深層学習を用いた多段 NIDS の提案

竹下　健太郎 [1]　原山　美知子 [2]

概要：近年のネットワーク利用の浸透に加え，新型コロナウイルスの影響により，在宅勤務，通信販売，遠隔会議など，ネットワークを使用する頻度や場所が拡大している．その中で，マルウェアの対策が急務となっている．その一つとして不正侵入検知システム(Intrusion Detection System, IDS)があげられる．現在使用されている IDS ではパターンファイルに登録されている不正通信の特徴と一致した場合に検知するシグネチャマッチング型識別器が主に使われているが，この識別方法では未知の攻撃に対応することができない．

そこで本研究では，深層学習の手法を用い，正当通信の特徴を学習させたアノマリ型識別器で不正通信を検知したあと，シグネチャマッチング型識別器でクラス別検知を行うという多段のネットワーク型 IDS(NIDS)を提案する．深層学習のアノマリ型識別器として SAE(Sparse AutoEncoder)，シグネチャマッチング型識別器として MLP(Multi Layer Perceptron)を用いる．実験では， CICIDS2017 データセットを使用し，検知精度を評価した．ここでは提案システムの構成と評価の結果を報告する．

キーワード：NIDS，SAE，MLP, アノマリ検知, CICIDS 2017

Proposal of Multi-Stage NIDS using Deep Learning

Kentaro Takeshita[1]　Michiko Harayama[2]

In recent years, the use of networks has become widespread, and due to the influence of the new coronavirus, remote works have increased. In this situation, countermeasures against malware like as the Intrusion Detection System (IDS) are urgently needed. Although currently used IDSs mainly use signature matching type discriminators, they cannot deal with unknown attacks.

Therefore, in this study, we use a deep learning method to detect unauthorized communication with an anomaly classifier that has learned the characteristics of authorized communication, and then detect each class with a signature matching classifier. NIDS) is proposed. SAE (Sparse Auto Encoder), one of the Neural Networks, was used as an anomaly type classifier, and MLP (Multi-Layer Perceptron) was used as a signature matching type classifier. The data set was CICIDS2017, and the detection accuracy was evaluated. Here, we report the configuration of the proposed system and the evaluation results.

Keywords: NIDS, SAE, MLP, Anomaly detection, CICIDS 2017

1. はじめに

2020 年初めに始まった新型コロナウィルス COVID-19 によるパンデミックは拡大の一途を辿り，未だ終息の見通しは立っていない．新型コロナウィルスの流行の中で感染を防止しつつ社会活動の継続していくためには，情報インフラであるネットワークを最大限に活用することが不可欠である．実際，在宅勤務，通信販売，遠隔会議など，ネットワークの利用が急加速している．それに伴い，情報セキュリティの重要性が高まっていることはいうまでもない．また，マルウェアやサイバー攻撃の手法は技術進歩とともに高度化しているため，対策や防止策を常に見直し改善していく必要がある．

主要なセキュリティ対策の一つに不正侵入検知システム(Network Intrusion Detection System, IDS)が挙げられる．本研究ではネットワーク型 IDS（NIDS）の検知システムに注目する．従来の検知手法であるシグネチャマッチングでは，知られている不正通信の特徴をパターンファイルに登録しておき，ネットワークに流入するパケットフローの特徴量を照合して不正通信を検知する．しかし，通信回線の容量が大きくなり，シグネチャ数が膨大になると検知に時間がかかる．また，日々，新しい不正通信が生まれているため，未知の不正通信を検知する必要性も高まっている．さらに NIDS の検知結果をファイヤウォールと連携して不正通信を遮断する不正侵入防止システム（Intrusion Prevention System, IPS）として運用するには，高速な検知とともに，正当な通信の誤検知による阻害を防ぐため高い検知精度が求められる．そのため検知手法の模索が行われている．

ここ数年，機械学習および深層学習の研究が進み，識別手法が充実してきた．そこで，最近では，これらの手法を IDS に適用することによる不正通信の検知精度を向上が試みられている[1,2]．また，マルウェア，不正通信，サイバ

1 岐阜大学大学院自然科学技術研究科
Dept. Intelligence Science and Engineering, Grad. Sch. Natural Science and Technology, Gifu University

2 岐阜大学工学部電気電子・情報工学科
Dept. Electric Electronics and Informatics, Engineering Faculty, Gifu University

