УДК 004.056.53
DOI 10.25205/1818-7900-2019-17-2-114-121

Применение методов глубинного обучения для обнаружения вторжений

В. А. Нечахин¹, Б. Н. Пищик^{1,2}

¹ Новосибирский государственный университет
Новосибирск, Россия

² Институт вычислительных технологий СО РАН
Новосибирск, Россия

Аннотация

Приведены результаты применения глубоких нейронных сетей для детектирования вредоносной активности в сетевом трафике. В процессе исследования реализованы два вида нейронной сети: рекуррентный автоэнкодер и генеративно-состязательная сеть. Приведены результаты исследования на наборе данных CICIDS2017.

Ключевые слова

обнаружение вторжений, глубинное обучение, CICIDS2017, автоэнкодер, генеративно-состязательная сеть.

Для цитирования

Нечахин В. А., Пищик Б. Н. Применение методов глубинного обучения для обнаружения вторжений // Вестник НГУ. Серия: Информационные технологии. 2019. Т. 17, № 2. С. 114–121. DOI 10.25205/1818-7900-2019-17-2-114-121

Using Deep Learning Methods for Intrusion Detection

V. A. Nechakhin¹, B. N. Pishchik^{1,2}

¹ Novosibirsk State University
Novosibirsk, Russian Federation

² Institute of Computational Technologies SB RAS
Novosibirsk, Russian Federation

Abstract

One of the ways of ensuring information security are intrusion detection systems (IDS). IDS are used to detect malicious activity on the network. The standard approach to the detection of attacks is looking for specific patterns, such as byte sequences in network traffic, or known malicious instruction sequences used by malware. This approach is highly efficient, but it does not able to detect the attacks without patterns. Modern approaches to detection of attacks use deep learning.

The purpose of this work was to explore the possibility of building a universal classifier of network traffic based on a deep neural network.

For this, a recurrent autoencoder was trained on TCP packets from the CICIDS2017 dataset. During training the neural network was a model in which the expected vector was set the same as the original one. And learning was on normal traffic. The main idea was that a recurrent autoencoder trained in this way should recover anomalous traffic with a high loss. The TCP package is considered malicious if the recovery loss is above the threshold. However, the accuracy of recovering normal TCP packets was low due to the insufficient model capacity and the lack of the suitable representation learning method.

After the results analyzing, we proposed an approach that can improve accuracy of detection for some attacks. Based on this approach, the VAEGAN network was trained on normal network flows from CICIDS2017. The VAEGAN was used to detect malicious network flows: to calculate the anomaly score for flow; if score is above the threshold – the flow is malicious.

The VAEGAN network showed a high percentage of attacks detection and the F-score value – 0.933.

© В. А. Нечахин, Б. Н. Пищик, 2019

Keywords

anomaly detection, deep learning, CICIDS2017, autoencoder, generative adversarial network

For citationNechakhin V. A., Pishchik B. N. Using Deep Learning Methods for Intrusion Detection. *Vestnik NSU. Series: Information Technologies*, 2019, vol. 17, no. 2, p. 114–121. (in Russ.) DOI 10.25205/1818-7900-2019-17-2-114-121**Введение**

Выявление вредоносной активности в сетевом трафике является актуальной задачей, к решению которой существует несколько подходов [1].

Классический подход состоит в написании шаблонов, по которым в сетевом трафике детектируется вредоносная активность. Данный подход является высокоэффективным, но не позволяет обнаруживать атаки, для которых отсутствует шаблон.

Современные подходы базируются на том, что детектирование вредоносного трафика является частным случаем классической задачи классификации, которая в настоящее время успешно решается с помощью машинного обучения, в частности, нейронных сетей. Однако простые алгоритмы машинного обучения являются эффективными только при правильном ручном извлечении признаков из исходных данных. А извлечение необходимых признаков во многих случаях само по себе является не тривиальной задачей. Применение методов *обучения признакам* позволяет автоматически выявлять признаки, которые необходимы для дальнейшей классификации. А для выявления высокоуровневых признаков используют методы глубинного обучения [2].

