АНАЛИЗ СОСТОЯНИЯ УСТРОЙСТВ ИНТЕРНЕТА ВЕЩЕЙ НА ОСНОВЕ БЭГГИНГА

DOI: 10.36724/2072-8735-2020-14-12-45-50

Manuscript received 12 October 2020
Accepted 24 November 2020

Сухопаров Михаил Евгеньевич,
Санкт-Петербургский федеральный исследовательский центр Российской академии наук, Санкт-Петербург, Россия,
sukhoparovm@gmail.com

Лебедев Илья Сергеевич,
Санкт-Петербургский федеральный исследовательский центр Российской академии наук, Санкт-Петербург, Россия,
isl_box@mail.ru

Ключевые слова: *анализ состояния, интернет вещей, бэггинг, дискриминантный анализ, мониторинг состояния, алгоритм классификации, байесовский классификатор, деревья решений*

Развитие концепции интернета вещей обусловливает необходимость поиска и совершенствования моделей и методов анализа состояния удаленных автономных устройств. В связи с возможным нахождением элементов интернета вещей вне контролируемой зоны возникает необходимость разработки универсальных моделей и методов идентификации состояния маломощных с вычислительной точки зрения устройств, использующих комплексные подходы анализа данных, поступающих от различных информационных каналов. Рассматривается подход к идентификации состояния устройств интернета вещей, на основе параллельно функционирующих классификаторов, обрабатывающих временные ряды, полученные от элементов в различных состояниях и режимах работы. Целью работы является разработка подхода идентификации состояния устройств интернета вещей на основе временных рядов, регистрируемых при выполнении различных процессов. Предлагаемое решение основано методах параллельной классификации и статистического анализа, требует начальной размеченной выборки. Применение ряда классификаторов, которые выдают "независимо" друг от друга ответ, дает возможность усреднить ошибку "коллективным" голосованием. Разработанный подход протестирован на последовательности классифицирующих алгоритмов, на вход которых подавались полученные экспериментальным путем в различных условиях функционирования временные ряды. Приведены результаты для наивного байесовского классификатора, деревьев решений, дискриминантного анализа, метода k ближайших соседей. Применение последовательности алгоритмов классификации, функционирующих параллельно, позволяет осуществлять масштабирование путем добавления новых классификаторов без потери скорости обработки. Метод дает возможность идентифицировать состояние устройства интернета вещей, обладая относительно небольшими требованиями к вычислительным ресурсам, простотой реализации, возможностями по масштабированию путем добавления новых классифицирующих алгоритмов.

Информация об авторах

Сухопаров Михаил Евгеньевич, *к.т.н., Старший научный сотрудник лаборатории интеллектуальных систем Санкт-Петербургского федерального исследовательского центра Российской академии наук, Санкт-Петербург, Россия,*

Лебедев Илья Сергеевич, *д.т.н., профессор, Заведующий лабораторией интеллектуальных систем Санкт-Петербургского федерального исследовательского центра Российской академии наук, Санкт-Петербург, Россия*

Для цитирования:
Сухопаров М.Е., Лебедев И.С. Анализ состояния устройств интернета вещей на основе бэггинга // T-Comm: Телекоммуникации и транспорт. 2020. Том 14. №12. С. 45-50.

For citation:
Sukhoparov M.E., Lebedev I.S. (2020) State analysis internet of things devices based on bagging. *T-Comm*, vol. 14, no. 12, pp. 45-50. (*in Russian*)

Введение

Стремительное развитие концепции Интернета вещей (IoT), базирующееся на беспроводных технологиях, облачных вычислениях, распределенных системах является основополагающим трендом для информационных и киберфизических систем. Реализация современных индустриальных, промышленных, социальных и бытовых систем невозможна без применения методов искусственного интеллекта межмашинного обмена отдельных элементов, автоматического сбора, анализа, хранения данных. Повсеместное использование различных сенсорных элементов и сетей направлено на решение огромного количества индустриальных, бытовых задач и общественных потребностей с минимальным участием человека, что с одной стороны приносит неоспоримые преимущества, а с другой – обуславливает необходимость решения проблемных вопросов анализа состояния и выявление аномалий функционирования, связанных со сбоями, отказами, некорректным выполнением процессов.

