ISSN 2782-2001
Системы анализа
и обработки данных
том 92, № 4, 2023, с. 69–84

http://journals.nstu.ru/vestnik
Analysis and data
processing systems
Vol. 92, No. 4, 2023, pp. 69–84

ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ
И ТЕЛЕКОММУНИКАЦИИ

INFORMATION
TECHNOLOGIES
AND TELECOMMUNICATIONS

УДК 519.873

DOI: 10.17212/2782-2001-2023-4-69-84

ИССЛЕДОВАНИЕ ПРОБЛЕМАТИКИ МЕТОДИК ОПРЕДЕЛЕНИЯ ТИПА КОНТЕНТА ВО ВХОДЯЩЕМ ТРАФИКЕ*

И.Л. РЕВА[a], М.А. МЕДВЕДЕВ[b], В.И. ВОРОНЦОВА[c]

630073, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет

[a] *reva@corp.nstu.ru* [b] *m.medvedev@corp.nstu.ru* [c] *i.voronczova@corp.nstu.ru*

Фильтрация контента в контексте кибербезопасности и доверенной среды является важной тактикой, используемой для обеспечения безопасности и функциональности сети. Он действует путем ограничения доступа к определенным веб-сайтам, электронным письмам, файлам или другому контенту, который может содержать вредоносные элементы или представлять существенный риск заражения. Фильтрация контента обеспечивает безопасность не только данных отдельного пользователя, но и всей сети организаций, учреждений, помогая минимизировать риск злонамеренных нарушений безопасности.

Исследование проблематики методик определения типа контента во входящем трафике является актуальным и важным направлением в области информационной безопасности и сетевой аналитики. В современном интернет-пространстве значительное количество данных передается через сети, и одной из ключевых задач является классификация этого трафика для обеспечения безопасности и эффективного управления сетью. Методики определения типа контента во входящем трафике представляют собой набор алгоритмов и подходов, которые позволяют автоматически определить, какой тип данных передается по сети. В ходе исследования проблематики методик определения типа контента во входящем трафике осуществляется сбор данных о сетевом трафике, выбирается набор данных для обучения модели, рассматриваются алгоритмы-классификаторы и акцентируется внимание на метриках оценки эффективности классификации.

Результаты исследования могут быть использованы для создания эффективных систем обнаружения вредоносного или нежелательного контента, фильтрации данных или оптимизации работы сетевых ресурсов. Исследование проблематики методик определения типа контента во входящем трафике имеет практическую значимость и может применяться в различных областях, включая информационную безопасность, сетевую аналитику, мониторинг сетевых ресурсов и оптимизацию сетевых процессов.

Ключевые слова: искусственный интеллект, компьютерные сети, семантическая контент-фильтрация, сетевой трафик, датасет, кибербезопасность, true-трафик, false-трафик, true positive, true negative, false positive, false negative, точность, полнота, *F*-мера

* *Статья получена 23 июня 2023 г.*

ВВЕДЕНИЕ

С развитием Интернета сетевая безопасность привлекла внимание людей. Можно сказать, что безопасная сетевая среда является основой быстрого и надежного развития Интернета.

Кибербезопасность стала первостепенным фактором в цифровом мире, где каждый день по различным сетям передается огромный объем информации. Среди всего этого контента неизбежно, что какая-то информация будет содержать вредоносные элементы, такие как вредоносное ПО, или элементы информации могут быть просто неуместными и неподходящими. В этом и заключается значение фильтрации контента. Это помогает смягчить угрозы и отсрочить пагубные последствия, вызванные небезопасным или неподходящим контентом.

Всё большую актуальность набирает разработка новых современных систем семантической контентной фильтрации сетевого трафика в связи с увеличением количества источников нежелательного контента.

Можно сказать, что фильтрация контента является краеугольным камнем комплексной стратегии кибербезопасности в борьбе с множеством онлайн-угроз, особенно со случаями вторжений. Он обеспечивает защиту от нежелательного спама или неприемлемого контента, тем самым защищая как отдельных пользователей, так и более крупные сети. Прежде чем начнем рассматривать требования к составу и содержанию методики определения типа контента во входящем трафике, дадим несколько общих определений нескольким ключевым понятиям.

Интернет-трафик – объем данных, которыми обмениваются между собой компьютеры и прочие устройства, подключенные к сети Интернет [33].

Трафик в Интернете может быть разделен на два типа: входящий и исходящий. Входящий трафик представляет собой данные, принимаемые устройством из Интернета, а исходящий трафик – данные, передаваемые устройством в сеть.