ー攻撃に関するベンチマークデータセットも公開されている[3-6]．筆者らも機械学習および深層学習手法を用いたIDSの研究を行ってきた．上野ら[7]の研究では，テストデータセット NSL-KDD[4] を使用して，3-Layer Neural Network (3NN) , Self-Taught Learning (STL), RNN (Recurrent NN), LSTM (Long-Short Term Memory)による多値分類を行なった．また，学習における重みの更新方式としてStochastic Gradient Descent(SGD)およびAdam[8] を比較した．その結果，RNNの検知精度が比較的高く，Adamの使用により全体的に検知精度が向上すること，LSTMは学習時間がかかることなどを示した．また，前田ら[9]の研究では, HTTP型ボットネット通信の検知に焦点をあて， MWSデータセット[6]を用いて Correlation-Graph Convolution Neural Network (C-GraphCNN) による検知を試みた． 22のフロー特徴量について，平均値の差，3NN，ランダムフォレスト，ラッソ回帰の４手法により特徴量の寄与度を解析し，特徴量が検知精度と学習時間に及ぼす効果を調べた．その結果，C-GraphCNN を用いた場合，フロー特徴量の寄与度に対して検知精度の依存は少なく特徴量数が多いほど検知精度が上がることが示された．

最近注目されている深層学習の手法の一つに Auto Encoder (AE) がある．AEを IDSに適用することにより未知の不正通信を検知することができる[1,2]．さらに，AEに正則化項を追加して識別精度を高めた Sparse Auto Encoder (SAE)[10]も提案されている．しかし，AE は二値分類器であるため，不正通信の種類を特定することはできない．そこで，本研究では，SAE と多層の Multi-Layer Perceptron（MLP）による２段構成の検知システムを提案する．これにより，誤検知を低減，未知の不正通信の検知，さらに不正通信の種類の特定を図る．ここでは，提案システムの概要を述べるとともに，ベンチマークテストデータCICIDS2017[5]を用いて，検知精度および学習時間，検知時間を計測した結果を報告する．

2. 関連研究

IDS の研究に関しては，検知アルゴリズムに関するものが多く，特にここ数年は機械学習アルゴリズムを使用したものが特に多い．Alshamy らのレビュー[1]では，2017 年以降で 69 件の論文が紹介されている．その中で用いられた機械学習アルゴリズムは，Support Vector Machine (SVM), Logistic regression (LR), Random Forest (RF) , K Nearest Neighbor (k-NN)などである．SVM や LR は検知精度が高いが二分類に限定される．RN はいくつかの改善によって検知精度が向上するが計算コストは高くなる．k-NN は学習データ数の増加するにつれて検出に時間がかかることなど，各アルゴリズムの長短がまとめられている， Neural Networks (NN) に関しても言及があり，特別な前提知識なしに検知ができる利点およびオーバーフィッティングの問題なども指摘されている．

しかし，ここ数年，検知アルゴリズムとして深層学習を用いた IDS の研究も増え，Kalimuthan らのレビュー[2]では 12 報の研究が紹介されている．用いられている Deep Neural Network (DNN)のタイプは，AE, MLP, STL, LSTM, RNN, CNN などである．不正通信を検知する場合，多くは二値分類を用いる．既知の不正通信を訓練データとして識別器を構成するが，AE や STL を用いると教師なし学習で識別器を生成することができる．一方，教師あり学習の MLP, LSTM, RNN, CNN では，出力ノード数を増やすことによって多値分類を行い，不正通信の種類を判別することもできるが，検知率や識別精度は十分高いとはいえない．

これらの研究で用いられているベンチマークテストデータとしては, KDD99, NSL-KDD, KYOTO 2006 +, ISCX2012, UNSW-NB 15，CICIDS2017，CIDDS-001 および CSE-CIC-IDS2018 が挙げられるが， KDD99，NSL-KDD を用いた研究が多い．しかし，KDD99[3]は，1999 年の KDD Cup 1999 で公開, NSL-KDD[4]はその後継として 2009 年に公開されたデータであり両方とも新しいデータとはいえない．サイバー攻撃のフロー特徴も年々変化していると考えられるため，なるべく新しいテストデータを使いたいところであるが必ずしも攻撃の種類をカバーしているわけではない．現状の公開データの中では，CICIDS2017[5]が比較的新しく，含まれている攻撃種類も多い．