Взаимодействие элементов IoT друг с другом посредством внешней среды предопределяет необходимость создания различных систем мониторинга состояния, в рамках которых обеспечивается безопасное межмашинное взаимодействие, сетевой доступ, передача данных, маршрутизация, интеллектуальная обработка данных и т.д. Одновременно с этим необходимо учитывать динамичность развития информационных систем, предполагающее одновременное использование и «старых» и «новых» устройств различных разработчиков, большого числа протоколов обмена, обработки данных в условиях постоянного добавления новых сегментов.

Производственные процессы типизации, унификации отдельных вычислительных элементов, отсутствие должной физической безопасности элементов интернета вещей, возникающее вследствие нахождения вне контролируемой зоны устройств, возможность появления ситуаций, связанных с обновлением прошивок, программного обеспечения, сбора информации, доступ к типовым устройствам позволяют применять методы реверс-инжиниринга с целью совершенствования контроля состояния в различных режимах работы.

Таким образом, возникает задача совершенствования систем контроля и мониторинга состояния устройств интернета вещей, где одним из направлений является разработка алгоритмов, моделей, методов обработки и анализа информации побочных, сторонних каналов, содержащих информацию о протекающих процессе [5-8].

Выполнение команд, предопределенных последовательностей действий, ставится в соответствие допустимыми значениям параметров функционирования, регистрируемых по различным каналам. Фиксируемые значения образуют временные ряды, на основе которых, применяя методы машинного обучения, статистического анализа, определяются шаблоны и вычисляются нормальное и аномальное состояние.

Существующие подходы

В ходе функционирования устройств интернета вещей могут возникать коллизии как на уровне информационной системы, так и отдельного устройства, например, внедрение программно-аппаратных прошивок, содержащих ошибки, при производстве бытовых устройств, таких как роутеры, принтеры, веб-камеры были связаны с рядом ситуаций, в результате которых происходила загрузка каналов, что ограничивало процессы приема и передачи информации [10-12].

В целях предотвращения подобных инцидентов происходит совершенствование и адаптация моделей, методов мониторинга состояния, направленных на оценку функциональности и производительности. В их основе лежат принципы статистического анализа, анализа причинно-следственных связей, переходов, формирования прецедентных, событийных моделей [5-9].

Модели, основанные на статистике, накапливают информацию о параметрах функционирования в различных режимах и состояниях, а в дальнейшем, для выявления аномальной ситуации с помощью методов нейронных сетей, марковских моделей, машинного обучения и других обрабатываются кортежи признаков [13,14].

Обнаружение внутренних сбоев и отказов функционирующих устройств применяются программы-мониторы, отслеживающие выполнение сегментов кода и дестабилизирующих работу ситуаций, например, переполнение буфера [8].

Другим направлением является обработка сторонних каналов, где состояние анализируется с помощью временных рядов параметров, отражающих изменения загрузки процессора, использования внутренней памяти, интенсивности обмена сообщениями.

Многообразие элементов интернета вещей, большое число объектов, протоколов взаимодействия, технологий обработки данных, неоднородность форматов, постоянно меняющаяся архитектура и изменения конфигурации может приводить к различным отказам и сбоям функционирования, влияющим на параметры функционирования. Анализ значений сторонних каналов (например, электромагнитных, акустических излучений, напряжения, потребляемой мощности) при выполнении различных операций и команд устройством позволяет реализовывать внешние, относительно независимые, не потребляющие вычислительные ресурсы устройств IoT системы мониторинга и контроля состояния.

Постановка задачи

Разработка устройств интернета вещей, программного и аппаратного обеспечения происходит с использованием типовых микросхем, стандартных библиотек разных производителей и разработчиков, что затрудняет анализ исходного кода. Методы быстрой разработки программных и аппаратных частей, позволяющие использовать готовые компоненты различных производителей, приводят к тому, что устройства представляют из себя «черный ящик».

Устройства IoT не обладают большими вычислительными ресурсами, имеют ограниченный набор выполняемых команд, что позволяет рассматривать и идентифицировать относительно не большое количество состояний и их переходов.

Во время функционирования процессы устройств IoT протекают в динамике, одновременно меняется множество параметров.

Состояние внешней среды $u(t)$, вызванное поступлением на устройство команд управления, приемом, передачей сообщений, функционирование элемента, определяемое внутренними ситуациями обработки данных и реализации вычислительных алгоритмов, характеризующееся переходными характеристики $h(t)$ дает возможность рассмотреть устройство как динамическую систему. Имеется q входов и d выходов [11], на вход подается управляющая команда и значения переменных внешней среды, на выходе появляются сигналы $S(t)$ (например, показывающие загрузку ресурсов),

регистрируемые различными датчиками. Получаемые по внешним каналам значения сигналов содержат значения шумовой составляющей $v(t)$, определяемой свойствами измерительного прибора, характеристик получаемого сигнала и т.д.