Фильтрация трафика – ограничение доступа пользователей к нежелательным сайтам [1]. Наряду с полезной информации в Интернете присутствует и вредоносная. Программное обеспечение, которое анализирует трафик, вычисляет и блокирует опасную информацию, позволяя избежать заражения вирусом или утечки данных.

1. СБОР ДАННЫХ О СЕТЕВОМ ТРАФИКЕ

Сетевым трафиком называют информацию, проходящую через компьютерную сеть. В нем фиксируется весь объем данных, получаемый в результате перехода с ресурса на ресурс.

Система контентной фильтрации трафика основана на определенных наборах записей (URL-адресов, IP-адресов, портов и т. п.). При поступлении пакета из внешней сети система фильтрации перехватывает этот пакет, проверяет его содержимое на совпадение с одной из записей набора правил и при обнаружении совпадения либо пропускает пакет в локальную сеть, либо отбрасывает его [2].

Методология сбора трафика подразделяется на сбор реального и имитируемого трафика [3].

Данные о реальном трафике собираются из операционной среды с помощью программного обеспечения для сбора трафика Wireshark. После сбора сведений данные анализируются и используются в рамках задач классификации.

Для генерации имитируемого трафика обычно используются целевые наборы данных трафика, построенные посредством модели сети в соответствии с заданными параметрами запроса. Наборы данных, сгенерированные с помощью этого метода, в большинстве случаев содержат больше необходимых характеристик, при которых должен сработать алгоритм-классификатор, поскольку изначально данные собираются с целью проверки работоспособности алгоритма.

Сгенерированные данные больше подходят для машинного обучения. Они более сбалансированные, чем наборы данных из трафика, собранного в процессе работы обычного пользователя. Однако сгенерированные наборы данных могут отличаться от реальных сетевых данных и неверно отражать реальные условия сетевого трафика, что повлечет за собой достаточное количество ложных срабатываний.

Для корректного обучения модели, предназначенной для распознавания true- и false-трафика, должны быть соблюдены следующие требования [4]:

– датасет должен обладать набором URL-адресов, которые однозначно можно классифицировать как true- или false-трафик;

– в датасете должна присутствовать информация о том, какой тип контента запрашивался на каждом URL-адресе (текст, изображения, аудио или видео и т. д.);

– набор данных должен содержать достаточное количество записей для каждой категории, чтобы модель могла научиться с высокой точностью распознавать тип трафика.

В случае присутствия в обучающей выборке данных true-трафика с признаками false-трафика или же неопределенных данных, вид которых нельзя однозначно определить, обучение алгоритма будет происходить медленно и классификатор не сможет с высокой точностью различать тип контента.

Захват сетевого трафика позволяет проанализировать данные, передаваемые через сеть. Существует множество методов захвата трафика как с «точки зрения» программ, так и с «точки зрения» уровней модели OSI, на которых захват может осуществляться. Далее рассмотрим подробно существующие методы.

2. ВЫБОР НАБОРА ДАННЫХ ДЛЯ ОБУЧЕНИЯ МОДЕЛИ

Требования к классификации датасета:

1) чистота выборки: трафик должен быть отфильтрован;
2) сессии, на которые трафик должен быть разделен;
3) разделение на пользователей;
4) разметка трафика по атрибутам: дата – адаптер – протокол – ip отравляющего – порт отравляющего – ip принимающего – порт принимающего – заголовки – тело пакета;

5) атрибуты трафика должны быть сохранены в формате CSV;
6) количество пакетов за сессию;
7) размер переданных/полученных данных за сессию.

При формировании набора данных, на которых будет обучаться конечная модель, важно помнить, что большее количество true- или false-трафика не даст лучших результатов. Наличие слишком большого количества примеров в наборе датасетов требует большого объема памяти и вычислений, увеличивает время обучения и влияет на точность модели.

3. АЛГОРИТМЫ-КЛАССИФИКАТОРЫ

Рассматриваемый сетевой поток для итогового алгоритма может быть разделен на два класса – класс приемлемого контента и класс неприемлемого контента. Для машинного обучения разделение на два класса данных является задачей классификации.

Предполагаем, что множество пар «объект, класс» $X \times Y$ является вероятностным пространством с неизвестной вероятностной мерой P. Для алгоритма имеется конечная обучающая выборка наблюдений $X^m = \{(x_1, y_1), \ldots, (x_m, y_m)\}$, сгенерированная согласно вероятностной мере P.

Требуется построить алгоритм a: $X \to Y$, способный классифицировать произвольный объект $x \in X$ [5].