[1]で紹介されている CICIDS2017 を用いた最近の研究では, RN で多値分類を行った例が報告されている．また， k-NN を用いた例では検知精度が高いが計算時間が 1,784 s と大きい．その他，C5.0, SVM, RF, NB を用いて DDoS 攻撃の検知を行った研究などがある．CICIDS2017 で深層学習を用いた例では 5 層 NN を用いた[11]や MLP と RN を組み合わせた報告[12]がある．

3. 提案システム

3.1 提案システムの概要

今回提案するシステムの概要を図 1 に示す．まず，SAE により正当データと異なった特徴をもっているデータを抽出する．抽出されたデータを，MLP を使用して不正通信のクラス別に分類する．

3.2 SAE によるアノマリ検知

Auto Encoder (AE) がある．これは自己学習を行う NN で，AE では入力データを訓練データとして学習させる．

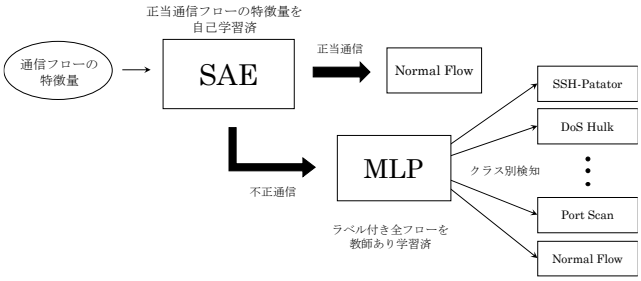


図 1 提案システムの概要
Fig.1 Outline of the Proposed System

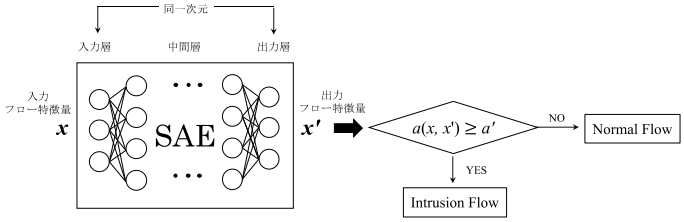


図 2 SAEによるアノマリ検知
Fig.2 Anomaly Detection using SAE

学習済の AE に入力データを与え，出力データが入力データと異なっていれば，学習させたデータではないということがわかる．これを利用して，正常なデータを学習させて異常データを識別する．SAE[10]は，AE にスパース正則化項を導入したものである．正則化項が導入されたことにより，多くのユニットが 0 になり，少ないユニット数で効率的な圧縮または復元が可能となる．正則化項を加えた誤差関数 Eは，

$$E = \sum_{n=1}^{N} \|x_n - x'_n\|^2 + \beta \sum_{j=1}^{M} KL(\rho||\rho'_j) \qquad (1)$$

と表せる．ここで，βは再構成誤差とのバランスをとるためのパラメータで，Nはバッチサイズ，ρは平均活性度の目標値，Mは潜在変数の次元数である．ρ'_jは中間層のユニット jの平均活性度，$KL(\rho \,||\, \rho'_j)$はカルバック･ライブラー情報量という確率分布の差異を測る尺度であり，求めた平均活性度が目標に対してどれだけ近いかを表している．β以降は正則化項で，ここでユニットをスパースにしていく．それぞれの定義式をいかに表す．

$$\rho'_j = \frac{1}{N} \sum_{n=1}^{N} f(W_j x_n + b_j) \qquad (2)$$

$$KL(\rho||\rho'_j) = \rho \log\left(\frac{\rho}{\rho'_j}\right) + (1-\rho)\log\frac{1-\rho}{1-\rho'_j} \qquad (3)$$

SAE は正当データのみを訓練データに用いる半教師あり学習であるため，不正データが入力されたとき，出力データと入力データは乖離した値になる．その仕組みを利用して入力データと出力データのユークリッド距離の差を求め，その値を異常度$a(x, x')$とし，異常度が予め設定した閾値a'を超えた場合にそのデータを不正データとする．以下に異常度を計算する式を示し，アノマリ検知の流れを図 2 に示す．ここでx_iは入力データで，x'_iは出力データである．

$$a(x, x') = \sum_{i=1}^{I} (|x_i - x'_i|) \qquad (4)$$

3.3 MLP 多値分類による不正通信特定

MLP は，入力，中間，出力各層で構成されており，３層 MLP の Forward Propagation は式(5),(6)での式で表現される．

$$z_j = g\left(\sum_{i=1}^{I} (w_{ji} x_i) + b_j\right) \qquad (5)$$

$$y_k = f\left(\sum_{j=1}^{J} (w_{kj} z_j) + b_k\right) \qquad (6)$$

ここで，入力データx_i，重みw_{ji}, w_{kj}，バイアスb_j, b_k，活性化関数$f(x), g(x)$である．式(5)は入力データx_iが入力層から中間層，式(6)は中間層から出力層の伝播である．不正通信の種類をラベルとしてもつフロー特徴量を訓練データとして重みとバイアスを学習させ，フロー特徴量から通信の種類を判別する．

