列 Σ の非対角成分を 0 とした対角行列にする手法を提案している．そこで，計算速度を向上させるために非対角成分を削除する．削除することにより，対角成分である $\pi_{i,i}$ 以外の成分が削除される．

SCW，カーネル法，対角成分の削除を擬似コードとしてまとめると Algorithm1 のようになる．

Algorithm 1 SCW with Kernel

Inputs:
  parameters $C > 0, \eta > 0$
Initialize:
  $\boldsymbol{\mu}_0 = (0, ..., 0)^T, \Sigma_0 = I$
for $i = 1, ...T$ **do**
  入力 $\boldsymbol{x}_t \in \mathbb{R}^d$
  識別 $\hat{y}_t = \sum_p^{i-1} \nu_p^{(i)} \boldsymbol{x}_p \cdot \boldsymbol{x}$
  教師 y_t
  損失 $l^\phi(N(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1}); (\boldsymbol{x}_t, y_t))$
  if $l^\phi(N(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1}); (\boldsymbol{x}_t, y_t)) > 0$ **then**
    更新
    $\nu_i^{(i)} = 1$
    $\nu_p^{(i+1)} = \nu_p^{(i)} + \alpha_i y_i \sum_q^{(i-1)} \pi_{p,q}^{(i)} \boldsymbol{x}^T \boldsymbol{x}_i$
    $\pi_{p,p}^{(i+1)} = -\beta_i \sum_{r,s} (\pi_{p,p}^{(i)})^2 \|\boldsymbol{x}_s\|^2 + \pi_{p,p}^{(i)}$
    $\pi_{p,q}^{(i+1)} = 0$ for $p \neq q$
    $\pi_{i,i}^{(i+1)} = -\beta_i$
  end if
end for

4. 実験

実験目的，実験対象，前処理方法，使用する特徴量，評価手法について記述する．

4.1 実験目的

実験目的は，カーネル法を用いたオンライン学習手法と既存のバッチ学習手法を精度で比較し，バッチ学習をベースとしてオンライン学習の精度がバッチ学習と同程度になるかどうかを検証することである．また，線形モデルでの精度とカーネル法の精度を比較することでカーネル法の有効性を評価する．

本実験ではバッチ学習である SVM とオンライン学習である SCW のそれぞれで攻撃通信と異常通信の学習・識別を行い，精度を比較する．既存研究 [8] において，TCP コネクション中の特徴量を利用した学習を SVM，ナイーブベイズ，ロジスティック回帰で比較した結果， SVM が最も精度が高くなっていることから，SVM と SCW の比較を行う．TCP コネクションとは同一のクライアントとなる PC とサーバの間で行われる通信で接続開始してから接続終了するまでの一連の流れを指す．

4.2 実験対象

提案手法による侵入検知は正常通信と攻撃通信を分離することであるため，その両方の通信ログが必要である．攻撃通信としてマルウェア解析技術の研究を目的に MWS（anti-Malware engineering WorkShop）が提供する CCC DATAset2011 を用いる [17]．このデータセットは一定期間中のハニーポットへの攻撃を記録したものであり，実際に行われた生の通信データに加え，どの通信が悪意あるソフトウェアによって行われたのかを記録している．正常通信には学内上のネットワークが安全であると仮定して用い

る．正常通信の枠組みは，正常通信としてよく使われる通信である Web サイトの閲覧，メールの送受信，音声通話などの通信ログを Wireshark[18] というアプリケーションを用い，一定時間収集した．

4.2.1 攻撃データ

攻撃データとして MWS が提供している CCC DATAset を用いた．MWS とはマルウェア対策研究人材育成ワークショップであり，研究を行う上で単独や小規模ではマルウェアの動作ログや検体の収集が難しいことや，研究によって使用されるデータが違うという問題点を解決するために企業や研究所が収集したデータを共通データとしてワークショップ参加者に提供している．また，CCC DATAset とは MWS の一環として CCC（Cyber Clean Center）がマルウェア解析技術の研究を目的に提供しているデータセットである．マルウェア収集のためにハニーポットを設置し，特定期間にハニーポットにより収集されたマルウェアのデータが提供される．このデータセットは「マルウェア検体」「攻撃通信データ」「攻撃元データ」の 3 つのデータで構成されている．