Алгоритмы машинного обучения, применяемые для решения задачи классификации, можно условно разделить на следующие группы: линейные классификаторы, метод опорных векторов, метрические классификаторы, алгоритмы на основе деревьев решений, ансамблевые алгоритмы, искусственные нейронные сети [6, 7].

Рассмотрим подробнее каждый алгоритм.

1. Линейными классификаторами называют алгоритмы машинного обучения, которые используются для решения задач классификации [7]. Они основаны на построении линейной функции (разделяющей гиперплоскости) для разделения объектов на классы. Данный алгоритм работает быстрее, чем большинство других алгоритмов, и с достаточно высокой точностью справляется с задачей классификации на небольшом наборе данных. Несмотря на достоинства, линейные классификаторы могут быть менее эффективны в задачах с нелинейными зависимостями между признаками и целевым классом. К наиболее популярным линейным классификаторам относятся логистическая регрессия, метод опорных векторов и Latent Diriehlet Allocation (LDA) [7].

2. Метод опорных векторов – это алгоритм машинного обучения, используемый для решения задач классификации и регрессии [7]. Алгоритм основан на геометрических методах для разделения данных на классы: он строит гиперплоскость максимального расстояния между классами объектов путем выбора опорных векторов – точек данных, которые находятся на границе между классами. Данный алгоритм хорошо работает для задач обработки текстов, классификации изображений и др. Помимо этого, он может использоваться для решения задач регрессии. Однако для метода опорных векторов требуется

большой набор данных для преобразования признаков, что может быть трудоемко [7].

3. Метрические классификаторы – это алгоритмы машинного обучения, которые классифицируют объекты на основе расстояния между ними [7]. Метод основан на поиске ближайших соседей для классификации данных. Метрические классификаторы эффективны в решении задач классификации с небольшим количеством признаков. Несмотря на достоинства, алгоритм обладает недостаточной точностью в случаях, когда признаков крайне много или они коррелируют друг с другом. Примерами метрических классификаторов являются k ближайших соседей и взвешенное голосование. K-Nearest Neighbors (KNN) является часто используемым алгоритмом метрической классификации [7].

4. Алгоритмы на основе деревьев решений – это алгоритмы машинного обучения, которые используют дерево решений для классификации объектов [7]. Алгоритм основан на построении бинарного дерева, в котором каждый узел представляет собой признак, а листья – классы объектов. Алгоритмы на основе деревьев решений легко интерпретируются и могут использоваться для определения важности признаков. Также для алгоритма на основе дерева не требуется нормализация данных. Из недостатков деревьев стоит отметить, что скорость работы может быть довольно низкой, и алгоритм склонен к переобучению при большом объеме данных. Примерами алгоритмов на основе деревьев решений являются CART, ID3 и C4.5 [8].

5. Искусственные нейронные сети (ИИ) – это алгоритмы машинного обучения, которые имитируют работу мозга, состоящую из нейронов, связей между ними и процесса обучения, и предназначены для работы с большими объемами данных [7]. ИИ основаны на математических моделях. Нейросети могут использоваться для решения задач классификации, регрессии и прогнозирования. Для улучшения точности работы нейросетей необходим достаточно большой объем данных. Однако для обучения искусственной нейронной сети может потребоваться длительное время, особенно на больших выборках [9].

Поскольку в настоящей работе рассматриваются характеристики, классифицирующие нежелательный контент в сети, на основе которых сформированы требования к методике автоматизированного определения типа контента, ниже будет рассмотрена теория о метриках оценки эффективности классификации.

3.1. МЕТОДЫ КОМПОЗИЦИИ ОБУЧАЮЩИХСЯ АЛГОРИТМОВ

При решении нетривиальных задач классификации возникает сложность с восстановлением желаемой зависимости, так как ее реализация слишком трудна либо на различных тестовых данных эффективность алгоритма нестабильна. Для решения вышеупомянутых проблем существуют методы композиции обучающихся алгоритмов, в которых ошибки могут быть взаимно скомпенсированы. Среди таких методов выделяют голосование, взвешенное голосование и смесь экспертов [10].

Пусть X – множество описаний объектов, Y – множество ответов, и существует некоторая зависимость – отображение f: $X \to Y$, значение которой известно только на объектах конечной обучающей выборки $(X, Y) = \{(x_1, y_1), \ldots, (X_{|(X,Y)|}, Y_{|(X,Y)|})\}$. Необходимо построить алгоритм a: $X \to Y$, аппроксимирующий целевую зависимость на всем множестве x.