Модель состояния IoT-устройства определяется соотношением [12]:

$$\sum_{i=1}^{q}\sum_{j=1}^{d}\int_0^t u_i(t)h_{ij}(t-\tau)d\tau=\sum_{j=1}^{d}\int_0^t f(s_j(t-\tau),v_j(t-\tau))d\tau \quad (1)$$

где q – количество каналов источников; h – переходные характеристики i-го канала для j-го регистрирующего, получаемые по каналу значения датчика; f – функция измеренных значений.

В дискретные моменты времени функционирования устройства $t_0, t_1,\dots, t_n$ происходит регистрация векторов числовых последовательностей. Значения $X(t)$ отражают данные, полученные от датчиков, содержащие смесь полезного сигнала $S(t)$ и шума, выраженного параметром $v(t)$:

$$X(t)=\mathrm{F}\left[S(t),v(t)\right],$$

где вектор X является результатом смешанных, взаимно независимых сигналов $S(t)$, имеющих искажение шумовой составляющей $v(t)$. Вектор X представляет собой временной ряд значений, полученный от регистрирующих устройств.

Векторы X_1, X_2 …, X_n отражают поведение процесса в многомерном координатном пространстве и определяют множество состояний Z. Состояния разделяются множеством классов C, где подмножества делятся на опасные C_1 и безопасные C_2 состояния.

Таким образом, имеется размеченная конечная обучающая выборка:

$$X=\{(x_{11},\ \dots,\ x_{n1}),\ (x_{12},\ \dots,\ x_{n2}),\ \dots,\ (x_{1m},\ \dots,\ x_{nm})\} \quad (3)$$

Требуется построить алгоритм классификации a входного вектора X_i для отображения $Z \to C$.

Предлагаемый подход

Размеченная обучающая выборка содержит значения временных рядов от регистрирующих устройств в заранее определенных состояниях и режимах работы. Известные состояния $\{z_1, \dots, z_l\} \in Z$, определены только на объектах наблюдаемых последовательностей $\{(x_{11}, \dots, x_{n1}), (x_{12}, \dots, x_{n2}), \dots, (x_{1m}, \dots, x_{nm})\}$.

От исследуемого устройства интернета вещей на интервале $t_0 \le t \le T$ наблюдается случайная векторная функция $X(t)=f(S(t),v(t))$, где в дискретные моменты времени $t_0,t_1,\dots,t_k$, регистрируется временной ряд $x_i=X(t_i)$.

Определено множество классов состояний $C=\{c_0, c_1, \dots, c_n\}$, в одном из которых в дискретный момент времени t_j может находится система.

Имеется k независимо друг от друга обученных классификаторов a_i, $i=1, \dots, k$. X – множество наборов признаков. $a_i(x_i)\to c_j \in C$ – ответ i-го классификатора. $\{P_i(c_j \mid x_i)\}_{j=0}^{n}$ – апостериорная вероятность для i-го классификатора после обучения. $w_i=\frac{1}{k}$ – весовые коэффициенты.

$a(x)=\arg\max\limits_{j=0,\dots,n}\sum\limits_{i=0}^{k} w_i P_i(c_j \mid x_i)$ – общий классификатор.

Схема модели параллельной последовательности классификаторов приведена на рисунке 1. Подобные модели могут обучаться независимо друг от друга, что дает возможность осуществлять распараллеливание процессов. Предложенный подход идентификации состояния отличается использованием технологии классификации, реализующей композиции независимо обученных алгоритмов, обрабатывающих временные ряды, отражающих функционирование устройства во время выполнения процессов, что позволяет определять состояние устройства, не увеличивая объема хранящейся информации.