Цель работы – исследовать возможность построения универсального классификатора сетевого трафика на основе глубокой нейронной сети.

CICIDS2017

Для обучения нейронной сети был сформирован набор данных CICIDS2017¹, который является публично доступным и предназначен для исследований в области кибербезопасности и обнаружения вторжений. Набор данных CICIDS2017 распространяется в виде файлов 2-х форматов: PCAP и CSV. Первые хранят дампы трафика, охватывающие пять дней сетевой активности, вторые содержат описание для каждого сетевого потока: статистические признаки и класс, к которому принадлежит поток.

Поток – это последовательность сетевых пакетов, у которых совпадают следующие поля: IP источника, IP получателя, порт источника, порт получателя и протокол. Для каждого потока выделяется более 80 признаков². Общие данные о наборе CICIDS2017 представлены в табл. 1.

Рекуррентный автоэнкодер

В качестве глубокой нейронной сети был выбран рекуррентный автоэнкодер [2] как нейронная сеть, предназначенная для обработки последовательностей различной длины. Обучение нейронной сети проходило на TCP пакетах из дампов трафика. Архитектура рекуррентного автоэнкодера представлена в табл. 2.

¹ Intrusion Detection Evaluation Dataset (CICIDS2017) // University of New Brunswick, Canadian Institute for Cybersecurity. URL: <https://www.unb.ca/cic/datasets/ids-2017.html> (дата обращения 21.11.2017).

² CICFlowMeter // University of New Brunswick, Canadian Institute for Cybersecurity. URL: <https://www.unb.ca/cic/research/applications.html#CICFlowMeter> (дата обращения 21.11.2017).

Таблица 1

Общие данные о CICIDS2017

Table 1

General Information about CICIDS2017

Класс	Вредоносность	Количество потоков
Benign	-	2 358 036
DoS Hulk	+	231 073
Port Scan		158 930
DDoS		41 835
DoS GoldenEye		10 293
FTPPatator		7 938
SSHPatator		5 897
DoS Slow Loris		5 796
DoS Slow HTTP Test		5 499
Botnet		1 966
Web Attack: Brute Force		1 507
Web Attack: XSS		652
Infiltration		36
Web Attack: SQL Injection		21
Heartbleed		11

Таблица 2

Архитектура рекуррентного автоэнкодера

Table 2

Recurrent Autoencoder Architecture

Слой	Число нейронов	Функция активации	Dropout
Энкодер			
GRU	128	ReLU	0.2
GRU	64	ReLU	0.2
Декодер			
GRU	128	ReLU	0.2
GRU	256	Softmax	0
Размерность входного слоя		256	
Оптимизатор		$RMSProp(lr = 10^{-5})$	
Функция ошибки		MSE	
Размер пакетов (batch size)		64	
Количество эпох		100	

Для более равномерного последующего нормирования входных значений каждому нейрону на вход подается 1 байт пакета. А выбранный размер входного слоя (256) является предположительно оптимальным с точки зрения качества работы сети и сложности вычислений.

При обучении рекуррентный автоэнкодер представлял собой модель, в которой ожидаемый вектор устанавливался таким же, как и исходный. А обучение происходило исключительно на легитимном трафике. Идея заключалась в том, что обученный таким образом рекуррентный автоэнкодер должен восстанавливать аномальный трафик с *высокой степенью* ошибки. Ошибка восстановления рассчитывается как значение близости исходного TCP пакета и восстановленных данных.

Для того чтобы определить, является ли пакет вредоносным, необходимо ввести параметр «порог ошибки восстановления». Таким образом, корректируя значение порога ошибки вос-

становления можно настраивать точность выявления вредоносной активности. Однако во время обучения на сконструированном рекуррентном автоэнкодере не удалось получить точность восстановления более 60 %.

