DOI: 10.15514/ISPRAS–2020–32(5)–6

Синтез модели машинного обучения для обнаружения компьютерных атак на основе набора данных CICIDS2017

М.Н. Горюнов, ORCID: 0000-0003-0284-690X <max.gor@mail.ru>
А.Г. Мацкевич, ORCID: 0000-0001-9557-3765 <mag3d.78@gmail.com>
Д.А. Рыболовлев, ORCID: 0000-0003-4524-655X <dmitrij-rybolovlev@yandex.ru>
Академия ФСО России,
302015, Россия, г. Орел, ул. Приборостроительная, д. 35

Аннотация. В работе рассмотрены вопросы построения и практической реализации модели обнаружения компьютерных атак на основе методов машинного обучения. Среди доступных публичных наборов данных выбран один из наиболее актуальных – CICIDS2017. Для рассматриваемого набора данных подробно разработаны процедуры предварительной обработки данных и сэмплирования. При проведении экспериментов для сокращения времени вычислений в обучающей выборке оставлен единственный класс компьютерных атак – веб-атаки (brute force, XSS, SQL injection). Последовательно описана процедура формирования признакового пространства, позволившая существенно снизить его размерность – с 85 до 10 наиболее значимых признаков. Произведена оценка качества десяти наиболее распространенных моделей машинного обучения на полученной предобработанной подвыборке данных. Среди моделей (алгоритмов), которые продемонстрировали наилучшие результаты (k-nearest neighbors, decision tree, random forest, AdaBoost, logistic regression), с учетом минимального времени выполнения обоснован выбор модели «случайный лес». На этапе настройки и обучения выбранной модели осуществлен квазиоптимальный подбор гиперпараметров, что позволило добиться повышения качества модели в сравнении с ранее опубликованными результатами исследований. Произведена апробация синтезированной модели обнаружения атак на реальном сетевом трафике, показавшая ее состоятельность только при условии обучения на данных, собираемых в конкретной защищаемой сети, в виду зависимости ряда значимых признаков от физической структуры сети и настроек используемого оборудования. Сделан вывод о возможности применения методов машинного обучения для обнаружения компьютерных атак с учетом указанных ограничений.

Ключевые слова: информационная безопасность; система обнаружения атак; машинное обучение; дерево решений; случайный лес; сетевой трафик; компьютерная атака

Для цитирования: Горюнов М.Н., Мацкевич А.Г., Рыболовлев Д.А. Синтез модели машинного обучения для обнаружения компьютерных атак на основе набора данных CICIDS2017. Труды ИСП РАН, том 32, вып. 5, 2020 г., стр. 81-94. DOI: 10.15514/ISPRAS–2020–32(5)–6

Synthesis of a Machine Learning Model for Detecting Computer Attacks Based on the CICIDS2017 Dataset

M.N. Goryunov, ORCID: 0000-0003-0284-690X <max.gor@mail.ru>
A.G. Matskevich, ORCID: 0000-0001-9557-3765 <mag3d.78@gmail.com>
D.A. Rybolovlev, ORCID: 0000-0003-4524-655X <dmitrij-rybolovlev@yandex.ru>
The Academy of Federal Security Guard Service of the Russian Federation,
35, Priborostroitelnaya st., Oryol, 302015, Russia

Abstract. The paper deals with the construction and practical implementation of the model of computer attack detection based on machine learning methods. Among available public datasets one of the most relevant was chosen - CICIDS2017. For this dataset, the procedures of data preprocessing and sampling were developed in detail. In order to reduce computation time, the only class of computer attacks (brute force, XSS, SQL injection) was left in the training set. The procedure of feature space construction is described sequentially, which allowed to significantly reduce its dimensions - from 85 to 10 most important features. The quality assessment of ten most common machine learning models on the obtained pre-processed dataset was made. Among the models (algorithms) that demonstrated the best results (k-nearest neighbors, decision tree, random forest, AdaBoost, logistic regression), taking into account the minimum time of execution, the choice of random forest model was justified. A quasi-optimal selection of hyper parameters was carried out, which made it possible to improve the quality of the model in comparison with the previously published research results. The synthesized model of attack detection was tested on real network traffic. The model has shown its validity only under the condition of training on data collected in a specific network, since important features depend on the physical structure of the network and the settings of the equipment used. The conclusion was made that it is possible to use machine learning methods to detect computer attacks taking into account these limitations.

Keywords: information security; intrusion detection system; machine learning; decision tree; random forest; network traffic; computer attack

For citation: Goryunov M.N., Matskevich A.G., Rybolovlev D.A. Synthesis of a machine learning model for detecting computer attacks based on the CICIDS2017 dataset. Trudy ISP RAN/Proc. ISP RAS, vol. 32, issue 5, 2020, pp. 81-94 (in Russian). DOI: 10.15514/ISPRAS–2020–32(5)–6

1. Введение

Бурное развитие информационных технологий в современном мире, расширение спектра и увеличение количества предоставляемых потребителям инфокоммуникационных услуг неизбежно сопровождается ростом числа угроз и факторов, приводящих к нарушению функционирования информационных систем и компьютерных сетей. По этой причине государственные институты и коммерческие компании уделяют повышенное внимание проблемам информационной безопасности и, как следствие, вопросам развития применяемых методов и средств обнаружения компьютерных атак [1].