Для построения алгоритмов вводится множество R – пространство оценок. Рассматриваются алгоритмы, имеющие вид суперпозиции $a(x) = C(b(x))$, где функция b: $X \to R$ называется алгоритмическим оператором, функция C: $X \to Y$ – решающим правилом [2].

Методы композиции обучающихся алгоритмов:

– простое голосование (Simple Voting)

$$b(x) = F(b_1(x), \ldots, b_t(x)) = \frac{1}{T}\sum_{t=1}^{T} b_t(x); \quad (1)$$

– взвешенное голосование (Weighted Voting):

$$b(x) = F(b_1(x), \ldots, b_T(x)) = \frac{1}{T}\sum_{t=1}^{T} w_t b_t(x), \quad (2)$$

$$\sum_{t=1}^{T} w_t = 1, w_t \geq 0; \quad (3)$$

– смесь экспертов (Mixture of Experts):

$$b(x) = F(b_1(x), \ldots, b_T(x)) = \sum_{t=1}^{T} w_t(x) b_t(x), \quad (4)$$

$$\sum_{t=1}^{T} w_t = 1, \ \forall x \in X_t. \quad (5)$$

Следует отметить, что каждый из приведенных методов является частным случаем, поэтому применение того или иного метода зависит исключительно от поставленной задачи.

Также стоит сказать, что обучение подобных алгоритмов осуществляется значительно дольше других из-за большого числа степеней свободы. Поэтому его применимость обоснована только в случае априорной информации о функциях компетенции.

3.2. ОШИБКИ ПЕРВОГО И ВТОРОГО РОДА

Все объекты выборки можно разделить на четыре непересекающихся множества (рис. 1) [2].

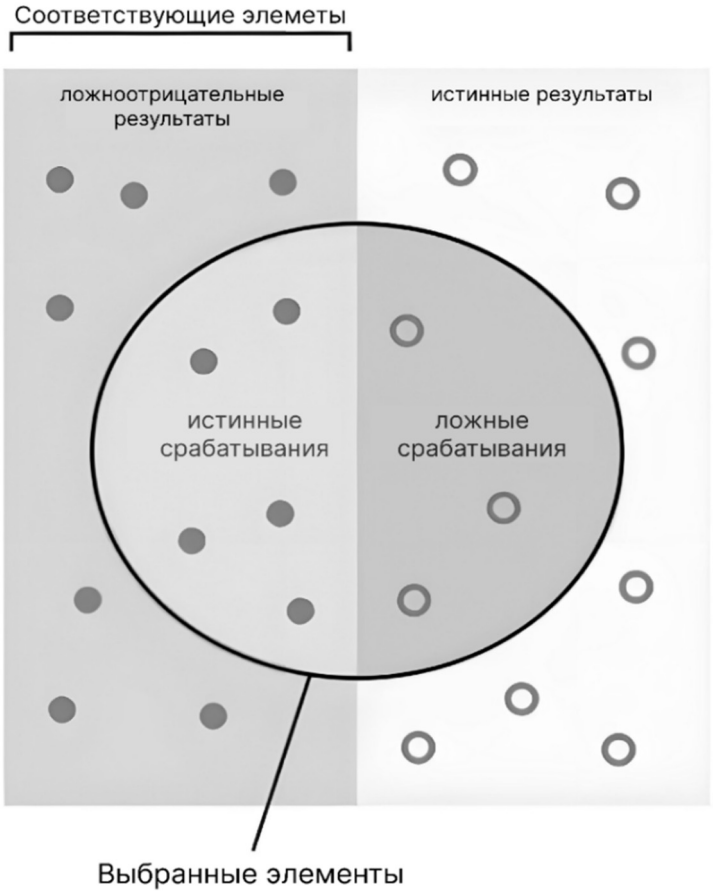

Рис. 1. Схема ошибок первого и второго рода

Fig. 1. Diagram of errors of the first and second types

Истинно положительные значения (true positive, *TP*) – это объекты, отнесенные моделью к положительному классу и действительно ему принадлежащие [11].

Истинно отрицательные значения (true negative, *TN*) – соответственно объекты, правильно распознанные моделью как принадлежащие отрицательному классу [11].

Ложноположительные значения (false positive, *FP*) – это те объекты, которые модель распознала как положительные, хотя на самом деле они отрицательные. В математической статистике такая ситуация называется ошибкой первого рода [12].

Ложноотрицательные значения (false negative, *FN*) – объекты, которые ошибочно отнесены моделью к отрицательному классу, хотя на самом деле они положительные. Такого рода ситуацию называют ошибкой второго рода [12].

