11. Селиверстов Я. А. Моделирование процессов распределения и развития транспортных потоков в мегаполисах // Изв. СПбГЭТУ «ЛЭТИ». 2013. № 1. С. 43–49.

12. Селиверстов С. А. Методы и алгоритмы интеллектуального анализа процесса организации транспортной системы // Вестн. ГУМРФ им. адмирала С. О. Макарова. 2013. Вып. 2. С. 92–99.

13. Селиверстов Я. А. Использование правила резолюций в вопросно-ответной процедуре транспортного планировщика // Вестн. ГУМРФ им. адмирала С. О. Макарова. 2013. Вып. 1 (20). С. 145–152.

14. Селиверстов Я. А., Селиверстов С. А. О логико-алгебраическом представлении транспортно-логистического процесса // Науч.-техн. ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2014. Т. 4, № 200. С. 57–68.

15. Селиверстов Я. А. О построении модели классификации межагентных отношений социально-экономического поведения городского населения в системах управления транспортными потоками мегаполиса // Интернет-журн. «Науковедение». 2014. № 5 (24). М.: Науковедение, 2014. URL: http://naukovedenie.ru/PDF/159TVN514.pdf.

16. Кокаев О. Г., Лукомская О. Ю., Селиверстов С. А. О технологии анализа транспортных процессов в современных условиях хозяйствования // Транспорт Российской Федерации. 2012. № 2 (39). С. 32–36.

17. Селиверстов С. А., Селиверстов Я. А. Основы теории бесконфликтного непрерывного транспортного процесса движения // Интернет-журн. «Науковедение». 2014. № 3 (22). М.: Науковедение, 2014. URL: http://naukovedenie.ru/PDF/74TVN314.pdf.

Ya. A. Seliverstov, S. A. Seliverstov
The Institute of transport problems to them. N. With. Solomenco of the Russian academy of sciences

A. L. Starichenkov
Saint-Petersburg state electrotechnical university «LETI»

FEATURES OF CONSTRUCTION OF URBAN TRANSPORTATION AND LOGISTICS MONITORING

Detailed review of the existing transport technologies for monitoring and technical systems for collecting traffic information were given in the article, statement of the problem of creating a system for monitoring vehicle and urban residents were made here. Subsystems: identification and authentication, positioning of vehicle and urban residents were considered here. The construction of the structural model of system for monitoring vehicle and urban residents were carried out here and were marked its prospects.

City monitoring, technology identification, authentication, positioning, recognition of states of control objects

УДК 004.891.3

М. А. Протасова
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина)

Нейросетевой классификатор аномалий телекоммуникационной сети

Рассмотрено применение искусственных нейронных сетей к решению задачи распознавания и классификации аномалий в телекоммуникационной сети. Приведена реализация нейросетевого классификатора с использованием языка программирования R. Выбрано признаковое пространство и сформирована обучающая выборка нейронной сети. Представлена реализация классификатора аномалий телекоммуникационной сети.

Обучающая выборка, телекоммуникационная сеть, нейросетевой классификатор

Направление информационной безопасности телекоммуникационных сетей, связанное с обнаружением и последующим реагированием на нарушения, появилось и начало активно развиваться в 80-е гг. XX в.

В связи с несовершенством существующих методов защиты компьютерных систем от сетевых атак разработка новых методов защиты информации, позволяющих повысить уровень защищенности компьютерных систем от несанкционированного воздействия, является актуальной и востребованной. В качестве примеров исследования предметной области можно привести работы [1]–[5].

Для обнаружения и классификации аномалий используется, как правило, один из трех основных подходов либо их комбинация [4]:

– статистический анализ;

– экспертные системы;

– нейронные сети.

Статистический анализ применяется при обнаружении аномального поведения. Отклонение от среднего значения (т. е. дисперсия) профиля нормального поведения дает сигнал администратору о том, что зафиксирована аномалия. Средние частоты и значения переменных вычисляются для каждого типа нормального поведения (например, количество входов в систему, количество отказов в доступе, время суток и т. д.). О возможных атаках сообщается, когда наблюдаемые значения выпадают из нормального диапазона, т. е. превышают заданный порог [3]–[5].

Экспертная система – это система, которая в контексте обнаружения атак принимает решение о принадлежности того или иного события к классу атак на основании имеющихся правил. Правила основаны на опыте специалистов и хранятся в специальном хранилище. Правила экспертной системы опираются на так называемые сигнатуры, которые и ищутся в признаковом пространстве [4].