В настоящее время основным методом выявления компьютерных атак, применяемым во всех современных средствах защиты информации, является сигнатурный анализ. Однако данный подход не позволяет обнаруживать новые виды деструктивных воздействий [2], что делает актуальным задачу разработки эвристических методов, способных детектировать ранее неизвестные типы атак [3].

Проведенный анализ ряда опубликованных на данный момент исследований [3-6] подтверждает возможность применения технологий машинного обучения для решения задач обнаружения компьютерных атак. Данное обстоятельство обусловливает целесообразность проведения прикладных исследований в указанной области, направленных на выработку конкретных предложений по построению моделей обнаружения и перспектив их практической реализации.

Цель исследования состоит в разработке модели машинного обучения для построения системы обнаружения компьютерных атак. Ее достижение предполагает решение

следующих основных задач: выбор обучающего набора данных, оценка значимости признаков и формирование признакового пространства, обоснование выбора модели машинного обучения и подбор квазиоптимальных параметров модели, оценка качества и апробация модели в реальных условиях. Новизна работы заключается в разработке макета системы обнаружения атак на основе современной модели машинного обучения и экспериментальной проверке применимости предлагаемых решений.

2. Постановка задачи и релевантные работы

Вопросы применения методов машинного обучения для обнаружения компьютерных атак активно обсуждаются в последние годы, при этом важным аспектом исследуемой предметной области является оценка возможности практической реализации разрабатываемых алгоритмов. По указанной тематике опубликовано достаточное количество работ, которые могут служить основой дальнейших исследований.

В статье [4] для решения задачи классификации и фильтрации сетевого трафика предложено использование модели типа «случайный лес» (random forest). В ходе экспериментов получены оценки эффективности работы предложенного алгоритма классификации в условиях наличия и отсутствия фонового сетевого трафика. В результатах исследования отмечено, что используемая модель машинного обучения демонстрирует высокую эффективность в отложенном режиме анализа (offline), но при обработке в реальном времени точность классификации снижается по причине высокой временной сложности и необходимости снижения сложности модели для соблюдения требований оперативности. Вместе с тем в работе не уточняются итоговые настройки используемой модели и не подтверждается их оптимальность.

В работе [5] рассматривается применение технологий нейронных сетей для обнаружения ботнет-атак. Предлагаемая модель (многослойный персептрон), обученная на публичном наборе данных CSE-CIC-IDS2018, демонстрирует на тестовых данных высокое качество обнаружения – близкое к единице значение F1-меры. Однако авторы не раскрывают особенностей формирования тестового набора данных и не оценивают возможность переобучения модели.

В исследовании [6] проводится экспериментальное сравнение семи различных моделей машинного обучения, используемых для обнаружения компьютерных атак. Рассмотрены алгоритмы: Naive Bayes, QDA, Random Forest, ID3, AdaBoost, MLP и K Nearest Neighbors. Обучение и тестирование моделей выполнено на публичном наборе данных CICIDS2017, при этом предварительно был проведен анализ значимости признаков и выполнено сокращение размерности признакового пространства. В работе получены высокие показатели точности обнаружения атак, однако предлагаемые решения не апробировались на реальном сетевом трафике. Кроме того, рассматриваемые в статье модели машинного обучения использовались с параметрами по умолчанию, что не позволяет сделать вывод о возможности оптимизации параметров и повышении точности обнаружения.

В отмеченных выше работах подробно описаны варианты реализации технологий машинного обучения, приведены оценки точности обнаружения атак в приложениях информационной безопасности. Однако опубликованные результаты носят недостаточно полный и системный характер с точки зрения оценки практической применимости, оставляя без ответов вопросы оптимальной настройки параметров моделей, апробации и применения предварительно обученных моделей на сетях с отличными характеристиками, встраивания разрабатываемых программных модулей в действующие системы и комплексы, формализации требований к производительности аппаратно-программной платформы и др.

Решаемая в данном исследовании задача заключается в практической реализации макета системы обнаружения атак на основе модели машинного обучения, оптимальном

(квазиоптимальном) выборе гиперпараметров модели и апробации предлагаемых решений на реальной сетевой инфраструктуре.

Макет разработан на языке Python с использованием свободно распространяемого пакета scikit-learn, исходный код проекта доступен для выполнения в среде Google Colaboratory: https://colab.research.google.com/github/infosecdemos/ml-2020/blob/master/ml-ids/web-attack-detection.ipynb.