Одной из основных проблем при использовании алгоритмов обнаружения нежелательного контента является наличие ошибок первого и второго рода в процессе классификации сетевого трафика. Эти ошибки негативно влияют на полноту и точность обнаружения, приводя к большому количеству ложных срабатываний или пропусков. Улучшение работы алгоритма сводится к уменьшению ошибок первого и второго рода, что позволит достичь более полного и точного обнаружения нежелательного контента [13].

4. МЕТРИКИ ОЦЕНКИ ЭФФЕКТИВНОСТИ КЛАССИФИКАЦИИ

Для оценки эффективности алгоритмов классификации используются такие метрики, как true positive (*TP*), true negative (*TN*), false positive (*FP*) и false negative (*FN*). На основе этих параметров вычисляются показатели точности, полноты и *F*-мера [2].

Основными метриками классификации являются достоверность (accuracy) классификации (6) и ошибка (error) (7) [2]. Также на практике используется параметр, противоположный параметру *TPR* – false positive rate (*FPR*), – частота ложноположительных результатов (8) [2]:

$$accuracy = TPR = \frac{TP + TN}{|S|} = \frac{TP + TN}{TP + TN + FP + FN}; \quad (6)$$

$$error = 1 - accuracy; \quad (7)$$

$$FPR = \frac{FP}{FP + TN}. \quad (8)$$

Для достижения более эффективного баланса между истинно положительными и ложноположительными результатами классификации применяют метрики точности (*precision*), полноты (*recall*) и *F*-меру (*F*-score) [2].

Точность (*precision*) показывает, какой процент объектов (9), для которых алгоритм предсказал класс 1, действительно относится к классу 1 [2]:

$$precision = \frac{TP}{TP + FP}. \quad (9)$$

Полнота (*recall*) показывает, для какого процента объектов класса 1 алгоритм предсказал класс 1 [2]:

$$recall = \frac{TP}{TP + FN}. \tag{10}$$

Обычно обе метрики вычисляются одновременно, так как они взаимно дополняют друг друга. Точность зависит от распределения данных, в отличие от полноты. Полнота не учитывает количество неверно помеченных положительных образцов, а точность не сообщает о количестве неправильно помеченных положительных образцов [14].

В большинстве случаев применяется комбинация полноты и точности в так называемой *F*-мере (11) [2], которая сочетает в себе упомянутые метрики и присваивает взвешенную важность либо точности, либо полноте, используя коэффициент β [2]:

$$F_{score} = \frac{(1+\beta^2) \cdot recall \cdot precision}{\beta^2 \cdot recall + precision}. \tag{11}$$

Однако чаще всего важность точности и полноты удобнее всего присваивать в равной степени. Поэтому вышеупомянутая формула принимает вид среднего гармонического этих двух величин [2]:

$$F_{score} = 2\,\frac{recall \cdot precision}{recall + precision}. \tag{12}$$

Формула (12) показывает, что значение *F*-меры напрямую зависит от значений обеих метрик. Следовательно, если хотя бы одна из двух метрик близка к нулю, *F*-мера тоже будет близка к нулю, что свидетельствует о низком качестве примененного алгоритма или отсутствии баланса входных обучающих данных [2].

Также существует метрика, с помощью которой можно оценить индуктивное смещение или среднегеометрическое классификатора, которая носит название *G-mean* [2]:

$$G_{mean} = \sqrt{\frac{TP \cdot TN}{(TP + FN)(TN + FP)}}. \tag{13}$$

Для более наглядного представления соотношения между *TP* и *FP* в выборке данных и настройке его баланса существует так называемая *ROC*-кривая (Receiver Operating Characteristic – рабочая характеристика приемника) (рис. 2) [2].