Нейронная сеть способна анализировать сетевой трафик с целью обнаружения аномального поведения при установлении соединения узлов. Вероятность правильного определения аномалии, ее типа и класса зависит от качества проведенного ранее обучения и обобщающей способности нейронной сети [6].

Методик построения нейронных сетей достаточное количество, одни используются чаще, другие реже. Тем не менее, можно с уверенностью утверждать, что не существует четкого алгоритма, используя который можно выбрать архитектуру сети, оптимально подходящую для решения конкретной задачи. В связи с этим исследователи, как правило, пользуются наиболее доступными с их точки зрения методами и инструментами моделирования нейронных сетей.

В данной статье описываются возможности инструментальных средств языка программирования R при решении задач нейросетевой классификации аномалий телекоммуникационных сетей.

Данные для обучения. В качестве базы данных для обучения и проверки классификатора была использована база KDD-99 [7]. База данных сформирована на основе дампов трафика в реальной сети, в которой тестировались несколько систем обнаружения атак в 1999 г. [7]. Эта база содержит 5 млн упорядоченных записей с 41 атрибутом.

Каждая запись промаркирована классом, к которому она относится: это либо нормальное соединение (normal), либо одна из 22 аномалий, которые в свою очередь можно разделить еще на 4 типа (DoS, U2R, R2L и Probe).

DoS-атаки – это сетевые атаки, направленные на создание ситуаций, при которых в рамках атакуемой системы происходит отказ в обслуживании. Данные атаки характеризуются генерацией большого объема трафика, что приводит к перегрузке и блокированию сервера. Выделяют 6 DoS-атак: back, land, neptune, pod, smurf, teardrop.

U2R-атаки предполагают получение зарегистрированным пользователем привилегий локального сетевого администратора. Выделяют 4 типа U2R-атак: buffer_overflow, loadmodule, perl, rootkit.

R2L-атаки характеризуются получением доступа незарегистрированного пользователя к компьютеру со стороны удаленного компьютера. Выделяют 8 типов R2L-атак: ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, warezmaster.

Probe-атаки заключаются в сканировании сетевых портов с целью получения конфиденциальной информации. Выделяют 4 типа Probe-атак: ipsweep, nmap, portsweep, satan.

Таким образом, в контексте построения классификатора сети имеем 41 вход и 23 выхода (22 аномалии и нормальное состояние) нейронной сети.

Следует отметить, что в базе данных не все аномалии представлены достаточным для обучения количеством записей. Так, например, аномалия loadmodule представлена всего лишь девятью примерами, а spy – двумя. Ввиду исходной неравномерности распределения обучающих примеров можно упростить исходную задачу до распознавания нормального и аномального состояний сети. Таким образом, в исходную выборку включается 150 значений нормального состояния, по 80 значений часто повторяющихся аномалий и все значения недостаточно представленных аномалий.

Для задачи классификации угроз имеет смысл рассматривать только 10 состояний телекоммуникационной сети (9 аномалий – нормальное состояние), так как остальные 12 классификатор не сможет распознать достоверно ввиду малого объема примеров. Согласно источникам [4], [7] для обнаружения и классификации девяти часто встречающихся атак достаточно 29 параметров, характеризующих сетевые соединения (табл. 1).

В контексте сформулированной таким образом задачи имеется 29 входов и 10 выходов нейронной сети. Первоначальная выборка была должным образом скорректирована. Для обеспечения более равномерного обучения выбиралось по 100 примеров каждого рассматриваемого состояния. В данном примере нормальное состояние сети никак не выделялось.

Перед использованием выборок для обучения сетей все значения параметров были подвергнуты нормировке (табл. 1).

Обучение сети. Одним из важнейших свойств нейронной сети является ее способность к обучению на основе поступающих данных. Количество ошибок сокращается за счет определенных правил с течением времени. Учитывая структуру базы данных KDD CUP99, был выбран алгоритм обучения нейронной сети с учителем.

В статье описано построение нейросетевого классификатора на основе двухслойной нейронной сети прямого распространения.

Обучающая выборка при полном наборе аномалий составлена из 1128 строк с 41 параметром, а при ограничении до девяти наиболее представленных – из 1000 строк с 29 параметрами. Выходы сети (целевое множество) кодируются унарным кодом. Следует отметить и то, что все манипуляции, производимые с обучающей выборкой, осуществлялись с использованием функционала R programming language.