3. Формирование признакового пространства

Для обучения системы обнаружения атак среди доступных публичных наборов данных (DARPA1998, KDD1999, ISCX2012, ADFA2013 и др.) выбран один из наиболее актуальных – «Intrusion Detection Evaluation Dataset» CICIDS2017 [8, 9]. Набор данных CICIDS2017 подготовлен Канадским институтом кибербезопасности по результатам анализа сетевого трафика в изолированной среде, в которой моделировались действия 25 легальных пользователей, а также вредоносные действия нарушителей [10]. Набор объединяет более 50 Гб «сырых» данных в формате PCAP и включает 8 предобработанных файлов в формате CSV, содержащих размеченные сессии с выделенными признаками в разные дни наблюдения. Краткое описание файлов и количественный состав набора данных представлены в табл. 1, 2.

Табл. 1. Описание файлов набора данных CICIDS2017
Table 1. Description of CICIDS2017 dataset files

№	Название файла	Содержащиеся атаки
1	Monday-WorkingHours.pcap_ISCX.csv	Benign (обычный трафик)
2	Tuesday-WorkingHours.pcap_ISCX.csv	Benign, FTP-Patator, SSH-Patator
3	Wednesday-workingHours.pcap_ISCX.csv	Benign, DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS slowloris, Heartbleed
4	Thursday-WorkingHours-Morning-WebAttacks.pcap_ ISCX.csv	Benign, Web Attack – Brute Force, Web Attack – Sql Injection, Web Attack – XSS
5	Thursday-WorkingHours-Afternoon-Infilteration.pcap_ ISCX.csv	Benign, Infiltration
6	Friday-WorkingHours-Morning.pcap_ISCX.csv	Benign, Bot
7	Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	Benign, PortScan
8	Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	Benign, DDoS

Табл. 2. Количественный состав набора данных CICIDS2017
Table 2. Number of attacks in the CICIDS2017 dataset

№	Тип записи	Количество записей
1	BENIGN	2359087
2	DoS Hulk	231072
3	PortScan	158930
4	DDoS	41835
5	DoS GoldenEye	10293
6	FTP-Patator	7938
7	SSH-Patator	5897
8	DoS slowloris	5796
9	DoS Slowhttptest	5499
10	Bot	1966
11	Infiltration	36
12	Heartbleed	11
13	Web Attack – Brute Force	1507
14	Web Attack – XSS	652
15	Web Attack – SQL Injection	21

3.1 Предварительная обработка данных

При проведении экспериментов для сокращения времени вычислений в обучающей выборке оставлен единственный класс компьютерных атак – веб-атаки (Brute Force, XSS, SQL Injection). Для этого использовался набор данных WebAttacks, подготовленный на основе обработки файла Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv. Набор WebAttacks включает 458968 записей, из которых 2180 относятся к веб-атакам, остальные – к нормальному трафику.

Каждая запись в наборе данных WebAttacks представляет собой сетевую сессию и характеризуется 84 признаками.

Этап предварительной обработки подвыборки веб-атак WebAttacks набора данных CICIDS2017 включает следующую последовательность действий:

1) исключение признака «Fwd Header Length.1» (признаки «Fwd Header Length» и «Fwd Header Length.1» являются идентичными);
2) удаление записей с null значениями идентификатора сессии «Flow ID» (из 458968 записей после удаления осталось 170366 записей);
3) замена нечисловых значений признаков «Flow Bytes/s», «Flow Packets/s» значениями -1;
4) замена неопределенных значений (NaN) и бесконечных значений значениями -1;
5) приведение строковых значений признаков «Flow ID», «Source IP», «Destination IP», «Timestamp» к числовым значениям методом label encoding;
6) кодирование ответов в обучающей выборке в соответствии с правилом: 0 – «нет атаки», 1 – «есть атака».

3.2 Сэмплирование

Предобработанный набор данных WebAttacks является несбалансированным: при общем количестве записей 170366 класс «нет атаки» объединяет 168186 экземпляров, класс «есть атака» – 2180 экземпляров [9]. Для устранения дисбаланса классов применяется метод случайного сэмплирования (субдискретизация, undersampling), заключающийся в удалении случайно выбранных экземпляров класса «нет атаки». Целевое соотношение количества экземпляров классов «нет атаки» и «есть атака» составляет 70% / 30%.

3.3 Оценка значимости и отбор признаков

При разработке модели машинного обучения важным является решение о том, какие признаки следует использовать в качестве входных данных для обучающего алгоритма [11]. Отбор признаков при формировании признакового пространства является обязательной процедурой как на подготовительном этапе (предшествующем обучению), так и на этапе оценки полученных результатов и последующей корректировки обучающей выборки и/или гиперпараметров модели [12].

Предварительно из признакового пространства исключены признаки «Flow ID», «Source IP», «Source Port», «Destination IP», «Destination Port», «Protocol», «Timestamp» в предположении, что признаки «формы» (соответствующие статистикам сетевого трафика) являются более значимыми для общего случая. Кроме того, исключаемые признаки адресации могут быть относительно легко подделаны злоумышленником и не должны учитываться при обучении [6].