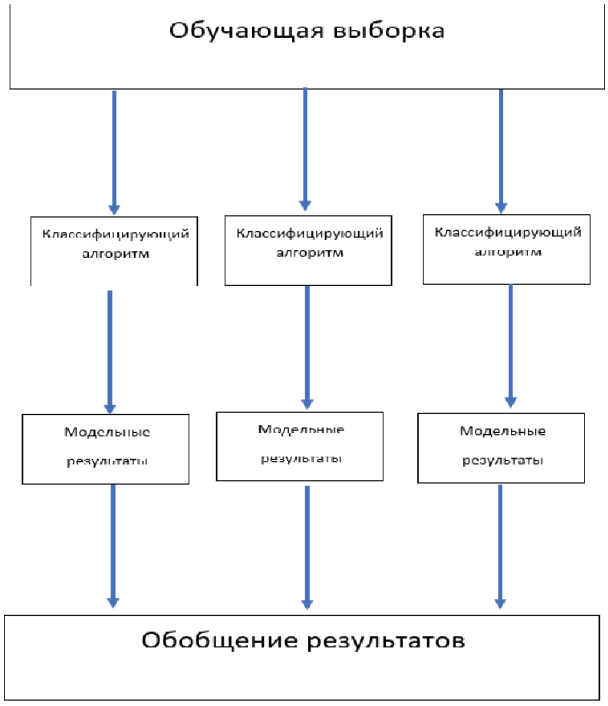

Рис. 1. Схема модели параллельной последовательности классификаторов

Использование предлагаемого подхода на первоначальном этапе предполагает «настройку» устройства в заранее заданных режимах работы, где происходит предобработка на основе обучающей выборки.

Эксперимент

Анализ приведенного подхода осуществлялся на основе эксперимента, в ходе которого производилось выявление состояния, определяемого алгоритмом обработки данных, вычислительного узла. В качестве входных данных использовались временные ряды, отражающие загрузку вычислительных ресурсов, регистрируемые программой монитором. Схема эксперимента приведена на рис. 2.

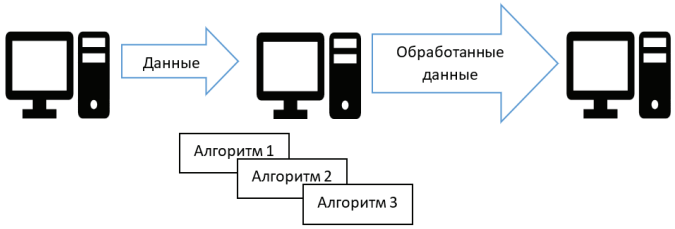

Рис. 2. Схема проведения эксперимента

На вычислительном устройстве запускались различные алгоритмы. В состоянии Z_1 функционировали только фоновые процессы. Во втором случае – узел C выступал в качестве транзитного узла, передававшего поступающую информацию без обработки (состояние Z_2). В третьей ситуации (состояние Z_3) кроме процессов приема и передачи дополнительно проводились процессы поиска заранее заданной информации (рис. 3-6).

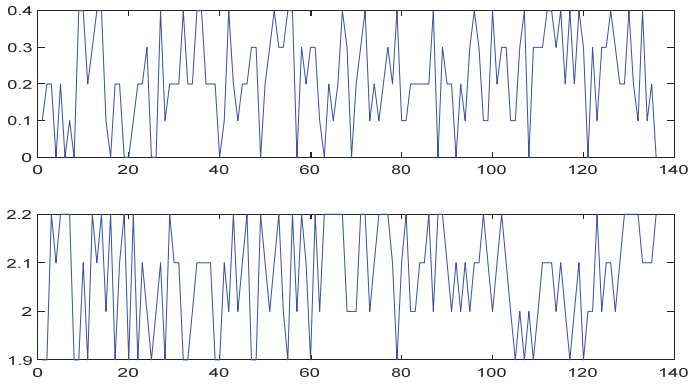

Рис. 3. Пример выборки процентной загрузки ресурсов (сверху вниз соответственно – сеть, процессор) от дискретов времени (временные отчеты от 0 до 140) для состояния Z_1

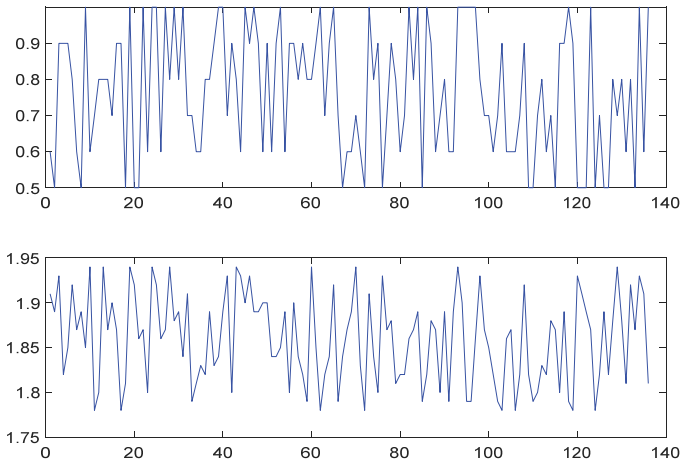

Рис. 4. Пример выборки процентной загрузки ресурсов (сверху вниз соответственно – сеть, процессор) от дискретов времени (временные отчеты от 0 до 140) для состояния Z_2