3.4 SAE と MLP の連携

SAE では正当データのみを学習させるため，正当データの識別精度は高い．しかし，二値分類であるため，不正データの種類を特定することはできない．そこで，MLP にクラス別教師あり学習を行い，SAE で不正データであると判定されたフローのクラスを識別させる．このとき，SAE では異常度を小さい値に設定して不正データを見逃さないようにするとともに，MLP では正当データクラスも分類できるようにする．これによって，不正データの検知とクラス別判定を効率よく行うことができる．

表 1 CICIDS 2017 データセット一覧
Table 1 CICIDS Datasets

データセット	含まれているデータ
Monday-WorkingHours.pcap_ISCX.csv	Benign
Tuesday-WorkingHours.pcap_ISCX.csv	Benign, FTP-Patator, SSH-Patator
Wednesday-workingHours.pcap_ISCX.csv	Benign, DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS Slowloris, Heartbleed
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	Benign, Web Attack-Brute Force, Web Attack-Sql Injection, Web Attack-XSS
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	Benign, Infiltration
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	Benign, DDoS
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	Benign, PortScan
Friday-WorkingHours-Morning.pcap_ISCX.csv	Benign, Bot

4. 実験手法

4.1 検証データ

今回の実験で用いた CICIDS 2017[5]はハニーポットを使用してフローデータを収集したもので，8 つのデータセットが含まれている．曜日や時間帯ごとに収集データが異なっている．それぞれのデータセットに含まれているデータの種類と件数について表 1，表 2 に示す．表 2 からわかるように，正当データの量が非常に大きく，Heartbleed などの一部データは含まれているデータ数が非常に少ない．この中から Monday-Working Hours.pcap_ISCX.csv 以外の各データから 7 割を訓練データとして使用し，残りの 3 割をテストデータとして使用した．しかし，Heartbleed, Web Attack-Sql Injection, Infiltration はデータ数が極端に少ないため訓練データを作らず，未知データと判断させるようにした．

表 2　各データの総数

Table 2 Number of data in CICIDS2017 datasets

ラベル	データ数
Benign	2359087
FTP-Patator	7938
SSH-Patator	5897
DoS GoldenEye	10293
DoS Hulk	231072
DoS Slowhttptest	5499
DoS Slowloris	5796
Heartbleed	11
Web Attack-Brute Force	1507
Web Attack-Sql Injection	21
Web Attack-XSS	652
Infiltration	36
DDoS	41835
PortScan	158930
Bot	1966

4.2 データセットの特徴量選択と正規化

CICIDS 2017 データセットを使用するにあたり，効率的に学習するために特徴量を選択し正規化する必要がある．

(1) 特徴量の選択

各データセットには 84 の特徴量とラベルが格納されている．含まれている特徴量を表 3 に示す．

今回は計算の効率化のため数値特徴のみを特徴量として選択したので，Flow ID, Source IP, Source Port, Destination IP, Destination Port, Protocol, Timestamp を除く 77 の特徴量を用いた．

(2) データの正規化

(1)で選択した 77 の特徴量に対してそれぞれのデータ x_i を下記のように正規化し，x'_i を入力データとした．これにより 77 の特徴量すべてを 0 から 1 の範囲に正規化することができ，学習の効率化を図った．