Предметная область (изображения, звук, текст) в существенной степени влияет на процесс обучения признакам. Например, сверточные нейронные сети в рамках задачи распознавания изображений используют иерархии объектов и базируются на особенностях работы зрительной коры головного мозга [3]. Для сетевого трафика такой концепции на данный момент не выработано. Отсутствие концепции обучения признакам и недостаточная ёмкость модели стали причинами низкой точности восстановления.

Сетевой трафик в наборе CICIDS2017 представляет собой совокупность потоков, в свою очередь, поток – это совокупность сетевых пакетов. Для обучения нейронной сети в качестве входных данных могут быть использованы сетевые пакеты или потоки в зависимости от того, какие классы атак необходимо детектировать. Достоинства и недостатки двух подходов можно проследить на примере обнаружения конкретных типов атак.

Атаки класса Brute Force сложно выявить инспекцией только отдельных сетевых пакетов. Ключевой особенностью таких атак является отправка множества однотипных пакетов за небольшой промежуток времени. Вредоносные и легитимные пакеты в данном случае слабо различимы.

Такие атаки, как SQL Injection или XSS, ключевая информация для детектирования которых содержится внутри полезной нагрузки, сложно детектировать на уровне потоков. При работе с сетевыми потоками информация заключенная внутри полезной нагрузки, теряется.

Вероятно, статистические признаки сетевого потока не позволяют закодировать достаточно много информации. Один поток может содержать сотни и тысячи сетевых пакетов, каждый из которых имеет собственные уникальные признаки. При агрегировании сетевых пакетов до уровня потоков теряется часть информации. Это то же самое, что взять предложения из некоторого текста и заменить вектором метаданных, который кодирует количество слов и символов в предложении.

Таким образом, перспективным можно считать подход, при котором для различных классов атак реализуется свой метод обнаружения, основанный на особенностях представления данных и применимости определенного типа нейронной сети. Это означает, что необходимо отказаться от идеи универсального классификатора и сосредоточить внимание только на определенной группе атак. Например, классы атак, представленные в CICIDS2017, можно разделить на две группы и использовать рекомендуемые типы нейронных сетей в соответствии с табл. 3.

Таблица 3

Рекомендуемые типы нейронных сетей для CICIDS2017

Table 3

Recommended Types of Neural Networks for CICIDS2017

Классы атак	Типы нейронной сети	Данные для обучения
DoS, DDoS, Port Scan, FTPPatator, SSHPatator, Bot, Web Attack: Brute Force, Infiltration, Heartbleed	Автоэнкодер и генеративно-состязательная сеть [4]	Сетевые потоки
Web Attack: SQL Injection и XSS	Seq2Seq автоэнкодер *	Сетевые пакеты

* Seq2seq-web-attack-detection // Positive Technologies. URL: <https://github.com/PositiveTechnologies/seq2seq-web-attack-detection> (дата обращения 22.02.2019).

VAEGAN

На основе предложенного подхода была разработана нейронная сеть, которая объединяет в себе вариационный автоэнкодер (Variational Autoencoder, VAE) и генеративно-состязатель-

ную сеть (Generative Adversarial Network, GAN). Сети такого типа были предложены в [5] и имеют название VAEGAN.

Обучение реализованной нейронной сети происходило без учителя, на легитимных сетевых потоках из набора CICIDS2017. В качестве тестовых данных были выбраны легитимные сетевые потоки, которые не входили в тренировочный набор, и сетевые потоки всех типов вредоносной активности. Общая информация о тестовой выборке представлена в табл. 4, архитектура реализованной VAEGAN сети – в табл. 5.