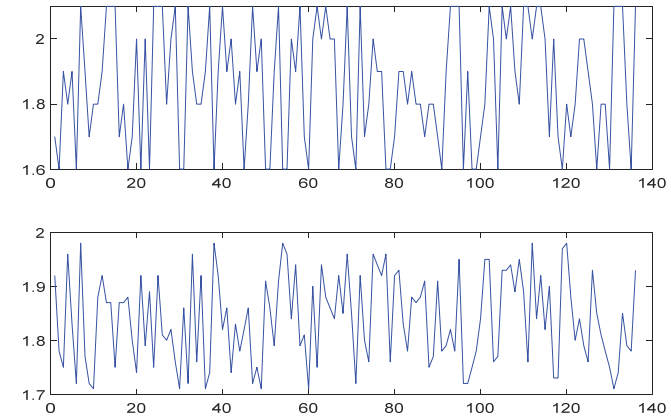

Рис. 5. Пример выборки процентной загрузки ресурсов (сверху вниз соответственно – сеть, процессор) от дискретов времени (временные отчеты от 0 до 140) для состояния Z_3

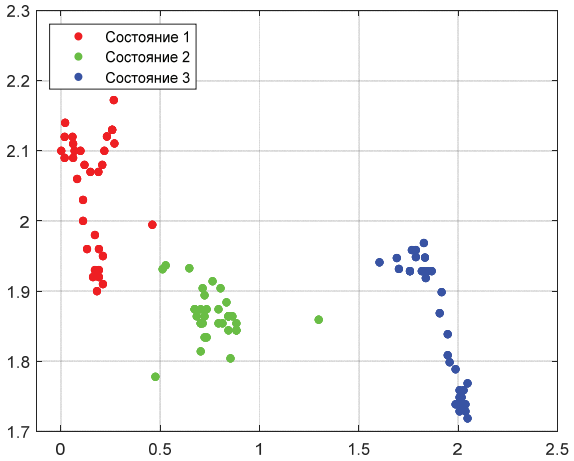

Рис. 6. Результаты состояний на двумерных координатных осях

В ходе эксперимента рассматривались алгоритмы классификации a_j входного вектора X_i для отображения $Z \rightarrow C$. Независимо друг от друга обученные и работающие классификаторы $\{a_1, a_2, \ldots, a_k\} \in a$ $k = 4$ (наивный байесовский классификатор, деревья решений, дискриминантный анализ, метод k ближайших соседей) выдавали последовательности результатов

$$Z = \{(z_{a_1}^{c_0}, z_{a_1}^{c_1}, \ldots, z_{a_1}^{c_n}), (z_{a_2}^{c_0}, z_{a_2}^{c_1}, \ldots, z_{a_2}^{c_n}), \ldots,$$
$$(z_{a_j}^{c_0}, z_{a_j}^{c_1}, \ldots, z_{a_j}^{c_n}), \ldots, (z_{a_k}^{c_0}, z_{a_k}^{c_1}, \ldots, z_{a_k}^{c_n})\}.$$

Результирующий класс c_i состояния z_i, предсказываемый каждой моделью, определяется усреднением значений вычисленных вероятностей:

$$a_{c_i} = \frac{1}{K} \sum_{k=1}^{K} w_k a_k(x_i)$$

Применим ряд «слабых» заранее обученных на размеченной выборке классификаторов a_{R_i}: наивный байесовский классификатор, деревья решений, дискриминантный анализ, метод k ближайших соседей. Области их оценки приведены на рисунке 7.