Анализ значимости признаков выполнен с помощью встроенного механизма метода sklearn.ensemble.RandomForestClassifier (атрибут feature_importances_), реализующего энтропийный подход к оценке важности признаков.

Первые результаты оценки значимости показали сильную взаимосвязь признаков «Init_Win_bytes_backward», «Init_Win_bytes_forward» с метками класса в обучающей

выборке, что может свидетельствовать о допущенных погрешностях при формировании набора данных. Указанные признаки исключены из признакового пространства.

Итоговые результаты анализа значимости представлены на рис. 1, список ограничен первыми двадцатью признаками.

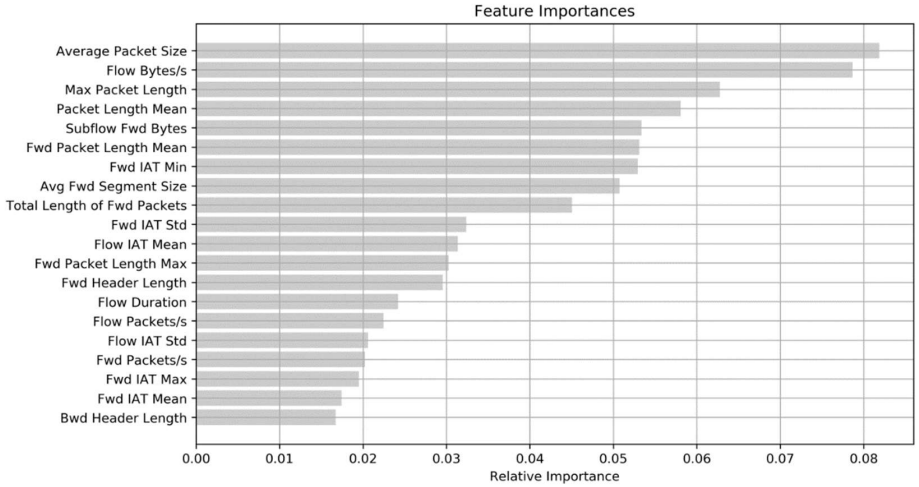

Рис. 1. Результаты анализа важности признаков
Fig. 1. Feature importance analysis results

3.4 Сокращение признакового пространства

На рис. 2 представлена корреляционная матрица с линейными коэффициентами корреляции (коэффициентами корреляции Пирсона), рассчитанными для всех пар двадцати наиболее значимых признаков. Насыщенность цвета заливки пропорциональна значению коэффициента корреляции.

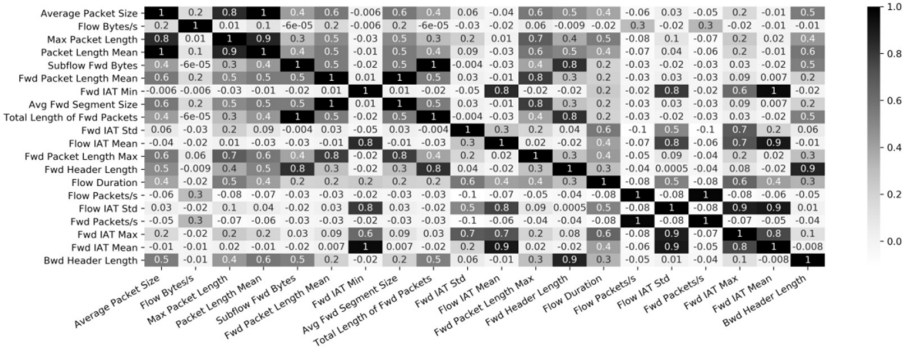

Рис. 2. Результаты корреляционного анализа двадцати наиболее значимых признаков
Fig. 2. The results of the correlation analysis of the twenty most significant features

Проведенный корреляционный анализ показал сильную зависимость между парами признаков:

1) «Average Packet Size» и «Packet Length Mean»;
2) «Subflow Fwd Bytes» и «Total Length of Fwd Packets»;

3) «Fwd Packet Length Mean» и «Avg Fwd Segment Size»;
4) «Flow Duration» и «Fwd IAT Total»;
5) «Flow Packets/s» и «Fwd Packets/s»;
6) «Flow IAT Max» и «Fwd IAT Max».

По результатам корреляционного анализа из признакового пространства исключены следующие признаки: «Packet Length Mean», «Subflow Fwd Bytes», «Avg Fwd Segment Size», «Fwd IAT Total», «Fwd Packets/s», «Fwd IAT Max».

После исключения признаков с наименьшей значимостью признаковое пространство сокращено до объединения 10 признаков:

1) «Average Packet Size», средняя длина поля данных пакета TCP/IP (далее – длина пакета);
2) «Flow Bytes/s», скорость потока данных;
3) «Max Packet Length», максимальная длина пакета;
4) «Fwd Packet Length Mean», средняя длина переданных в прямом направлении пакетов;
5) «Fwd IAT Min», минимальное значение межпакетного интервала (IAT, inter-arrival time) в прямом направлении;
6) «Total Length of Fwd Packets», суммарная длина переданных в прямом направлении пакетов;
7) «Fwd IAT Std», среднеквадратическое отклонение значения межпакетного интервала в прямом направлении пакетов;
8) «Flow IAT Mean», среднее значение межпакетного интервала;
9) «Fwd Packet Length Max», максимальная длина переданного в прямом направлении пакета;
10) «Fwd Header Length», суммарная длина заголовков пакетов, переданных в прямом направлении.