マルウェア検体はハニーポットが収集したマルウェア検体をハッシュ値である．

攻撃通信データはハニーポットの通信をハニーポットが動作する OS 上の tcpdump でパケットキャプチャしたファイルである．ハニーポットは Windows XP で動作し，データを 2010 年 8 月 18 日から 8 月 31 日，2011 年 1 月 18 日から 1 月 31 日の期間中に収集する．

攻撃元データは攻撃通信データの収集時期を含む 2010 年 5 月 1 日から 2011 年 1 月 31 日までの期間にハニーポットが記録したマルウェア取得時のログである．攻撃元データには表 1 のようなログが記録されている．本実験では攻撃通信データを攻撃元データの情報を基にフィルタリングし，攻撃データを作成した．

表 1 CCC DATAset の攻撃ログに含まれるデータ

項目
マルウェア検体の取得時刻
送信元 IP アドレス
送信元ポート番号
宛先 IP アドレス
宛先ポート番号
TCP, UDP
マルウェア検体の取得時刻ハッシュ値
ウィルス名称
ファイル名

4.2.2 正常データ

正常データについては大学内環境で正常である通信を想定して表 2 の環境下で取得した．

正常データは関連研究 [19] を参考にした．関連研究ではオンラインゲーム，BitTorrent，MSN Messenger などの使用が確認されたが現在使われていないものを現在多く使われているアプリケーションに置き換え，業務などで使われない P2P やゲームの通信を除外した．

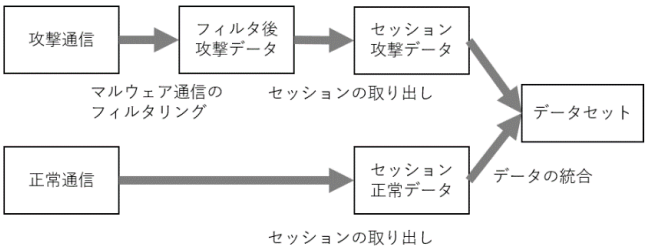

図 4 前処理の手順

表 3 サービスに使用した通信を示す．サービスの使用前にキャプチャを開始しサービス使用終了後にキャプチャを止めるという方法で記録を行い，pcap 形式でファイルを出力した．Line のチャット機能と Skype の音声通話についてはプロトコルの詳細は非公開であったが，プロトコルは 443 番を使用していることを確認した．また，正常及び攻撃データの量については，後述する前処理後のコネクションの数は表 4 のようなサイズになっている．

表 2 正常データ取得環境

項目	内容
OS	Windows XP Professional (32bit)
使用ソフト	Wireshark 1.12.13

4.3 データの前処理

データを分類するためには，元の通信データから分類に必要な特徴量を取り出す前処理の工程が必要である．データ前処理は攻撃通信，正常通信に対して，図 4.3 のように行う．まず，CCC DATAset の生の通信データから，悪意あるソフトウェアの通信ログにより，攻撃通信をフィルタリングする．その後，正常通信と攻撃通信について TCP コネクションごとに分割，特徴量抽出を行い，それぞれのデータを統合する．TCP コネクションごとの分割には Wireshark を用い，分割されたデータから Python ライブラリである Pandas0.18.1 を用いて集計を行う．

4.4 特徴量

TCP コネクションから 5 つの特徴量を使用する．山内らは独自の特徴量として TCP コネクション中のセッション時間，送受信バイト数，送受信パケット量を設計し，SVM，ロジスティック回帰，ナイーブベイズにより分類を行い，SVM の有効性を示している [8]．そこで本実験では，山内らの提案する特徴量を用いて SVM とオンライン学習手法の比較を行う．

表 3 正常データの使用通信

サービス	プロトコル	ポート番号
FTP による送受信	FTP	23
メールの送受信	SMTP over SSL	465
SSH によるコマンド	SSH	22
Web ブラウザ	HTTP	80, 443
ファイルダウンロード	HTTP	80
Skype 音声通話	非公開 (TLS)	443
Line によるチャット	非公開 (TLS)	443
音声ストリーミング	RTMP	1935
動画ストリーミング	HTTP	80, 443