Таблица 4

Общая информация о тестовой выборке

Table 4

General Information about Test Dataset

Класс	Вредоносность	Количество потоков
Benign	–	1 000
Port Scan	+	1 000
DDoS		1 000
DoS		1 000
FTPPatator		1 000
SSHPatator		1 000
Botnet		1 000
Web Attack: Brute Force		1 000
Web Attack: XSS		652
Infiltration		36
Web Attack: SQL Injection		21
Heartbleed		11

Таблица 5

Архитектура VAEGAN

Table 5

VAEGAN Architecture

Слой	Число нейронов	Функция активации	Dropout
Энкодер (E)			
Dense	64	LeakyReLU($\alpha = 0.2$)	0.2
Dense	32	LeakyReLU($\alpha = 0.2$)	0.2
Dense	32	LeakyReLU($\alpha = 0.2$)	0.2
Декодер (G)			
Dense	32	LeakyReLU($\alpha = 0.2$)	0.2
Dense	32	LeakyReLU($\alpha = 0.2$)	0.2
Dense	64	LeakyReLU($\alpha = 0.2$)	0.2
Dense	80	Tanh	0
Дискриминатор (D)			
Dense	64	LeakyReLU($\alpha = 0.3$)	0.2
Dense	64	LeakyReLU($\alpha = 0.3$)	0.2
Dense	32	LeakyReLU($\alpha = 0.3$)	0.2
Dense	1	Sigmoid	0
Оптимизатор		Adam($\alpha = 0.001$, $\beta_1 = 0.5$)	
Размер пакета (batch size)		64	
Размерность входного слоя энкодера и дискриминатора		80	
Размерность кода		20	
Начальная инициализация весов		Xavier normal	

В VAEGAN вариационный автоэнкодер получает на вход вектор признаков x для сетевого потока и учится восстанавливать максимально похожий вектор $\tilde{x} = G(E(x))$ на выходе. Дискриминатор принимает на вход восстановленный $\tilde{x}$ и исходный x векторы и учится предсказывать, какой вектор является настоящим, а какой восстановленным.

Выявление вредоносной активности реализовано на основе подхода, представленного в [6]. Для того чтобы определить, является ли сетевой поток вредоносным, вычисляется значение аномальности:

$$\mathcal{L}(x) = 0.1 * \mathcal{L}_D + 0.9 * \mathcal{L}_R,$$

где

x – сетевой поток,

$\tilde{x} = G(E(x))$ – восстановленный вариационным автоэнкодером сетевой поток,

$\mathcal{L}_D(\tilde{x}) = \sum |D_l(x) - D_l(\tilde{x})|$ – ошибка на l -м слое дискриминатора,

$\mathcal{L}_R(\tilde{x}) = \sum |x - \tilde{x}|$ – ошибка вариационного автоэнкодера.

Поток считается вредоносным, если значение $\mathcal{L}(x)$ превышает установленный порог. Корректировать точность выявления вредоносной активности можно, регулируя значение порога аномальности. Полученные результаты представлены на рисунке.

Результат работы VAEGAN
The VAEGAN Results

Каждая точка на этом рисунке отображает значение аномальности для сетевого потока. Красная линия – установленный порог. В результате полученная точность составила 0,983, полнота – 0,889, значение F-меры – 0,933. Процент правильной классификации по классам представлен в табл. 6.

Результат работы VAEGAN

Таблица 6

Table 6

The VAEGAN Results

Класс	Правильно классифицированные потоки, %
Benign	88,2
Port Scan	98,3
DDoS	100
DoS	100
FTPPatator	100
SSHPatator	99,7
Botnet	63,9
Web Attack: Brute Force	98,6
Web Attack: XSS	48,3
Infiltration	100
Web Attack: SQL Injection	4,8
Heartbleed	100

Заключение

В настоящее время многие задачи классификации успешно решаются с помощью глубоких нейронных сетей. В работе был сконструирован рекуррентный автоэнкодер для задачи классификации сетевого трафика. Использование нейронной сети данного типа было обусловлено тем, что исходными данными являлись последовательные сетевые пакеты различной длины. Однако не было заложено никакой концепции обучения признакам. В результате точность восстановления оказалась низкой.