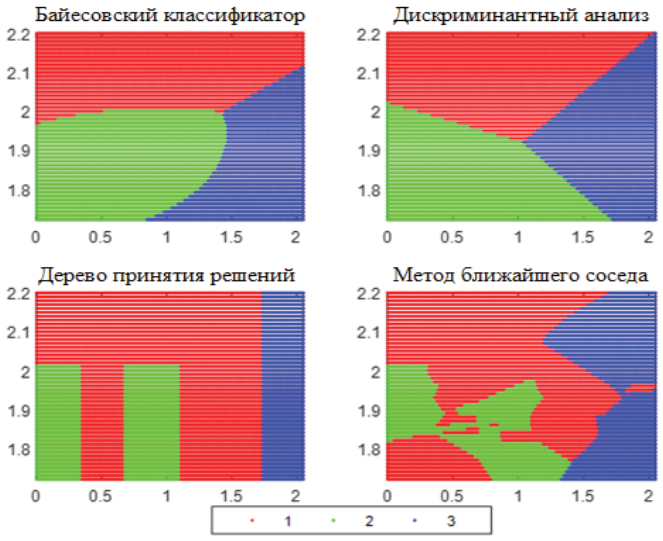

Рис. 7. Области оценки классификаторов a_{R_i}

В таблице 1 представлены вероятности ошибочной классификации, полученные в результате применения «слабых» классификаторов a_{c_i}.

Таблица 1

Вероятность ошибочно классифицированных значений выборок

	Кластер 1	Кластер 2	Кластер 3	Всего для выборки
Наивный байесовский классификатор	0,18	0,02	0,02	0,07
Дискриминантный анализ	0,16	0,02	0,02	0,07
Дерево принятия решений	0,08	0,04	0,04	0,05
Метод ближайшего соседа	0,2	0,08	0,06	0,11

На рисунке 8 представлена визуализация вероятностной оценки ошибочной классификации.

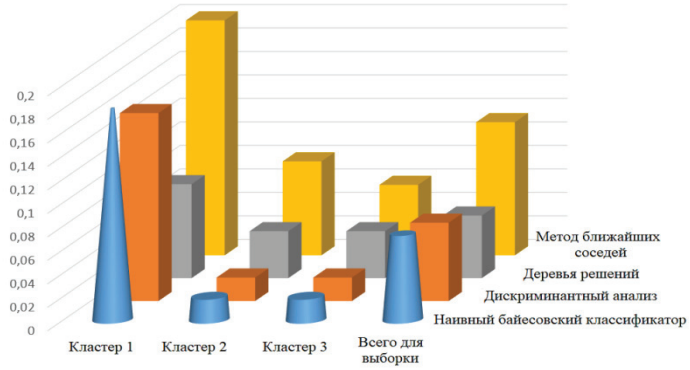

Рис. 8. Вероятности ошибочной классификации

Общая точность подхода на полученных экспериментальных данных для случая полной классификации составила 0,93. При этом необходимо отметить, что данные не проходили предобработку, не очищались от шумов, частоты дискретизации получаемых значений была относительно низкой. Таким образом, предлагаемый подход позволяет определить класс текущего состояния. Представленное решение может использоваться в качестве теоретической базы для интеграции методов машинного обучения при анализе состояния информационной безопасности IoT устройств.

Выводы

Анализ большого числа различных изменяющихся в динамике показателей с целью определения состояний устройств интернета вещей является трудоемким процессом, требующим автоматизации.

Разнородные характеристики последовательностей, полученных от регистрирующих устройств в различных режимах функционирования, обладают несбалансированностью, имеют «выбросы», которые не всегда могут правильно быть идентифицированы различными классификаторами по отдельности. Применение последовательности различных классификаторов оказывает влияние на результаты метода, позволяет обойтись без детального анализа возможных скрытых закономерностей, разбалансировки и корреляции последовательностей.

Предлагаемый подход направлен на использование ряда классификаторов, которые выдают «независимо» друг от друга ответ и усредняют ошибку «коллективным» голосованием.

Применение классификаторов в параллельном режиме обработки поступающих последовательностей дает возможность снизить время на обработку при определении класса текущего состояния.

Основным ограничением предложенного подхода является необходимость выбора синхронизированных временных рядов от регистрирующих устройств, а в случае усреднения – длин рассматриваемых интервалов.

Основным достоинством предложенного подхода является относительно небольшие требования к вычислительным ресурсам, простота его реализации, возможности по масштабированию путем добавления новых классификаторов.

Литература

1. *Farwell J. P., Rohozinski R.* Stuxnet and the Future of Cyber War. Survival, 2011. Vol. 53, no. 9. P. 23-40.

2. *Yeung D. Y., Ding Y.* Host-based intrusion detection using dynamic and static behavioral models. Pattern recognition, 2003. Vol. 36. P. 229-243.