$$x_i' = \frac{x_i - Min_i}{Max_i - Min_i}$$

表 3　CICIDS2017 に含まれる特徴量

Table 3 Feature Quantities in CICIDS2017 datasets

Flow ID	Flow IAT Mean	Bwd Packets/s	Fwd Avg Bulk Rate
Source IP	Flow IAT Std	Min Packet Length	Bwd Avg Bytes/Bulk
Source Port	Flow IAT Max	Max Packet Length	Bwd Avg Packets/Bulk
Destination IP	Flow IAT Min	Packet Length Mean	Bwd Avg Bulk Rate
Protocol	Fwd IAT Total	Packet Length Std	Subflow Fwd Packets
Timestamp	Fwd IAT Mean	Packet Length Variance	Subflow Fwd Bytes
Flow Duration	Fwd IAT Std	FIN Flag Count	Subflow Bwd Packets
Total Fwd Packets	Fwd IAT Max	SYN Flag Count	Subflow Bwd Bytes
Total Backward Packets	Fwd IAT Min	RST Flag Count	Init_Win_bytes_forward
Total Length of Fwd Packets	Bwd IAT Total	PSH Flag Count	Init_Win_bytes_backward
Total Length of Bwd Packets	Bwd IAT Mean	ACK Flag Count	act_data_pkt_fwd
Fwd Packet Length Max	Bwd IAT Std	URG Flag Count	min_seg_size_forward
Fwd Packet Length Min	Bwd IAT Max	CWE Flag Count	Active Mean
Fwd Packet Length Mean	Bwd IAT Min	ECE Flag Count	Active Std
Fwd Packet Length Std	Fwd PSH Flags	Down/Up Ratio	Active Max
Bwd Packet Length Max	Bwd PSH Flags	Average Packet Size	Active Min
Bwd Packet Length Min	Fwd URG Flags	Avg Fwd Segment Size	Idle Mean
Bwd Packet Length Mean	Bwd URG Flags	Avg Bwd Segment Size	Idle Std
Bwd Packet Length Std	Fwd Header Length	Fwd Header Length	Idle Max
Flow Bytes/s	Bwd Header Length	Fwd Avg Bytes/Bulk	Idle Min
Flow Packets/s	Fwd Packets/s	Fwd Avg Packets/Bulk	Destination Port

4.3 実験パラメータと検知精度の判定指標

実験環境としては，プロセッサ Intel Core i7-7700 CPU3.60Ghz，実装メモリ(RAM)：8.00GB(7.87GB 使用可能)および python3.6.3, Tensorflow1.1.0, keras2.0.8 を使用した．

今回の実験で用いた SAE と MLP のパラメータを表 4 に示す．SAE ではこの他に L1 ノルムとして 1.0×10^{-4} の正則化項を与えている．

表 4　各識別器の諸元

Table.4 Specification of SAE and MLP identifier

	SAE	MLP
入力層	77	77
中間層	90,120,160,120,90	90,120,160,110,90
出力層	77	12
活性化関数	中間層：ReLU 出力層：sigmoid	中間層：ReLU 出力層：softmax
エポック数	100	100
バッチサイズ	64	64

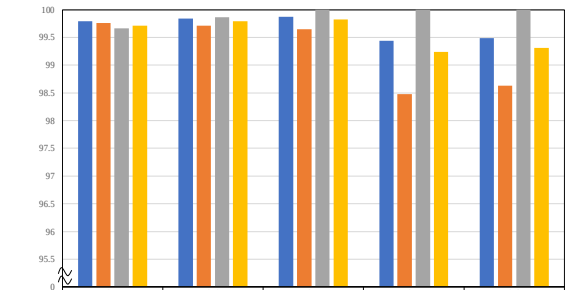


図 3　SAE アノマリ検知の異常度

Fig.3 Influence of Abnormality on SAE Anomaly Detection

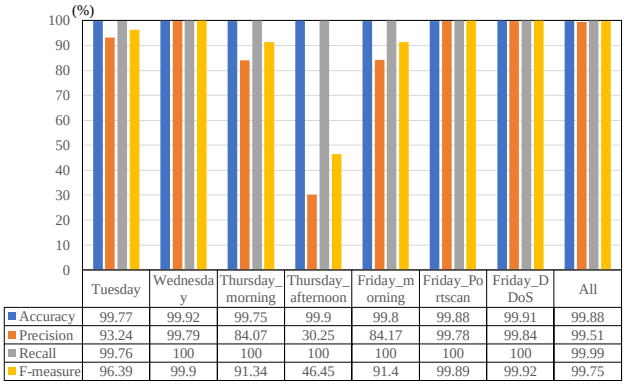
5. 実験結果

5.1 SAE によるアノマリ検知

この実験では TP, TN, FP, FN を求め，これらのデータをもとに Accuracy，Precision，Recall，F-Measure の指標を求めた．各指標の定義と意味については文献[1, 7]などを参照されたい．