Для работы с нейронной сетью на официальном сайте R-сообщества CRAN существует 2 пакета – «neuralnet» и «nnet» [8]. Оба имеют как плюсы, так и минусы, хотя по функциональности практически идентичны. Решающим фактором в выборе пакета стало то, что пакет «nnet» поддерживает стандарт

Таблица 1

№	Название параметра	Пояснение
1	duration	Продолжительность соединения
2	protocol_type	Тип протокола
3	service	Служба
4	flag	Флаг терминального состояния IP-соединения
5	src_bytes	Количество байт, переданных от источника к приемнику
6	dst_bytes	Количество байт, переданных от приемника к источнику
7	land	Равенство порта отправителя порту получателя
8	wrong_fragment	Количество отброшенных пакетов
9	urgent	Число пакетов с флагом URG
10	hot	Количество hot-индикаторов
11	count	Количество соединений между удаленным и локальным хостами
12	srv_count	Количество соединений к локальной службе
13	serror_rate	Процентное число соединений с ошибкой типа syn для данного хоста-источника
14	srv_serror_rate	Процентное число соединений с ошибкой типа SYN для данной службы источника
15	rerror_rate	Процентное число соединений с ошибкой типа REJ для данного хоста-источника
16	srv_rerror_rate	Процентное число соединений с ошибкой типа REJ для данной службы источника
17	same_srv_rate	Процентное число соединений к службе
18	diff_srv_rate	Процентное число соединений к различным службам
19	srv_diff_host_rate	Процентное число соединений к различным хостам
20	dst_host_count	Количество соединений к локальному хосту, установленных удаленной стороной
21	dst_host_srv_count	Количество соединений к локальному хосту, установленных удаленной стороной и использующих одну и ту же службу
22	dst_host_same_srv_rate	Процентное число соединений к локальному хосту, установленных удаленной стороной и использующих одну и ту же службу
23	dst_host_diff_srv_rate	Процентное число соединений к локальному хосту, установленных удаленной стороной и использующих различные службы
24	dst_host_same_src_port_rate	Процентное число соединений к данному хосту при текущем номере порта источника
25	dst_host_srv_diff_host_rate	Процентное число соединений к службе разных хостов
26	dst_host_serror_rate	Процентное число соединений с ошибкой типа syn для данного хоста-приемника
27	dst_host_srv_serror_rate	Процентное число соединений с ошибкой типа SYN для данной службы приемника
28	dst_host_rerror_rate	Процентное число соединений с ошибкой типа REJ для данного хоста-приемника
29	dst_host_srv_rerror_rate	Процентное число соединений с ошибкой типа REJ для данной службы приемника

Таблица 2

Параметр	Название параметра	Пояснение
x	Вектор/матрица	Матрица или набор значений для обучения (исходные данные)
y	Вектор/матрица	Матрица или набор целевых данных для обучения
size	Целое положительное число или 0	Количество нейронов в скрытом слое. Может быть 0, если есть обходные соединения
Wts	Вектор/матрица	Начальный вектор весов сети. Если отсутствует, генерируется случайным образом
linout	TRUE/FALSE	Переключение на линейное отображение выходных данных. По умолчанию установлена логическая система единиц
skip	TRUE/FALSE	Переключатель, добавляющий обходное соединение от входа к выходу
decay	Малое число	Параметр распада веса. По умолчанию – 0
maxit	100	Максимальное число итераций (эпох)
trace	TRUE/FALSE	Переключение для отслеживания оптимизации. Значение по умолчанию – true
abstol	1.0e-4	Если критерий ошибки становится меньше abstol, то задача обучения выполнена
reltol	1.0e-8	Остановка, если оптимизатор не может уменьшить нужный критерий хотя бы на одну величину reltol

PMML, что немаловажно при необходимости использования полученной нейронной сети в других статистических системах [9].

Пакет «nnet» содержит несколько функций, позволяющих работать как с нейронной сетью, так и с лог-линейными моделями. Для обучения сети предназначена функция «nnet», а для использования в режиме классификации – «predict».

Параметры функции «nnet», использованные при проведении эксперимента, приведены в табл. 2.

По умолчанию в функции используются следующие параметры:

```
nnet(x, y, size, Wts, linout = FALSE,
    skip = FALSE, decay = 0,
    maxit = 100, trace = TRUE,
    abstol = 1.0e-4, reltol = 1.0e-8, ...).
```

При построении нейронной сети используется сигмоидальная функция активации нейронов и метод наименьших квадратов для минимизации ошибки в процессе обучения. Так как единых правил для определения структуры нейронной сети нет, для построения классификатора использовался алгоритм подбора критериев. Основными задачами подбора являлись:

1. Подобрать параметры сети так, чтобы ответ классификатора был максимально приближен к действительности. Для выполнения этой задачи был создан зацикленный алгоритм подбора параметров.