4. Выбор, настройка, обучение модели

4.1 Выбор модели

На этапе выбора модели для решения рассматриваемой задачи классификации была произведена оценка качества наиболее распространенных моделей машинного обучения на сбалансированной и предобработанной подвыборке веб-атак WebAttacks набора данных CICIDS2017.

Качество ответов классификаторов (моделей) сравнивалось с использованием следующих метрик:

- доля правильных ответов (accuracy);
- точность (precision, насколько можно доверять классификатору);
- полнота (recall, как много объектов класса «есть атака» определяет классификатор);
- F1-мера (F1-measure, гармоническое среднее между точностью и полнотой).

При определении значений метрик качества используются элементы матрицы ошибок (confusion matrix), соответствующие количеству правильных и неправильных ответов по результатам тестирования классификатора (табл. 3).

Табл.3. Описание матрицы ошибок классификатора
Table 3. Classifier error matrix description

	Ответ классификатора: класс 0, «нет атаки»	**Ответ классификатора: класс 1, «есть атака»**
Правильный ответ: класс 0, «нет атаки»	TN	FP
Правильный ответ: класс 1, «есть атака»	FN	TP

TP (True Positive) обозначает истинно-положительный ответ, TN (True Negative) – истинно-отрицательный ответ, FP (False Positive) – ложно-положительный ответ (ложное срабатывание, ошибка первого рода), FN (False Negative) – ложно-отрицательный ответ (пропуск атаки, ошибка второго рода). С учетом приведенных обозначений используемые метрики качества определяются следующими выражениями:

Accuracy = TP + TN / (TP + FP + FN + TN);
Precision = TP / (TP + FP);
Recall = TP / (TP + FN);
F1 = 2 * Precision * Recall / (Precision + Recall).

Для сравнения были выбраны следующие модели (алгоритмы) машинного обучения (в скобках указывается сокращенное обозначение и соответствующая реализация модели из состава пакета scikit-learn):

1) метод k ближайших соседей (KNN, sklearn.neighbors.KNeighborsClassifier);
2) метод опорных векторов (SVM, sklearn.svm.SVC);
3) дерево решений (CART, алгоритм обучения CART, sklearn.tree.DecisionTreeClassifier);
4) случайный лес (RF, sklearn.ensemble.RandomForestClassifier);
5) модель адаптивного бустинга над решающим деревом (AdaBoost, sklearn.ensemble.AdaBoostClassifier);
6) логистическая регрессия (LR, sklearn.linear_model.LogisticRegression);
7) байесовский классификатор (NB, sklearn.naive_bayes.GaussianNB);
8) линейный дискриминантный анализ (LDA, sklearn.discriminant_analysis.LinearDiscriminantAnalysis);
9) квадратичный дискриминантный анализ (QDA, sklearn.discriminant_analysis.QuadraticDiscriminantAnalysis);
10) Многослойный персептрон (MLP, sklearn.neural_network.MLPClassifier).

Оценка качества классификаторов производилась на сбалансированной и предобработанной подвыборке веб-атак WebAttacks набора данных CICIDS2017 (соотношение нормального и аномального трафика 70% / 30%, 20 наиболее значимых признаков). В табл. 4 приведены полученные значения метрик качества, усредненные по результатам 5 итераций кросс-валидации.

Табл. 4. Результаты оценки качества классификаторов
Table 4. Results of evaluation of quality of classifiers

Модель (алгоритм)	**Accuracy**	**Precision**	**Recall**	**F1**	**Время выполнения, с**
KNN	0.971	0.942	0.961	0.969	4.57
SVM	0.705	0.669	0.036	0.602	176.04
CART	0.975	0.973	0.946	0.969	1.53
RF	0.971	0.978	0.943	0.970	1.14
AdaBoost	0.978	0.962	0.965	0.973	23.40
LR	0.955	0.939	0.914	0.963	15.80
Naive Bayes	0.722	0.520	0.956	0.754	0.47
LDA	0.939	0.921	0.872	0.941	2.23
QDA	0.872	0.978	0.597	0.949	1.28
MLP	0.904	0.921	0.912	0.776	93.83

Наилучшие результаты демонстрируют модели (алгоритмы) KNN, CART, RF, AdaBoost, LR. Принимая во внимание минимальное время выполнения, применение модели «случайный лес» (RF) для решения поставленной задачи является обоснованным выбором.

4.2 Настройка и обучение модели

В качестве используемого классификатора в исследовании выбрана модель типа «случайный лес» RandomForestClassifier свободно распространяемого пакета scikit-learn, реализующая построение ансамбля деревьев решений (decision tree) и характеризуемая потенциально высокой обобщающей способностью при решении рассматриваемого класса задач [7].