まず，SAE での見逃しを最小にするため，Wednesday のデータセットを用いて SAE の異常度による各指標の変化を調べた．図 3 に示すように．a'=0.5 で， Recall が最も高い値となったため，以降は a'=0.5 で実験した．

SAE を用いた曜日ごとの検知及びテストデータ全体の検知結果を図 4 に示す．All となっている項目は，すべてのデータセットに含まれるテストデータを合わせたものである．いずれのデータセットにおいても正当データの割合が非常に大きいため，すべてのデータセットにおいて Accuracy の値が大きくなっている．Thursday_afternoon では Precision が 30.25%となっているが，不正データが 36 件しか含まれておらず，FP が与える影響が大きいためと考えられる．また，All の Accuracy, F-Measure が高いことから曜日毎の SAE を用いる必要はないと考えられる．



	Tuesday	Wednesday	Thursday_morning	Thursday_afternoon	Friday_morning	Friday_Portscan	Friday_DDoS	All
Accuracy	99.77	99.92	99.75	99.9	99.8	99.88	99.91	99.88
Precision	93.24	99.79	84.07	30.25	84.17	99.78	99.84	99.51
Recall	99.76	100	100	100	100	100	100	99.99
F-measure	96.39	99.9	91.34	46.45	91.4	99.89	99.92	99.75

図 4　実験結果：SAE アノマリ検知

Fig.4 Experimental Results: SAE Anomaly Detection

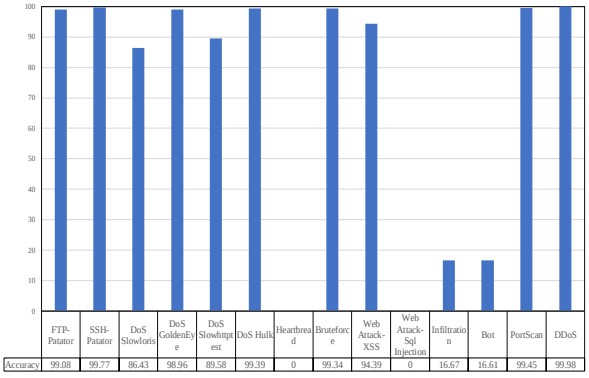
5.2 MLP 単体による多値分類による不正通信特定

MLP を用いて，曜日毎に多値分類した結果を図 5 に示す．データの種類によって検知精度に大きなが差がみられる．MLP に学習させたテストデータに関してはほとんどのデータで 90%を超える精度だったが，未知データとして判断される Heartbleed, Web Attack-Sql Injection, Infiltration の検知ができておらず，別の不正データと判断されていることがわかった．Bot の検知精度が 16.61%と最も低い結果となった．

5.3 SAE/MLP の連携

全体のテストデータについて MLP 単体での検知結果と SAE と MLP とを多段に連携させた検知結果の比較を図 6 に示す．いずれの検知精度についても多段にすることによって微小ながら向上していることがわかる．データ数が多いため微小な向上であっても大きな違いがある．Precision と Recall はトレードオフの関係にあるため，統合的に識別器の精度を測定するために F-Measure を見ると，0.88%向上している．

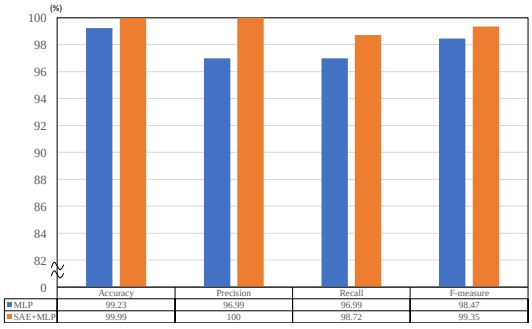
テストデータ全体についての学習時間と判定時間について表 6 に示す．SAE+MLP ではそれぞれの識別器を組み合わせているため，学習時間・判定時間ともに MLP 単体よりも時間がかかるという結果となっているが，正当と不正の判別は SAE で処理されるため，MLP 単体とほぼ同じである．



	FTP-Patator	SSH-Patator	DoS Slowloris	DoS GoldenEye	DoS Slowhttptest	DoS Hulk	Heartbreed	Bruteforce	Web Attack-XSS	Web Attack-Sql Injection	Infiltration	Bot	PortScan	DDoS
Accuracy	99.08	99.77	86.43	98.96	89.58	99.39	0	99.34	94.39	0	16.67	16.61	99.45	99.98