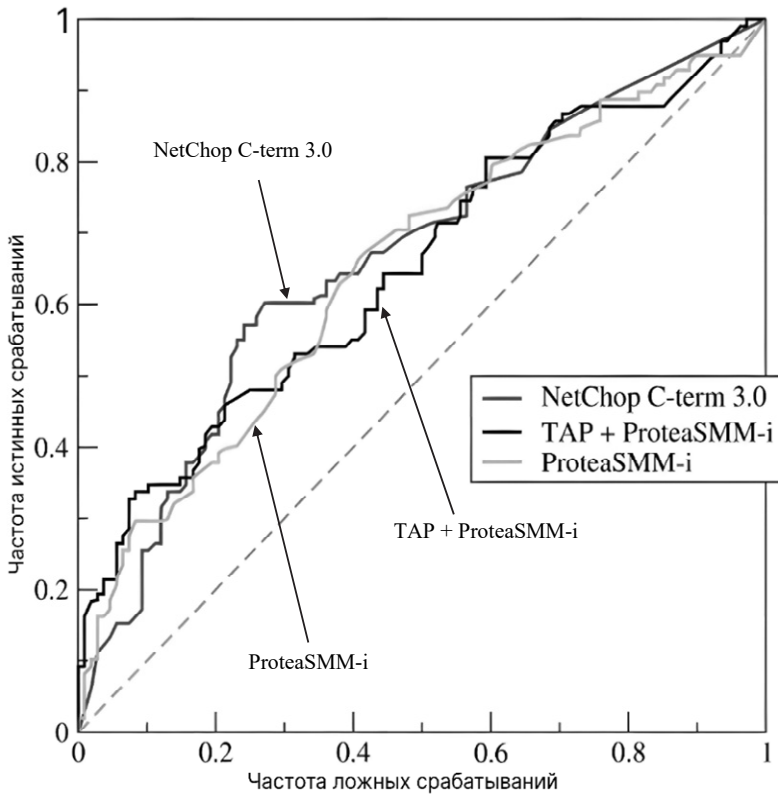

Рис. 2. *ROC*-кривая

Fig. 2. The *ROC*-curve

ROC-кривая представляет собой зависимость между частотой истинно положительных результатов и частотой ложноположительных [2].

ROC-кривая позволяет оценить классификатор при варьировании порога, решающего правило, однако не с числовой точки зрения. Для того чтобы численно сравнить два классификатора, вводится метрика *AUC* (Area Under Curve – площадь под кривой) [2]:

$$AUC = \int_{\infty}^{-\infty} y(T)\,x'(T)\,dT \int_{\infty}^{-\infty} TPR(T)\cdot FPR'(T)\,dT = \langle TPR \rangle, \qquad (14)$$

где $\langle . \rangle$ означает взятие среднего.

Чем больше *AUC*, тем лучше классификатор. В случае бинарной классификации ситуация, когда $AUC = 0{,}5$, также считается неприемлемой, поскольку означает то, что алгоритм не может определить признаки того или иного класса и выдает случайный результат [2].

R2_score – это статистический показатель, который отражает степень соответствия предсказанных результатов модели реальным [2]. Значение R2_score находится в диапазоне от нуля до единицы. При значении R2_score, равном единице, модель идеально соответствует данным и нет разницы между прогнозируемым значением и фактическим значением. Однако когда R2_score

равен нулю, модель не предсказывает результаты в модели и не изучает никаких взаимосвязей между зависимыми и независимыми переменными.

В случае отрицательного значения R2_score задача прогнозирования не выполняется, так как условная прямая линия предсказывает результаты лучше, чем обученная модель, а используемые признаки не объясняют целевую величину.

Метрика R2_score, также известная как коэффициент детерминации, может измерять изменчивость зависимой переменной *Y*, которая объясняется независимыми переменными X_i при тестировании модели на предмет предсказания класса переменной.

В первую очередь вычисляется среднее значение целевой / зависимой переменной y (обозначается $\underline{y}$), затем – общая сумма квадратов путем вычитания каждого наблюдения y_i из $\underline{y}$, возведения его в квадрат и суммирования по всем значениям [2]:

$$ss_{tot} = \sum_{i=1}^{n} (y_i - \underline{y})^2 . \tag{15}$$

Далее оценивается параметр модели с использованием подходящей регрессионной модели, например, с помощью линейной регрессии или SVM-регрессора.

На втором шаге рассчитывается сумма квадратов, полученная в результате регрессии, которая обозначается SSR и вычисляется путем вычитания каждого прогнозируемого значения *y*, обозначаемого $\hat{y}_i$, из y_i, возведения этих разностей в квадрат и последующего суммирования всех *n* слагаемых [2]:

$$SSR = \sum_{i=1}^{n} \left(\hat{y}_i - \underline{y} \right)^2 . \tag{16}$$

В последнюю очередь рассчитывается сумма квадратов SS_{res}, которая объясняет неучтенную изменчивость зависимого y после прогнозирования по независимой переменной в модели [2]:

$$ss_{res} = \sum_{i=1}^{n} (y_i - \underline{y}_i)^2 . \tag{17}$$

Таким образом, R2_score можно рассчитать с помощью формул (18) и (19) [2]:

$$R^2 = \frac{SSR}{SS_{tot}} ; \tag{18}$$

$$R^2 = 1 = \frac{SSR}{SS_{tot}} . \tag{19}$$

Хотя при помощи R2_score можно получить важные сведения о возможности прогнозирования модели с помощью этой статистической меры, не стоит полагаться только на нее при оценке модели. Метрика не дает информации о связи между зависимыми и независимыми переменными.