3. *Семенов В.В., Лебедев И.С., Сухопаров М.Е.* Идентификация состояния информационной безопасности беспилотных транспортных средств с использованием искусственных нейронных сетей // Методы и технические средства обеспечения безопасности информации: Материалы 28-й научно-технической конференции 24-27 июня 2019 г. 2019. № 28. С. 46-47

4. *Igure V., Laughter S., Williams R.* Security issues in SCADA networks. Computers & Security, 2006. Vol. 25, no. 7. P. 498-506.

5. *Зикратов И. А., Зикратова Т. В., Лебедев И. С.* Доверительная модель информационной безопасности мультиагентных робототехнических систем с децентрализованным управлением // Научно-технический вестник информационных технологий, механики и оптики. 2014. № 2 (90). С. 47-52.

6. *Gao D., Reiter M., Song D.* Beyond output voting: Detecting compromised replicas using HMM-based behavioral distance. IEEE Transactions on Dependable and Secure Computing, 2009. Vol. 6, no. 2. P. 96-110.

7. *Devesh M., Kant A. K., Suchit Y. R., Tanuja P., Kumar S. N.* Fruition of cps and iot in context of Industry 4.0. Intelligent Communication, Control and Devices. Advances in Intelligent Systems and Computing, 2020. Vol. 989. P. 367-375.

8. *Семенов В.В., Лебедев И.С., Сухопаров М.Е.* Подход к классификации состояния информационной безопасности элементов киберфизических систем с использованием побочного электромагнитного излучения // Научно-технический вестник информационных технологий, механики и оптики. 2018. № 1. С. 98-105.

9. *Bevir M. K., O'Sullivan V. T., Wyatt D. G.* Computation of electromagnetic flowmeter characteristics from magnetic field data. Journal of Physics D Applied Physics, 1981. Vol. 14, no. 3. P. 373-388.

10. *Semenov V.V., Lebedev I.S., Sukhoparov M.E., Salakhutdinova K.I.* Application of an autonomous object behavior model to classify the cybersecurity state. Internet of Things, Smart Spaces, and Next Generation Networks and Systems, 2019. P. 104-112.

11. *Сошникова Л. А., Тамашевич В. Н., Усбе Г., Шефер М.* Многомерный статистический анализ в экономике: учебное пособие для вузов. М.: ЮНИТИ – Дана, 1999. 598 с.

12. *Сухопаров М.Е., Семенов В.В., Салахутдинова К.И., Лебедев И.С.* Выявление аномального функционирования устройств «Индустрии 4.0» на основе поведенческих паттернов // Проблемы информационной безопасности. Компьютерные системы. 2020. № 1 (41). С. 96-102.

13. *Бендат Д., Пирсол А.* Применение корреляционного и спектрального анализа. М.: Мир, 1983. 312 с.

14. *Засов В.А., Тарабардин М.А., Никоноров Е.Н.* Алгоритмы и устройства для идентификации входных сигналов в задачах контроля и диагностики динамических объектов // Вестник Самарского государственного аэрокосмического университета. 2009. № 2. С. 115-123.

STATE ANALYSIS INTERNET OF THINGS DEVICES BASED ON BAGGING

Mikhail E. Sukhoparov, *Saint-Petersburg federal research center of Russian science academy, Saint-Petersburg, Russia,* ***sukhoparovm@gmail.com***

Ilya S. Lebedev, *Saint-Petersburg federal research center of Russian science academy, Saint-Petersburg, Russia,* ***isl_box@mail.ru***

Abstract

The development of IoT concept makes it necessary to search and improve models and methods for analyzing the state of remote autonomous devices. Due to the fact that some devices are located outside the controlled area, it becomes necessary to develop universal models and methods for identifying the state of low-power devices from a computational point of view, using complex approaches to analyzing data coming from various information channels. The article discusses an approach to identifying IoT devices state, based on parallel functioning classifiers that process time series received from elements in various states and modes of operation. The aim of the work is to develop an approach for identifying the state of IoT devices based on time series recorded during the execution of various processes. The proposed solution is based on methods of parallel classification and statistical analysis, requires an initial labeled sample. The use of several classifiers that give an answer "independently" from each other makes it possible to average the error by "collective" voting. The developed approach is tested on a sequence of classifying algorithms, to the input of which the time series obtained experimentally under various operating conditions were fed. Results are presented for a naive Bayesian classifier, decision trees, discriminant analysis, and the k nearest neighbors method. The use of a sequence of classification algorithms operating in parallel allows scaling by adding new classifiers without losing processing speed. The method makes it possible to identify the state of the Internet of Things device with relatively small requirements for computing resources, ease of implementation, and scalability by adding new classifying algorithms.