2. Минимизировать вероятность переобучения. В алгоритм подбора параметров была добавлена проверка на переобучение с помощью тестовой выборки.

Алгоритм подбора:

Шаг 1. Выбираются параметры сети.

Шаг 2. Обучающая выборка делится случайным образом на 2 части так, чтобы в обучающей и тестовой выборках было примерно равное количество записей каждого состояния сети. Создаются эталоны выходов нейронной сети для обеих выборок.

Шаг 3. С использованием функции 'nnet', выбранных параметров и обучающей выборки производится обучение.

Шаг 4. Обученная сеть проверяется с использованием тестовой выборки и функции 'predict' (на тестовой выборке используется обученный классификатор, а затем результат сравнивается с эталоном).

Таким образом не только подбираются параметры нейронной сети, но и исключается возможность ее переобучения, так как при дальнейшем отборе учитывается не только конечное значение ошибки при обучении, но и процент несоответствий при проверке на тестовой выборке.

В результате были получены следующие параметры нейронной сети, используемой для классификации всех видов аномалий: 11 нейронов в скрытом слое, коэффициент распада 0.0015, количество эпох 240:

```
nnet(x, y, 11, linout = FALSE,
    skip = FALSE, decay = 0,0015,
    maxit = 240, trace = TRUE,
    abstol = 1.0e-4, reltol = 1.0e-8);
```

для классификации девяти видов аномалий: 8 нейронов в скрытом слое, коэффициент распада 0 и количество эпох 180:

```
nnet(x, y, 8, linout = FALSE,
    skip = FALSE, decay = 0,
    maxit = 180, trace = TRUE,
    abstol = 1.0e-4, reltol = 1.0e-8).
```

Таблица 3

№	Имя выхода НС	Полученный результат	Эталон	Погрешность, %
1	normal	76	75	1
2	back	38	40	5
3	buffer_overflow	12	15	20
4	ftp_write	2	4	50
5	guess_passwd	26	27	4
6	imap	3	6	50
7	ipsweep	40	40	0
8	land	9	10	10
9	loadmodule	0	5	100
10	multihop	0	4	100
11	neptune	39	40	3
12	nmap	36	40	10
13	perl	0	2	100
14	phf	2	2	0
15	pod	39	40	3
16	portsweep	40	40	0
17	rootkit	0	5	100
18	satan	34	40	15
19	smurf	40	40	0
20	spy	0	1	100
21	teardrop	38	40	5
22	warezclient	40	40	0
23	warezmaster	10	10	0
Итоговая погрешность				29

Результаты. В качестве результатов представлены данные, полученные при тестировании нейронной сети с наилучшими параметрами.

В табл. 3 приведены результаты обучения нейронной сети при вхождении в тестовую выборку всех аномалий. Как и предполагалось, суммарная погрешность определения аномалий велика, в то время как погрешность определения нормального состояния мала.

Для следующего эксперимента выборка была осуществлена только среди аномалий, представленных в достаточном количестве и нормального состояния.

В табл. 4 показаны результаты обучения нейронной сети без включения в выборку аномалий, представленных менее 100 раз в базе данных. По горизонтали представлены полученные результаты, а по вертикали – ожидаемые. Таким образом, в случае идеального обучения получилась бы диагональная матрица.

Средняя погрешность определения аномалий в данном примере на порядок меньше, чем в предыдущем, из чего следует вывод о том, что данные, представленные в недостаточном количестве, существенно искажают результат классификации и усложняют работу алгоритма.

Рассмотренный подход к реализации нейросетевого классификатора отличается простотой и основан на построении НС с одним скрытым слоем и сигмоидальной функцией активации. Из полученных результатов следует, что R programming language – удобная среда для разработки и использования нейронных сетей. Изучены результаты обучения и подтвержден факт недостаточности некоторых данных в базе KDD CUP99 для полноценного обучения сети.