Она также не информирует о качестве прогнозируемой модели. Следовательно, R2_score необходимо использовать с другими метриками, а затем делать выводы об умении модели предсказывать результат классификации [2].

При многоклассовой классификации чаще всего используются методы усреднения для расчета точности, полноты и *F*-меры. К таким методам относятся *macro*, *weighted* и *micro average*.

Macro average (макроусреднение) – самый простой из перечисленных методов. Значение макроусреднения вычисляется путем взятия среднего арифметического (также известного как невзвешенное среднее) всех оценок ($F1$) для каждого класса (20). Этот метод обрабатывает все классы одинаково, независимо от их приоритетности [2]:

$$macro\_average = \frac{\sum_{i=1}^{n} F1_i}{n}. \quad (20)$$

Weighted average (средневзвешенное) рассчитывается путем вычисления среднего значения всех оценок $F1$ для каждого класса с учетом поддержки каждого класса. Поддержка относится к количеству фактических вхождений класса в наборе данных [2].

Micro average (микроусреднение) вычисляет глобальное среднее значение оценки $F1$ путем подсчета сумм истинно положительных (*TP*), ложноотрицательных (*FN*) и ложноположительных (*FP*) результатов классификации. В первую очередь суммируются соответствующие значения *TP*, *FP* и *FN* по всем классам и подставляются в уравнение $F1$ с целью получения микрооценки $F1$ [15].

Можно сказать, микроусреднение вычисляет долю правильно классифицированных наблюдений от общего числа наблюдений. Данное определение также присуще точности, поэтому в таблицах расчета метрик данная величина может быть подписана как *accuracy*.

ЗАКЛЮЧЕНИЕ

Хорошо известно, что сегодня в Интернете наблюдается большой и постоянно растущий объем нежелательного трафика, включая флуд-атаки и другие атаки типа «отказ в обслуживании», сканирование, фишинг и т. д. Источниками нежелательного трафика обычно являются скомпрометированные хосты, зараженные вредоносным кодом.

Контент-фильтрация помогает предотвратить доступ к вредоносным или опасным веб-сайтам, которые могут содержать вирусы, фишинговые атаки и другие угрозы безопасности или способствовать мошенничеству.

Это особенно важно в обществе, где сетевая безопасность становится всё более приоритетной задачей. В целом, исследование проблематики методик определения типа контента во входящем трафике подчеркивает важность разработки точных и надежных методов для обработки и управления разнообразным контентом в сети. Проблематика связана с постоянным появлением новых форматов и требует обновления методик. Комбинирование различных методов и регулярное обновление помогут повысить точность определения типа контента.

СПИСОК ЛИТЕРАТУРЫ

1. *Skurichina M., Duin R.P.W.* Limited bagging, boosting and the random subspace method for linear classifiers // Pattern Analysis and Applications. – 2002. – Vol. 5 (2). – P. 121–135.

2. Сравнительный анализ современных трендов в области моделей трафика сетей передачи данных / И.Л. Рева, А.В. Иванов, М.А. Медведев, И.А. Огнев // Системы анализа и обработки данных. – 2022. – № 2 (86). – С. 55–68.

3. *Pustejovsky J.*, *Stubbs A.* Natural language annotation for machine learning. – Sebastopol, CA: O'Reilly Media, 2013. – 326 p.

4. *Медведев М.А., Рева И.Л.* Анализ подходов к фильтрации трафика и эффективность применения черных и белых списков // Вестник СибГУТИ. – 2023. – Т. 17, № 1. – С. 107–116.

5. UserGate Web Filter. Руководство администратора. – URL: https://www.rosnovotech.ru/files/UGWF-administrator-manual-ru.pdf (дата обращения: 30.11.2023).

6. *Abe N.* Recent developments in the theory and applications of machine learning // Journal Japanese Society for Artificial Intelligence. – 1999. – Vol. 14 (5). – P. 762. – URL: https://www.ai-gakkai.or.jp/en/published_books/journals_of_jsai/past_journals/in1999/vol14_no5/ (accessed: 30.11.2023).

7. *Архипова А.Б., Медведев М.А., Реутов В.В.* Перспективность использования машинного обучения для классификации сетевого трафика в технологиях доверенного взаимодействия // Безопасность цифровых технологий. – 2022. – № 3 (106). – С. 49–61.

