L-002

時系列特徴に基づく機械学習を用いたサイバー攻撃検知
Cyber Attack Detection Using Machine Learning Based on Time-Series Features

湯山 樹弥† 八槇 博史‡
Yuyama Tatsuya Yamaki Hirofumi

1. はじめに

近年，サイバー攻撃検知技術に機械学習を用いる研究は数多く行われている．しかしながら，時系列を扱う技術は未だ発展途上にあり，攻撃検知のための最適なアルゴリズムは提案されていない．

サイバー攻撃は本質的に時系列特徴を持つ．単発で攻撃が行われるのではなく，予兆，本格的な攻撃，余波という流れに沿って行われる．サイバー攻撃検知のために機械学習を利用するには，パケットログからの学習が必要になる．この際，IP ヘッダーの内容は改竄されうる可能性が高い．一方で，受信日時は受け取った時刻であり信頼できる．

本稿では，一つのサイトに対するトラフィックからの攻撃検知を課題としている．LSTM(Long short-term memory)及び GRU(Gated recurrent unit)という既存の時系列モデルについて調査・検証を行った．評価対象には Kyoto 2016 Dataset を活用した．攻撃パケットを週単位で分割し，ある週の攻撃動向を元にその翌週の攻撃をどれだけの精度で検知できるのか検証した．

2. データセット及びアルゴリズム

2.1 Kyoto 2016 Dataset

トレーニングデータおよびテストデータには Kyoto2016Data を活用した．これは，京都大学内で設置されたハニーポットから収集した通信データである．収集期間は，2006 年 11 月 1 日から 2015 年 12 月 31 日までの 10 年間に上る．全体で 24 の特徴量が存在し[1]，それぞれ Duration, Service, Source byte, Destination bytes, Count, Same srv rate, Serror rate, Srv serror rate, Dst host count, Dst host srv count, Dst host same src port rate, Dst host serror rate, Dst host srv serror rate, Flag, IDS detection, Malware detection, Ashula detection, Label, Source IP Address, Source Port Number, Destination IP Address, Destination Port Number, Start Time, Duration.1 である．それぞれの意味は文献[1]を参照されたい．

2.2 アルゴリズム

2.2.1 LSTM(Long short-term memory)

LSTM とは，短期記憶を長期に渡って活用することを可能にしたニューラルネットワークである．RNN から派生しており，時系列データを扱いやすいという特長がある．LSTM には，通常のニューロンに input gate, forget gate, output gate を追加したメモリセルを使用している．ここで，各ユニットでの出力に着目してみると，次のような計算がされている．

$$f_t = \sigma(w_{xf}\, x(t) + w_{hf}\, h(t-1) + b_f)$$
$$i_t = \sigma(w_{xi}\, x(t) + w_{hi}\, h(t-1) + b_i)$$
$$o_t = \sigma(w_{xo}\, x(t) + w_{ho}\, h(t-1) + b_o)$$

ここで，w は各ユニットでの重み，x(t)は時刻 t での入力データ，h(t)は時刻 t での隠れユニットを示す．これらの計算を以て，最終出力 o(t)が得られる．この際，forget gate にフィードバック機構を付与することによって，RNN で発生していた勾配消失及び勾配発散といった問題を解消している[2]．

2.2.2 GRU(Gated recurrent unit)

GRU は，reset gate と update gate で構成されている．LSTM における input gate と forget gate とを update gate として一つのゲートに統合している[3]．各ゲートでは次のような計算がされている．

$$z = \sigma(x(t)u_z + s(t-1)w_z + b_z)$$
$$r = \sigma(x(t)u_r + s(t-1)w_r + b_r)$$

w は各ユニットでの重み，x(t)は時刻 t での入力データ，s(t)は時刻 t での隠れユニットを示す．LSTM は無制限の計数を容易に実行できるが GRU では不可能だとしているため，LSTM は GRU よりも「厳密に強力」である．これが LSTM によって学習可能な単純な言語の学習を GRU が失敗する理由だとしている[4]．

3. 実験

Ubuntu Server 20.04 を用い Python3.7 環境を構築した．ディープラーニングを実行するためのライブラリに Keras 及び Tensorflow を，主成分分析には sklearn を使用した．

3.1 特徴選択

特徴量を精査するために，フィルター法を用いて Label との相関関係及び重要度を算出した．一部適用できなかったものを除き，文字列特徴に対してラベルエンコーディングを施した．week1 を対象とした結果，相関関係について上位 5 件は Duration.1, Flag, Same srv rate, Srv serror rate, Dst host srv count であった．重要度については同様に，Malware detection, Srv serror rate, Duration.1, Count, Flag であった．

3.2 データ整形と前処理

第一に，学習コストを削減するために元のデータセットから，Label の比率を維持しつつ 10%抽出した．Start Time は時刻のみの情報であったため，月日の情報を付与した．日時順にソートを行い，週単位の攻撃パケットとして一つのファイルに保存した．

2.1 で示した特徴のうち，IDS detection, Malware detection, Ashula detection, Source IP Address, Destination IP Address を除いたものを使用した．文字列特徴にはラベルエンコーディングを施し，その後全体に MinMaxScaler を適用した．目標変数をダミー変数化するために One Hot Encoding を適用した．アルゴリズムに渡すためのタイムステップは 1 と定めた．

† 東京電機大学 情報環境学研究科 Tokyo Denki University Graduate School of Information Environment
‡ 東京電機大学 システムデザイン工学部 Tokyo Denki University School of System Design and Technology

3.3 攻撃検知

検知のためのアルゴリズムには，LSTM 及び GRU を活用した．目標変数が三値(正常通信，既知攻撃，未知攻撃)で表現されているため，分類・予測アプローチを用いた．多値分類であるため，評価指数には Accuracy を採用した．ある週の攻撃パターンを学習させ，その次の週の攻撃パターンを検知している．