При анализе полученных результатов были рассмотрены ключевые особенности сетевого трафика в рамках задачи классификации. Это позволило сделать предположение о неэффективности универсального классификатора и предложить подход, повышающий точность обнаружения вредоносного трафика. На основе сформулированного подхода на легитимных сетевых потоках была обучена VAEGAN сеть. В результате удалось получить высокую точность выявления сетевых потоков для выделенных классов атак. Низкий процент детектирования оказался для атак SQL Injection, XSS и Botnet, для успешного выявления которых требуется инспекция полезной нагрузки и применение рекомендуемого в табл. 3 типа нейронной сети.

Список литературы / References

1. **Браницкий А., Котенко И.** Анализ и классификация методов обнаружения сетевых атак // Тр. СПИИРАН. 2016. №. 45. С. 207-244. DOI: <https://doi.org/10.15622/sp.45.13>
Branitskii A., Kotenko I. Analiz i klassifikatsiya metodov obnaruzheniya setevykh atak [Analysis and classification of network attack detection methods]. In: Trudy SPIIRAN (SPIIRAS Proceedings), 2016, vol. 45, p. 207–244. (in Russ.) DOI 10.15622/sp.45.13
2. **Goodfellow I., Bengio Y., Courville A.** Deep Learning. MIT Press, 2016, 802 с.
3. **Masakazu M., Mori K., Mitari Y., Kaneda Y.** Subject Independent Facial Expression Recognition With Robust Face Detection Using a Convolutional Neural Network. *Neural Networks*, 2003, no. 16, p. 555–559. DOI 10.1016/S0893-6080(03)00115-1

4. **Zenati H., Foo C. S., Lecouat B., Manek G., Chandrasekhar V. R.** Efficient GAN-based anomaly detection. In: 6th International Conference on Learning Representations, 2018.
5. **Larsen A. B. L., Sonderby S. K., Larochelle H., Winther O.** Autoencoding beyond pixels using a learned similarity metric. In: ICML'16 Proceedings of the 33rd International Conference on International Conference on Machine Learning, 2016, no. 48, p. 1558–1566.
6. **Schlegl T., Seeböck P., Waldstein S. M., Schmidt-Erfurth U., Langs G.** Unsupervised Anomaly Detection with Generative Adversarial Networks to Guide Marker Discovery. In: The proceedings of the international conference on Information Processing in Medical Imaging (IPMI), 2017. DOI 10.1007/978-3-319-59050-9_12

Материал поступил в редколлегию

*Received
02.04.2019*

Сведения об авторах / Information about the Authors

Нечахин Владислав Александрович, магистрант факультета информационных технологий Новосибирского государственного университета (ул. Пирогова, 1, 630090, Новосибирск, Россия)

Vladislav A. Nechakhin, Master's Student, Faculty of Information Technologies, Novosibirsk State University (1 Pirogov Str., Novosibirsk, 630090, Russian Federation)

v.nechakhin@g.nsu.ru

Пищик Борис Николаевич, кандидат технических наук, заведующий кафедрой компьютерных систем факультета информационных технологий Новосибирского государственного университета (ул. Пирогова, 1, 630090, Новосибирск, Россия), старший научный сотрудник Института вычислительных технологий СО РАН (пр. Академика Лаврентьева, 6, Новосибирск, 630090, Россия)

Boris N. Pishchik, Candidate of Science (Technology), Head of the Computer Systems Department at the Faculty of Information Technology, Novosibirsk State University (1 Pirogov Str., Novosibirsk, 630090, Russian Federation), Senior Researcher, Institute of Computational Technologies SB RAS (Pr. Academician Lavrentiev, 6, Novosibirsk, 630090, Russian Federation)

b.pishchik@nsu.ru

PishchikBN@ict.nsc.ru