Таблица 4

№	Имя	1	2	3	4	5	6	7	8	9	10	Эталон	Погрешность, %
1	normal	50	0	0	0	0	0	0	0	0	0	50	0
2	back	1	49	0	0	0	0	0	0	0	0	50	2
3	neptune	0	0	49	0	0	0	1	0	0	0	50	2
4	pod	0	0	0	50	0	0	0	0	0	0	50	0
5	smurf	0	1	0	1	46	0	0	2	0	0	50	8
6	teardrop	0	0	0	0	0	48	2	0	0	0	50	4
7	ipsweep	3	1	0	0	0	0	46	0	0	0	50	8
8	nmap	0	0	0	0	1	0	0	49	0	0	50	2
9	portsweep	0	0	0	1	0	0	0	0	47	2	50	6
10	satan	0	0	0	0	0	0	0	0	1	49	50	2
Итоговая погрешность													3.4

На следующем шаге исследований планируется организовать комитет нейронных сетей, объединенных для принятия решений.

СПИСОК ЛИТЕРАТУРЫ

1. Системы обнаружения компьютерных угроз. URL: http://www.nestor.minsk.by/sr/2008/05/sr80513.html.

2. Гамаюнов Д. Ю. Обнаружение компьютерных атак на основе анализа поведения сетевых объектов: дис. ... канд. физ.-мат. наук / Моск. гос. ун-т им. М. В. Ломоносова. М., 2007.

3. Лукацкий А. В. Обнаружение атак. 2-е изд., перераб. и доп. СПб.: БХВ-Петербург, 2003.

4. Тимофеев А. В., Браницкий А. А. Исследование и моделирование нейросетевого метода обнаружения и классификации сетевых атак // Information Technologies & Knowledge. 2012. Vol. 6, № 3. С. 257–265.

5. Технологии обнаружения сетевых атак. URL: http://www.bstu.by/~opo/ru/uni/bstu/science/ids/.

6. Дорогов А. Ю., Абатуров В. С. Экспериментальная оценка обобщающей способности нейронных сетей// Нейроинформатика-2013: сб. науч. тр. Ч. 2 / МИФИ. М., 2013. С. 244–251.

7. KDD Cup 1999. Data. URL: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

8. The Comprehensive R Archive Network. URL: http://cran.r-project.org/.

9. Дорогов А. Ю., Абатуров В. С., Раков И. В. PMML модели быстрых нейронных сетей и спектральных преобразований // Нейроинформатика-2013: сб. науч. тр. Ч. 3 / МИФИ. М., 2013. С. 156–166.

M. A. Protasova
Saint-Petersburg state electrotechnical university «LETI»

NEURAL NETWORK QUALIFIER ANOMALY TELECOMMUNICATION

The application of artificial neural networks to solve the problem of recognition and classification of anomalies in the telecommunication-network is considered. The implementation of the neural network classifier using the programming language R is considered. Selected feature space and formed the training sample of neural network. The implementation of the telecommunications network classifier anomalies is presented.

Training sample, a telecommunications network, the neural network classifier

УДК 621.391

О. Г. Бородина, В. Ю. Цветков
Белорусский государственный университет информатики и радиоэлектроники

Выделение изолированных прямых линий на изображениях с использованием форм-фактора

Предложен метод выделения изолированных прямых контурных линий на основе анализа их длин и размеров. Сущность метода состоит в сопоставлении с единицей значения форм-фактора – отношения размера контурной линии (расстояния между концевыми точками) к ее длине (числу образующих контурных пикселей). Метод позволяет повысить скорость выделения прямых контурных линий по сравнению с методами LSD и на основе преобразования Хафа.

Выделение прямых линий, контурная обработка изображений

Методы обработки перекрывающихся изображений (наложение, сравнение, построение панорам и т. д.) основаны на локализации реперных элементов. Наиболее эффективные методы локализации реперов, такие, как SIFT [1] и SURF [2], используют характерные точки. Это позволяет снизить вычислительную сложность сопоставления изображений по сравнению с корреляционными методами [3], однако приводит к низкой устойчивости результатов локализации реперов, к изменению яркости, контраста и зашумлению изображения. Прямые линии обладают значительно большей устойчивостью по сравнению с точками. Для их выделения на изображениях широко используются методы, основанные на преобразовании Хафа [4], масочном поиске [5], вычислении градиента [6] и квантовании по ориентации [7]. Однако вычислительная сложность этих методов высока. Устранить данный недостаток можно за счет учета соотношения длин контурных линий и расстояний между их концевыми точками.

Метод выделения изолированных прямых контурных линий на основе форм-фактора. Предлагаемый метод основан на анализе значения форм-фактора – отношения размера контур-