Для проведения квазиоптимального подбора гиперпараметров модели и оценки обобщающей способности используются четыре метрики качества: доля правильных ответов, точность, полнота, F1-мера.

Среди настраиваемых гиперпараметров выбраны следующие: количество деревьев в лесу (n_estimators), минимальное число объектов в одном листе дерева (min_samples_leaf), максимальная глубина дерева (max_depth), максимальное количество признаков для одного дерева (max_features).

Пример результатов подбора одного гиперпараметра (max_depth) при фиксированных значениях других гиперпараметров (n_estimators, min_samples_leaf, max_features) представлен на рис. 3 в виде зависимости метрики качества (F1-меры) от значения настраиваемого параметра (max_depth).

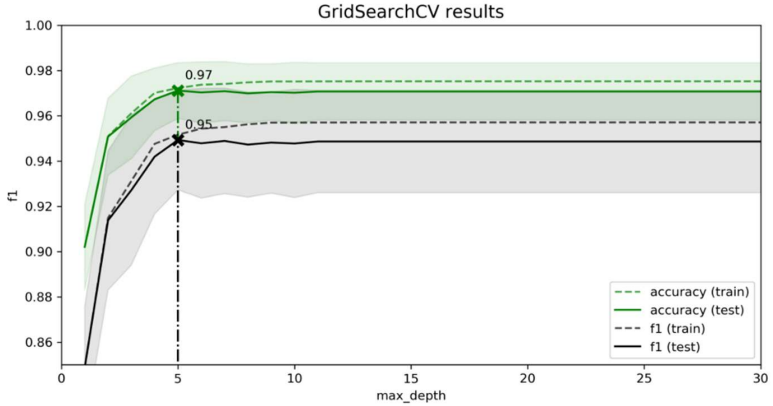

Рис. 3. Графики зависимости метрики качества (F-мера) модели «случайный лес» от максимальной глубины дерева в лесу (max_depth)
Fig. 3. Graphs of the dependence of the F-measure on the max_depth hyper parameter

Проведенный экспертный анализ дополнен результатами встроенного метода оптимизации параметров GridSearchCV библиотеки scikit-learn, итоговые значения параметров модели «случайный лес» представлены ниже:

```
RandomForestClassifier (bootstrap=True,
          class_weight=None, criterion='gini',
          max_depth=17, max_features=10, max_leaf_nodes=None,
          min_impurity_decrease=0.0, min_impurity_split=None,
          min_samples_leaf=3, min_samples_split=2,
          min_weight_fraction_leaf=0.0, n_estimators=50,
          n_jobs=None, oob_score=False, random_state=1, verbose=0,
          warm_start=False)
```

5. Тестирование и апробация модели

Настроенная и обученная модель RandomForestClassifier на валидационной выборке позволила получить оценку полноты (recall) 0.961 и F1-меры 0.971 (запуск № 1 в протоколе

эксперимента, табл. 5). Достигнутый результат свидетельствует о возможности повышения точности модели за счет квазиоптимального подбора гиперпараметров (в сравнении с результатами исследования [6], recall 0.94 и F1-мера 0.94).

Для апробации модели на реальной сетевой инфраструктуре разработан сетевой анализатор – сниффер (язык программирования C#, среда разработки Microsoft Visual Studio 2019). Анализатор позволяет перехватить передаваемый сетевой трафик и с использованием алгоритмов реконструкции TCP сессий свободно распространяемых программных продуктов WireShark и TCP Session Reconstruction Tool выделить отдельные сессии. Для каждой сохраненной сессии сниффер на основе алгоритма CICFLOWMETER [13] выделяет признаки и таким образом формирует набор данных.

В качестве атакуемого веб-приложения использовалась разработанная консоль администратора безопасности (язык программирования PHP) с единственным включенным модулем авторизации, функционирующая под управлением веб-сервера Apache.

Нормальный трафик соответствовал запросам легальных пользователей на подключение к консоли администратора и авторизацию. Вредоносный (аномальный) трафик моделировался программным средством OWASP ZAP и включал три типа атак: Brute Force, XSS, SQL Injection. Соотношение нормального и аномального трафика в реальном тестовом наборе данных составило 70% / 30%.