表 4 正常通信と攻撃通信のデータ量

	コネクション数
正常通信	4007
攻撃通信	3511
合計	7518

4.5 実験・評価手順

正しく分類できるか評価するため，各アルゴリズムで前処理を行った上記 5 つの特徴量に対して学習，予測を行う．評価指標として精度 (Accuracy)，適合率 (Precision)，再現率 (Recall) を測定する．精度とは全データ中で正しく分類できている割合，適合率は不正な通信と判断したデータ中で本当に不正な通信である割合，再現率は本当に不正な通信の中で不正な通信であると分類した割合を示している．

提案手法では，カーネル法をオンライン学習に適用しているが，カーネル関数については限定していない．そこで，実験では，汎用的な分類問題に使用されるガウシアンカーネルを用いて実験を行う．ガウシアンカーネルは

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = -\gamma \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2 \tag{15}$$

となり，1 つのパラメータ γ により計算することができる．

また，学習を行うために学習パラメータを設定する必要がある．予測結果についてはより精度が高くなるパラメータを適切に設定するために 3-Fold 交差検証を行う．SVM はソフトマージン SVM を用い，パラメータである C を C={10.0, 100.0, 1000.0}，に設定し，SCW はパラメータである η,C をそれぞれ、η={1.0, 10.0}, C={1.0, 10.0, 100.0, 1000.0}，に事前に設定し，それぞれの学習アルゴリズムに対してカーネル法を適用し，ガウシアンカーネルのパラメータとして，γ={0.0001, 0.001, 0.001} を設定し，これらの組み合わせによって交差検証を行った．これらのパラメータについては，他パラメータについても試したが，精度が大きくなるようなパラメータに絞り，全組み合わせを検証した．

プログラムの実装については，SVM は Python のライブラリである sckit-learn 0.17.1 を用い，SCW は既存のライブラリがなかったため，Python 3.6.0, Numpy1.11.0, Scipy 0.17.1 を用いて自作した．これらのプログラムを Amazon が提供する Amazon EC2　インスタンス m4.large 上で実行した．

5. 結果

SVM と SCW でパラメータを最適化した際の精度，適合率，再現率を，表 5 に示す．

表 5 正常通信と攻撃通信の割合

手法	精度	適合率	再現率
SCW(ガウシアンカーネル)	0.9367	0.9454	0.9361
線形 SCW	0.7626	0.8169	0.7362
SVM(ガウシアンカーネル)	0.9042	0.9307	0.8941
線形 SVM	0.7256	0.7356	0.7485

SCW（ガウシアンカーネル）では最も精度が高くなっている．線形モデルの SVM と線形モデルの SCW を比較すると線形モデルの SCW の精度が高くなっている．SCW（ガウシアンカーネル）では線形 SCW より精度が高くなった．

また SCW（ガウシアンカーネル）パラメータについては、$\gamma = 0.001$ の場合に，他のパラメータに関係なく高い精度，適合率，再現率を得ている．

実行時間については 3-Fold 交差検証の経過時間は学習時間，識別時間を合計して 18 分程度であった．

6. 考察

SVM の実験結果から SVM（ガウシアンカーネル）ではパラメータの調整により 9 割程度の精度，適合率，再現率で識別していることを確認したことに対し，線形 SVM では精度，適合率が 7 割程度の識別結果となった．

線形 SCW の結果では 7 割から 8 割の精度で検出していることがわかり，SCW(ガウシアンカーネル) では 9 割程度の精度，適合率，再現率で識別していることを確認した．また，適合率と再現率には大きな偏りは見られなかった．

SVM での実験において，山内らの研究では 99%の精度で分類を行っている [8] が，データの正規化を行っていることや，もともとの正常データの取得方法，攻撃データの抽出方法に差があるために識別精度に差が出ていると考える．さらに，本実験ではマルウェアのログデータからフィルタリングにより全データを抽出しているため，精度の低下が発生していると考える．