Keywords: *state analysis, internet of things, bootstrap aggregating, discriminant analysis, state monitoring, classification algorithm, Bayesian classifier, decision trees.*

References

1. J.P. Farwell and R. Rogozinski (2011). Stuxnet and the future of cyber war. Survival. *Global Politics and Strategy*. Vol. 53, Issue 1, pp. 23-40. DOI: 10.1080/00396338.2011.555586.
2. D.Y. Yeung and Y. Ding (2003). Host-based intrusion detection using dynamic and static behavioral models. *Pattern recognition*. Vol. 36, Issue 1, pp. 229-243. DOI: 10.1016/S0031-3203(02)00026-2.
3. V. Igure, S. Laughter, and R. Williams (2006). Security issues in SCADA networks. *Computers & Security*. Vol. 25, Issue 7, pp. 498-506. DOI: 10.1016/j.cose.2006.03.001.
4. V.V. Semenov, I.S. Lebedev and M.E. Sukhoparov (2019). State identification of information security of unmanned vehicles using artificial neural networks. Methods and information security engineering. *Proceedings of the 28th Scientific and Technical Conference*, June 24-27, 2019. Issue 28, pp. 46-47 (in Russian).
5. I.A. Zikratov, T.V. Zikratova and I.S. Lebedev (2014). Trusted model of information security of multi-agent robotics systems with decentralized control. *Scientific and Technical Journal of Information Technologies. Mechanics and Optics*. Issue 2 (90), pp. 47-52 (in Russian).
6. D. Gao, M.K. Reiter, and D. Song (2009). Beyond output voting: Detecting compromised replicas using HMM-based behavioral distance. *IEEE Transactions on Dependable and Secure Computing*. Vol. 6, Issue 2, pp. 96-110. DOI: 10.1109/TDSC.2008.39.
7. M. Devesh, A.K. Kant, Y.R. Suchit, P. Tanuja and S.N. Kumar (2020). Fruition of CPS and IoT in context of Industry 4.0. Intelligent Communication, Control and Devices. *Advances in Intelligent Systems and Computing*. Vol. 989, pp. 367-375.
8. M.K. Bevir, V.T. Osullivan and D.G. Wyatt (1981). Computation of electromagnetic flowmeter characteristics from magnetic field data. *Journal of Physics D: Applied Physics*. Vol. 14, Issue 3, pp. 373-388. DOI: 10.1088/0022-3727/14/3/007.
9. V.V. Semenov, I.S. Lebedev, M.E. Sukhoparov and K.I. Salakhutdinova (2019). Application of an autonomous object behavior model to classify the cybersecurity state. Internet of Things, Smart Spaces, and Next Generation Networks and System, pp. 104-112. DOI: 10.1007/978-3-030-30859-9_9.
10. V.V. Semenov, I.S. Lebedev and M.E. Sukhoparov (2018). Approach to classification of the information security state of elements of cyber physical systems by applying side electromagnetic radiation. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*. Vol. 18, Issue 1, pp. 98-105 (in Russian). DOI: 10.17586/2226-1494-2018-18-1-98-105.
11. L.A. Soshnikova, V.N. Tamashevich, G. Ushbe and M. Shefer (1999). Multivariate statistical analysis in economics: manual for graduate students. Moscow: UNITI – Dana. 598 p. (in Russian).
12. M.E. Sukhoparov, V.V. Semenov, K.I. Salakhutdinova and I.S. Lebedev (2020). Detection of abnormal functioning of Industry 4.0 devices based on behavioral patterns. Information Security Problems. Computer Systems. Issue 1 (41), pp. 96-102. (in Russian).
13. Dzh. Bendat and A. Pirsol (1983). Application of correlation and spectral analysis. Translation from English. Moscow: Mir Publ. 312 p. (in Russian).
14. V.A. Zasov, M.A. Tarabardin and E.N. Nikonorov (2009). Algorithms and devices for identification of input signals in tasks of control and diagnostics of dynamic objects. *VESTNIK of Samara University. Aerospace and Mechanical Engineering*. Issue 2, pp. 115-123 (in Russian).

Information about authors:

Mikhail E. Sukhoparov, *Senior researcher of Intelligent Systems Laboratory, Saint-Petersburg federal research center of Russian science academy, Saint-Petersburg, Russia*

Ilya S. Lebedev, *Head of Intelligent Systems Laboratory, Saint-Petersburg federal research center of Russian science academy, Saint-Petersburg, Russia*