Табл. 5. Протокол эксперимента
Table 5. Experiment protocol

Эксперимент / Характеристика	Запуск №1	Запуск №2	Запуск №3
Этап обучения модели			
Используемый набор данных	Сбалансированная и предобработанная подвыборка веб-атак WebAttacks набора данных CICIDS2017. 7267 записей, из них 5087 экземпляров класса «нет атаки» и 2180 экземпляров класса «есть атака».		Сформированный набор данных, соответствующих реальному сетевому трафику
Обучающая выборка	70% записей используемого набора данных		70% записей используемого набора данных
Модель машинного обучения	RandomForestClassifier(max_depth=17, max_features=10, min_samples_leaf=3, n_estimators=50)		RandomForestClassifier (max_depth=None, max_features='auto', min_samples_leaf=1, n_estimators=250)
Признаковое пространство	1. Average Packet Size 2. Flow Bytes/s 3. Max Packet Length 4. Fwd Packet Length Mean 5. Fwd IAT Min 6. Total Length of Fwd Packets 7. Fwd IAT Std 8. Flow IAT Mean 9. Fwd Packet Length Max 10. Fwd Header Length		Flow Packets/s Flow IAT Max Bwd Packet Length Min Flow Duration Flow IAT Mean Flow IAT Std Average Packet Size Fwd Packet Length Max Total Packets Fwd Header Length
Этап тестирования модели			
Тестовая выборка	30% записей используемого набора данных. Тестовая и обучающая выборка не имеют пересечений.	100% записей сформированного набора данных, соответствующих реальному сетевому трафику	30% записей используемого набора данных. Тестовая и обучающая выборка не имеют пересечений.

Значения метрик качества			
Accuracy	0.983	0.456	0.858
Precision	0.982	0.812	0.812
Recall	0.961	0.033	0.966
F1	0.971	0.064	0.882

Проведенные эксперименты на сформированном наборе данных (запуски № 2, № 3 в протоколе эксперимента, табл. 5) показали невозможность применения модели, обученной на наборе данных CICIDS2017, по следующим причинам.

1. Анализ обучающей выборки (набор данных CICIDS2017) показывает, что характер моделируемых компьютерных атак в исследовании [10] отличается от реального. Так, атаки типа Brute Force присутствуют в сессиях с максимальными скоростями до 10 Кбит/с, что не соответствует случаям применения автоматизированных средств перебора паролей.
2. Среди десяти признаков с наибольшей значимостью четыре признака – «Flow Bytes/s» (скорость потока данных), «Fwd IAT Min» (минимальное значение межпакетного интервала в прямом направлении), «Flow IAT Std» (среднеквадратическое отклонение значения межпакетного интервала), «Flow IAT Mean» (среднее значение межпакетного интервала) – непосредственно зависят от физической структуры сети, в которой производится сбор сетевого трафика, а также настроек сетевого оборудования. В обучающем наборе данных сессии с признаками веб-атак записаны с низкими значениями скорости потока и высокими значениями межпакетных интервалов, что не соответствует характеристикам реальной сетевой инфраструктуры (сеть Ethernet 100 Мбит/с).

Полученные результаты свидетельствуют о необходимости обучения предлагаемой модели машинного обучения на наборе данных, полученном на основе анализа сетевого трафика в защищаемой сети. В противном случае, при использовании предобученной модели обязательным является соответствие физической структуры защищаемой сети и сети, в которой обучалась модель, а также настроек сетевого оборудования.

Оценка вычислительной сложности производилась косвенным способом: разработанный в среде Jupyter Notebook макет системы обнаружения веб-атак запускался на персональном компьютере (процессор Intel Core i5-2300 CPU @ 2300 ГГц, ОЗУ 8 Гб) в режиме обнаружения. Тестовый набор данных содержал около 70000 записанных сессий, время обнаружения составило 0,74669 с. Таким образом скорость обнаружения веб-атак оценивается величиной порядка 100000 сессий в секунду.

6. Заключение

Для оценки применимости современных методов машинного обучения в системах обнаружения компьютерных атак в исследовании проведен эксперимент с настройкой модели «случайный лес», обучением на публичном наборе данных CICIDS2017 и тестированием в реальных условиях.

Настройка параметров выбранного классификатора RandomForestClassifier свободно распространяемого пакета scikit-learn позволила на валидационной выборке получить оценку полноты (recall) 0.961 и F1-меры 0.971 для набора данных CICIDS2017, а также 0.966 и 0.882 соответственно для сформированного в исследовании набора данных.

Невозможность применения модели на тестовой выборке, полученной на основе анализа сетевого трафика в реальной компьютерной сети, объясняется тем, что при формировании обучающей выборки характер моделируемых компьютерных атак отличался от реального. Кроме того, среди десяти признаков с наибольшей значимостью четыре признака (скорость потока данных, минимальное значение межпакетного интервала в прямом направлении, среднеквадратическое отклонение значения межпакетного интервала, среднее значение

межпакетного интервала) непосредственно зависят от физической структуры сети, в которой производился сбор сетевого трафика, а также настроек сетевого оборудования. Отличия в физической структуре сетей и настройках оборудования приводят к возникновению ошибок классификатора и снижению точности модели.

Применение модели «случайный лес» становится возможным при условии предварительного обучения модели на наборе данных, полученном на основе анализа сетевого трафика в защищаемой сети (аналоге с соответствующими характеристиками) и содержащем признаки классифицируемых компьютерных атак. При этом на этапе сбора и подготовки обучающей выборки необходимо избегать разбалансированности распределения нормальных и аномальных записей, что может стать причиной переобучения модели и/или резкого увеличения числа ложных срабатываний классификатора [3].