3.4 実験結果

実験の結果を Fig1 に示す．縦軸には Accuracy を，横軸にはテストデータとして使用した週をとった．この結果，LSTM では平均 92.7611878%，GRU では平均 92.564549%の精度で検知できた．一方，精度の変遷に着目すると week5 では LSTM で 13.806%，GRU で 13.4488%と他に比べて大きく落ち込んでいる．

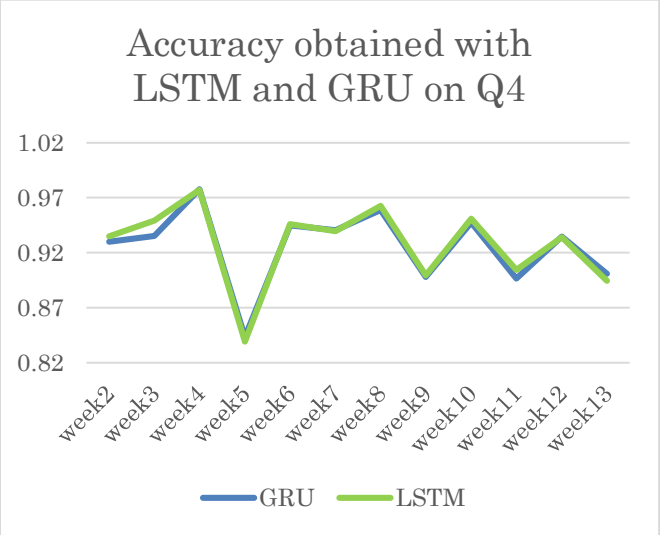

Fig1. Accuracy obtained with LSTM and GRU on Q4

4. 考察

実験の結果，サイバー攻撃の検知においても LSTM 及び GRU における優劣が存在しないと判明した．Week5 における落ち込みを調査するために主成分負荷量の計算を行った.数値特徴を対象とした結果を Table1 に示す．week4 と week5 では，第 2 主成分までは類似した傾向があるものの，第 3 主成分では異なる傾向があると判明した．week4 では Dst host srv count の寄与が最大である一方，week5 では Serror rate の寄与が最大となった．この背景には攻撃動向の変遷があると推察できる．より新しいデータを利用することで，検知精度や攻撃動向の変遷についても検討を行う．

Table1. Principal component loading on week4 and week5 in Q4

Week	Vector	Duration	Source byte	Destination bytes	Count	Same srv rate	Serror rate	Srv serror rate	Dst host count	Dst host srv count	Dst host same src port rate	Dst host serror rate	Dst host srv serror rate	Label	Source Port Number	Destination Port Number
4th	Vec1	0.012	-0.001	0.004	0.333	0.417	0.188	-0.319	0.480	0.494	0.045	0.209	0.164	0.152	-0.020	-0.086
4th	Vec2	0.053	-0.002	-0.016	-0.274	-0.129	0.239	0.225	-0.057	0.002	0.368	0.553	0.539	-0.185	-0.164	-0.027
4th	Vec3	0.056	0.041	0.022	0.018	0.130	0.404	0.280	0.098	0.041	-0.462	-0.095	-0.059	-0.256	-0.509	0.416
4th	Vec4	0.662	0.034	0.673	-0.025	-0.124	-0.213	-0.108	0.043	0.038	0.007	0.026	0.003	0.068	-0.155	0.043
4th	Vec5	0.170	0.017	0.202	-0.224	0.370	0.610	0.030	-0.227	-0.164	-0.006	-0.105	-0.006	0.236	0.473	0.009
5th	Vec1	-0.011	-0.001	0.000	0.196	0.436	0.006	-0.436	0.453	0.466	-0.108	-0.074	-0.116	0.298	0.147	-0.131
5th	Vec2	0.025	-0.002	0.000	0.026	0.087	0.279	0.035	0.170	0.193	0.336	0.597	0.560	-0.105	-0.210	-0.088
5th	Vec3	0.032	0.001	0.000	-0.495	-0.263	-0.579	-0.197	0.051	0.073	0.272	0.143	0.115	0.148	0.324	-0.260
5th	Vec4	0.694	0.011	0.669	-0.071	-0.042	-0.051	-0.021	0.074	0.056	-0.157	-0.005	0.001	-0.063	-0.151	0.031
5th	Vec5	0.097	0.069	0.274	0.282	0.182	0.238	0.018	-0.287	-0.215	0.514	-0.044	-0.033	0.210	0.545	-0.072

5. おわりに

機械学習によるサイバー攻撃検知の発展のためには，時系列モデルの検討が必須である．本稿では， LSTM 及び GRU を活用した．週単位でパケットを分割し，多値分類ゆえ評価指数には Accuracy を利用した．この結果，二者の性能には現時点では特には優劣がないことがわかった．今後は最新のデータを使用することで，サイバー攻撃が持つ時系列特徴についてさらなる検討を進める．

参考文献

[1] 多田竜之介ほか 3 名, “NIDS 用評価データセット:Kyoto 2016 Dataset の作成”, 情報処理学会論文誌 Vol.58, No.9 (2017), pp. 1450-1463
[2] Flex A. Gers, Jurgen Schmidhuber and Fred Cummins, “Learning to Forget: Continual Prediction with LSTM”, Technical Report IDSIA-01-99, January 1999
[3] Kyunghyun Cho et al., “Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translatio”, arXiv:1406.1078, September 2014
[4] Gail Weiss, Yoav Goldberg and Eran Yahav, “On the Practical Computational Power of Finite Precision RNNs for Language Recognition”, arXiv:1805.04908 , May 2018