SVM はカーネル法により非線形の分離を可能にすることで精度が大幅に向上している．SVM(ガウシアンカーネル) と線形 SCW を比較すると線形 SCW では SVM(ガウシアンカーネル) ほどの精度が得られていないことがわかる．しかし，線形 SCW にカーネル法を適用した場合は，精度が向上し，SVM の精度と同程度の精度，適合率，再現率で識別することが可能になった．カーネル法を用いることで精度の向上を考えることは有効であり，バッチ学習による精度に近づけることができるとわかった．

実験ではオンライン学習を使用することによる精度の検

討について行ったが，時間計測，メモリ消費量の計測は行っていない．今後の課題として，オンライン学習の利点を証明するために時間計測を行い比較することが必要となる．ただし，アルゴリズムの特性やオンライン学習の特徴から線形 SCW は線形 SVM，SVM（ガウシアンカーネル）と比較して，学習速度が速くなり，メモリ消費量が少なくなることが予想される．しかし，実行時間については SCW にカーネル法を適用する場合，SVM(ガウシアンカーネル) と比較しても計算量は大きくなる．カーネル法は過去に入力のデータとの内積によって値を計算することや，過去のデータ分パラメータ更新に時間が掛かるからである．実験環境で Python によるコードを実行すると，18 分かかった．

そこで，過去のデータを統合することや，同一計算の保持などの工夫が必要になると考えられ，今後の課題となる．また，実験ではカーネル関数としてガウシアンカーネルを用いたが，カーネル関数を設計することで計算量を抑えることができる可能性もあるため，今後の展望となっている．

それら工夫により高速化を実現できれば，攻撃通信の検知システムなどを企業や学校に導入し，機械学習手法の利点である未知のデータの分類を実現するコストが下げることができると考える．

7. まとめと今後の展望

侵入を検知する手段はいくつか考えられるが，本研究では通信をする際にネットワークを通るパケットからマルウェアを検知する問題に取り組んだ．

検知する手法として機械学習手法の中のカーネル法を用いたオンライン学習を用いることを提案した．それにより，新たな事例データを入手した際に，事例データをこれまでに学習されたモデルに追加入力する形で学習させることができれば，過去のデータを含めて学習する負担を軽減することができる．オンライン学習手法には SCW を用い，カーネル法を適用した．

実験では，SVM と SCW の比較を行った．攻撃データには CCC DATAset2011 から抽出したものを使用し，正常データには学内ネットワークで取得したパケットを利用し，交差検証を行い精度を算出した．

実験の結果より SCW は SVM と同程度の精度，適合率，再現率で分類することができたため，十分な精度を得られる結果になったと考える．

今後の展望として計算コストがある．カーネル法は過去のデータとの内積によるモデルが使用されるため，データ数をより大規模にした際に大きな計算コストが掛かる．そこで，過去のデータの統合やデータを保存しない手法などにより計算を単純化する必要があると考える．

また，カーネル関数としてガウシアンカーネルを用いたが，それらの工夫は，オンライン学習に特化したカーネル関数を定義することによって，改善できると考えられるため，今後の展望として，カーネル関数を適切に設計することなどが挙げられる．

謝辞 マルウェア解析技術の研究を目的に CCC(サイバークリーンセンター) が提供する CCC DATAset2011 を活用しました．貴重なデータを提供いただきありがとうございました．

参考文献

[1] Snort: Snort - Network Intrusion Detection & Prevention System, (online), available from ⟨https://www.snort.org/⟩ (accessed 2017-05-09).

[2] 株式会社日立ソリューションズ：次世代不正侵入検知/防御（IDS/IPS）アプライアンス IBM Security Network Protection XGS ｜日立ソリューションズ『IBM Security Network Protection XGS』のシステム、サービス概要・価格や、解決出来る課題をご紹介，（オンライン），入手先 ⟨http://www.hitachi-solutions.co.jp/ips_xgs/⟩（参照 2017-05-11）．

[3] Hoi, S. C. H., Wang, J. and Zhao, P.: Exact Soft Confidence-Weighted Learning., *ICML*, icml.cc / Omnipress (2012).