Реализация предлагаемых решений в системах реального (близкого к реальному) времени предполагает эффективную обработку и анализ высокоскоростных потоков данных в условиях признакового пространства большой мощности и возможна лишь при наличии высокопроизводительной программно-аппаратной платформы. Снижение требований к производительности возможно за счет применения «многоуровневых» классификаторов, объединяющих быстрые низкоэффективные модели на этапе предварительной обработки и эффективные вычислительно сложные модели на более высоких уровнях [6].

Указанные обстоятельства вместе с известными результатами исследований предметной области позволяют сделать вывод о возможности применения методов машинного обучения для поиска аномалий и обнаружения компьютерных атак.

В заключении необходимо отметить, что перспективным направлением дальнейших исследований является разработка алгоритмов обнаружения компьютерных атак, основанных на использовании независимых от физической структуры сети и настроек используемого оборудования признаков, а также использовании технологий глубокого обучения нейронных сетей (deep learning), которые демонстрируют более высокие результаты по сравнению с другими методами при решении широкого круга задач (распознавания речи, классификации изображений, автоматического перевода и др.) [14].

Список литературы / References

[1]. Lee K.-F. AI Superpowers: China, Silicon Valley, and the New World Order. Houghton Mifflin Harcourt, 2018, 272 p.

[2]. Talabis M, McPherson R., Miyamoto I., Martin J. Information Security Analytics. Elsevier, 2015, 166 p.

[3]. Sumeet D., Xian D. Data Mining and Machine Learning in Cybersecurity. Auerbach Publications, 2011, 223 p.

[4]. Шелухин О.И., Ванюшина А.В., Габисова М.Е. Фильтрация нежелательных приложений интернет-трафика с использованием алгоритма классификации Random Forest. Вопросы кибербезопасности, № 2 (26), 2018 г., стр. 44-51. / Sheluhin O., Vanyushina A., Gabisova M. The Filtering of Unwanted Applications in Internet Traffic Using Random Forest Classification Algorithm. Voprosy kiberbezopasnosti, № 2 (26), 2018, pp. 44-51 (in Russian).

[5]. Kanimozhi V., Jacob T.P. Artificial Intelligence based Network Intrusion Detection with hyper-parameter optimization tuning on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing. ICT Express, vol. 5, issue 3, 2019, pp. 211-214.

[6]. Kostas K. Anomaly Detection in Networks Using Machine Learning. Master thesis. School of Computer Science and Electronic Engineering, University of Essex, 2018, 70 p.

[7]. Scikit-learn documentation. Random forest classifier. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier, accessed 16.08.2020.

[8]. Intrusion Detection Evaluation Dataset (CICIDS2017). Available at: https://www.unb.ca/cic/datasets/ids-2017.html, accessed 16.08.2020.

[9]. Panigrahi R., Borah S. A detailed analysis of CICIDS2017 dataset for designing Intrusion Detection Systems. International Journal of Engineering & Technology, vol 7, no 3.24, 2018, pp. 479-482..

[10]. Sharafaldin I., Lashkari A.H., Ghorbani Ali A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proc. of the 4th International Conference on Information Systems Security and Privacy (ICISSP), 2018, pp. 108-116.
[11]. Leskovec J., Rajaraman A., Ullman J. Mining Of Massive Datasets. Cambridge University Press, 2014. 476 p.
[12]. Domingos P. A Few Useful Things to Know about Machine Learning. Communications of the ACM, vol. 55, № 10, 2012. pp. 78-87.
[13]. Lashkari H. Characterization of Tor Traffic Using Time Based Features. In Proc. of the 3rd International Conference on Information System Security and Privacy, 2017, pp. 253-262.
[14]. McAfee A., Brynjolfsson E. Machine, Platform, Crowd. W.W. Norton & Company, 2017. 416 p.

Информация об авторах / Information about authors

Максим Николаевич ГОРЮНОВ – кандидат технических наук. Сфера научных интересов: информационная безопасность, системы обнаружения вторжений, системы анализа защищенности, машинное обучение.

Maxim Nikolaevich GORYUNOV – Ph.D. Research interests: information security, intrusion detection systems, security analysis systems, machine learning.

Андрей Георгиевич МАЦКЕВИЧ – кандидат технических наук, доцент. Сфера научных интересов: информационная безопасность, системы обнаружения вторжений, системы антивирусной защиты, машинное обучение, криптографические методы защиты информации.

Andrey Georgievich MATSKEVICH – Ph.D., associated professor. Research interests: information security, intrusion detection systems, anti-virus protection systems, machine learning, cryptographic methods for protecting information.

Дмитрий Александрович РЫБОЛОВЛЕВ – кандидат технических наук. Сфера научных интересов: информационная безопасность, системы обнаружения вторжений, машинное обучение, криптографические методы защиты информации.

Dmitry Aleksandrovich RYBOLOVLEV – Ph.D. Research interests: information security, intrusion detection systems, machine learning, cryptographic methods for protecting information.