図 5 実験結果：MLP による不正通信特定

Fig.5 Experimental Results: Intrusion Detection by MLP



	Accuracy	Precision	Recall	F-measure
MLP	99.23	96.99	96.99	98.47
SAE+MLP	99.99	100	98.72	99.35

図 6　実験結果：SAE/MLP 連携検知

Fig.6 Experimental Results: SAE/MLP Cooperative Detection

表 5　実験結果：学習及び検知にかかる時間

Table 6 Experimental Results: Computational Time for Training and Detection

	学習時間(秒)	判定時間()
SAE	624.5	19.11
MLP	697.2	17.78
SAE+MLP	1022	23.04

6. 考　察

図 6 に示した通り，提案手法は MLP のみでの検知精度よりも約 1%程度向上した．割合でみると小さな差ではあるが，ネットワークフロー数が多い場合には精度に大きく差ができる．今回使用したデータセットの一部を例に挙げる．Tuesday のデータセットには 1 時間当たり約 34000 件のフローが含まれている．この中の 1%は 340 件となり決して無視できない件数となっている．

今回は， MLP の各出力ノードの最大値が 0.5 未満の場合，未知データと判定するようにし，件数が少ないデータを未知データとしてテストした．しかし，実験の結果，SAE では不正データと判定されたものの，MLP では，ほとんどのデータは他の不正データと判定され，未知データと判定することはできなかった．

7. おわりに

本研究では，アノマリ検知型識別器である SAE とシグネチャ型識別器の MLP を連携させ多段検知を提案した．SEA による正当・不正の判定を先行させることにより，MLP 単体での多値分類に比べ，正当データの検出精度は高い．正当データは，学習時間および判定時間ともに同じ程度で抽出することができる．その後，MLP を適用することにより攻撃の種類を判定することができ，防御対策につなげることができる．

データ件数の少ない攻撃は検知率が低いため，今後，訓練データが少ない攻撃の検知率を改善する手法が必要である．また，未知データを検出する手法についてもさらに検討していきたい．

参考文献

[1] Alshamy, R., & Ghurab, M. A Review of Big Data in Network Intrusion Detection System: Challenges, Approaches, Datasets, and Tools, IJCSE, vol.8, no.7, pp. 62-75, 2020.

[2] Kalimuthan, C., & Renjit, J. A. Review on intrusion detection using feature selection with machine learning techniques. *Materials Today:Proceedings*, https://www.sciencedirect.com/science/article/pii/S2214785320346861, (参照 2020-8-17), 2020.

[3] Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. A detailed analysis of the KDD CUP 99 data set. In 2009 IEEE symposium on CISDA, pp. 1-6, 2009.

[4] Revathi, S. and Malathi, A. A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection. Internal Jornal of Engineering Research& Technology, v. 2, i. 12, pp. 1848-1853, 2013.

[5] 高田雄太, 寺田真敏, 松木隆宏, 笠間貴弘, 荒木粧子, 畑田充弘. マルウェア対策のための研究用データセット ～ MWS 2018 Datasets ～. 研究報告セキュリティ心理学とトラスト(SPT), vol. 2018-SPT-29, no. 38, p. 1-8, 2018.

[6] Panigrahi, R., & Borah, S., A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems, International Journal of Engineering & Technology, vol.7(3.24), pp. 479-482, 2018.

[7] 上野智輝, 原山美知子, 不正侵入パケットの検知における深層学習手法の評価信学技報, vol.117, no. 488, pp.107-113, 2018.

[8] Kingma, D. P., & Ba, J., Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[9] 前田塁, 原山美知子, C-GraphCNN を用いた HTTP 型ボットネット通信の検知, CSS2019 論文集, pp. 430-435, 2019.

[10] Ng, Andrew, et al., Sparse autoencoder, *CS294A Lecture notes*, 72.2011, pp.1-19, 2011.

[11] Chamou, D., Toupas, P., Ketzaki, E., Papadopoulos, S., Giannoutakis, K. M., Drosou, A., & Tzovaras, D. , Intrusion Detection System Based on Network Traffic Using Deep Neural Networks, In 2019 IEEE 24th International Workshop on CAMAD, pp. 1-6, 2019.

[12] Deore, B., Kyatham, A., & Narkhede, S., A novel approach to ensemble MLP and random forest for network security. In ITM Web of Conferences, vol. 32, i. 03003, pp.1-5, 2020.