[4] Bilge, L., Balzarotti, D., Robertson, W., Kirda, E. and Kruegel, C.: Disclosure: Detecting Botnet Command and Control Servers Through Large-scale NetFlow Analysis, *Proceedings of the 28th Annual Computer Security Applications Conference*, ACSAC '12, New York, NY, USA, ACM, pp. 129–138 (online), DOI: 10.1145/2420950.2420969 (2012).

[5] Panda, M.: NETWORK INTRUSION DETECTION USING NAIVE BAYES, *IJCSNS International Journal of Computer Science and Network Security*, Vol. 7, pp. 258–263 (2007).

[6] Li, Y., Xia, J., Zhang, S., Yan, J., Ai, X. and Dai, K.: An efficient intrusion detection system based on support vector machines and gradually feature removal method, *Expert Systems with Applications*, Vol. 39, No. 1, pp. 424–430 (online), DOI: 10.1016/j.eswa.2011.07.032 (2012).

[7] Buczak, A. L. and Guven, E.: A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection, *IEEE Communications Surveys Tutorials*, Vol. 18, No. 2, pp. 1153–1176 (online), DOI: 10.1109/COMST.2015.2494502 (2016).

[8] 山内 一将，川本 淳平，堀 良彰，櫻井 幸一：C&C トラフィック分類のための機械学習手法の評価，情報処理学会論文誌，Vol. 56, No. 9, pp. 1745–1753 (2015).

[9] Shalev-Shwartz, S.: Online Learning and Online Convex Optimization, *Found. Trends Mach. Learn.*, Vol. 4, No. 2, pp. 107–194 (online), DOI: 10.1561/2200000018 (2012).

[10] Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S. and Singer, Y.: Online Passive-Aggressive Algorithms, *Journal of Machine Learning Research*, Vol. 7, pp. 551–585 (2006).

[11] Dredze, M., Crammer, K. and Pereira, F.: Confidence-weighted Linear Classification, *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, New York, NY, USA, pp. 264–271 (online), DOI: 10.1145/1390156.1390190 (2008).

[12] Gentile, C.: A New Approximate Maximal Margin Classification Algorithm, *Journal of Machine Learning Research*, Vol. 2, No. Dec, pp. 213–242 (online), available from ⟨http://www.jmlr.org/papers/v2/gentile01a.html⟩

(2001).
[13] Crammer, K. and Lee, D. D.: Learning via Gaussian Herding, pp. 451–459 (online), available from ⟨http://papers.nips.cc/paper/3893-learning-via-gaussian-herding.pdf⟩ (2010).
[14] Crammer, K., Kulesza, A. and Dredze, M.: Adaptive Regularization of Weight Vectors, *Advances in Neural Information Processing Systems 22* (Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I. and Culotta, A., eds.), Curran Associates, Inc., pp. 414–422 (online), available from ⟨http://papers.nips.cc/paper/3848-adaptive-regularization-of-weight-vectors.pdf⟩ (2009).
[15] Hoi, S. C., Wang, J. and Zhao, P.: LIBOL: A Library for Online Learning Algorithms, *Journal of Machine Learning Research*, Vol. 15, pp. 495–499 (online), available from ⟨http://jmlr.org/papers/v15/hoi14a.html⟩ (2014).
[16] Kivinen, J., Smola, A. J. and Williamson, R. C.: Online Learning with Kernels, *Advances in Neural Information Processing Systems 14* (Dietterich, T. G., Becker, S. and Ghahramani, Z., eds.), MIT Press, pp. 785–792 (2002).
[17] 畑田 充弘，中津留 勇，秋山 満昭：マルウェア対策のための研究用データセット 〜MWS 2011 Datasets 〜，コンピュータセキュリティシンポジウム 2011 論文集，Vol. 2011, No. 3, pp. 1–5 (2011).
[18] Wireshark: Wireshark Go Deep, (online), available from ⟨https://www.wireshark.org/⟩ (accessed 2017-08-04).
[19] 市田達也：特徴量の時間的な状態遷移を考慮したマルウェア感染検知手法に関する研究，修士論文，早稲田大学理工学術院基幹理工学研究科 修士論文 (2011).