8. A distance-based method for building an encrypted malware traffic identification framework / J. Liu, Z. Tian, R. Zheng, L. Liu // IEEE Access. – 2019. – Vol. 7. – P. 100014–100028. – DOI: 10.1109/ACCESS.2019.2930717.

9. Data mining and knowledge discovery handbook / ed. by L. Rokach, O. Maimon. – 2nd ed. – Springer, 2005. – URL: https://tanthiamhuat.files.wordpress.com/2015/04/data_mining_and_knowledge_discovery_handbook.pdf (accessed: 30.11.2023).

10. *Стрекалов И.Э., Новиков А.А., Лопатин Д.В.* Методы динамической фильтрации вебконтента // Вестник российских университетов. Математика. – 2014. – Т. 19, № 2. – С. 668–669. – URL: https://cyberleninka.ru/article/n/metody-dinamicheskoy-filtratsii-veb-kontenta\ (дата обращения 30.11.2023).

11. *Bruch J., Hinz O., Reinking J.* A process model for requirements engineering of digital ecosystems // Empirical Validation in the Industrial Context. Requirements Engineering. – 2018. – Vol. 23 (1). – P. 49–66.

12. *Бабенко А.А., Бахрачева Ю.С., Алеева А.Р.* Система фильтрации нежелательных приложений интернет-ресурсов // НБИ технологии. – 2020. – Т. 14, № 4. – URL: https://cyberleninka.ru/article/n/sistema-filtratsii-nezhelatelnyh-prilozheniy-internet-resursov (дата обращения 30.11.2023).

13. *Breiman L.* Bagging predictors // Machine Leaning. – 1996. – Vol. 24 (2). – P. 123–140.

14. Performance evaluation of secured network traffic classification using a machine learning approach / A.A. Afuwape, Y. Xu, J.H. Anajemba, G. Srivastava // Computer Standards and Interfaces. – 2021. – Vol. 78. – DOI: 10.1016/j.csi.2021.103545.

15. *Friedman J.H.* Stochastic gradient boosting // Computational Statistics and Data Analysis. – 2002. – Vol. 38 (4). – P. 367–378. – DOI: 10.1016/S0167-9473(01)00065-2.

Рева Иван Леонидович, кандидат технических наук, доцент, декан факультета автоматики и вычислительной техники Новосибирского государственного технического университета. В настоящее время специализируется в области безопасности и защиты информации в информационных системах, основ информационной безопасности. E-mail: reva@corp.nstu.ru

Медведев Михаил Александрович, ассистент кафедры защиты информации Новосибирского государственного технического университета. В настоящее время специализируется в области машинного обучения и информационной безопасности. E-mail: M.medvedev@corp.nstu.ru

Воронцова Инна Владимировна, аспирант кафедры защиты информации Новосибирского государственного технического университета. В настоящее время специализируется в области машинного обучения и информационной безопасности. E-mail: v.vorontcova@corp.nstu.ru

Reva Ivan L., Ph.D., associate professor, Dean of the Faculty of Automation and Computer Engineering, Novosibirsk State Technical University. Currently specializes in the field of security and information protection in information systems, as well as the basics of information security. E-mail: reva@corp.nstu.ru

Medvedev Mikhail A., assistant at the Department of Information Security, Novosibirsk State Technical University. Currently specializes in the field of machine learning and information security. E-mail: M.medvedev@corp.nstu.ru

Vorontsova Inna V., graduate student at the Department of Information Security, Novosibirsk State Technical University. Currently specializes in the field of machine learning and information security. E-mail: v.vorontcova@corp.nstu.ru

DOI: 10.17212/2782-2001-2023-4-69-84

Study of the issues of methods for determining the type of content in incoming traffic[*]

I.L. REVA[a], *M.A. MEDVEDEV*[b], *I.V. VORONTSOVA*[c]

Novosibirsk State Technical University, 20 K. Marx Prospekt, Novosibirsk, 630073, Russian Federation

[a] *reva@corp.nstu.ru* [b] *m.medvedev@corp.nstu.ru* [c] *v.vorontcova@corp.nstu.ru*

Abstract

Content filtering in the context of cybersecurity and trusted environments is an important tactic used to ensure network security and functionality. It works by restricting access to certain websites, emails, files or other content that may contain harmful elements or pose a significant risk of infection. Content filtering ensures the security not only of an individual user's data, but also of an entire network of organizations and institutions, helping to minimize the risk of malicious security breaches.

The study of methods for determining the type of content in incoming traffic is a relevant and important area in the field of information security and network analytics. In today's Internet space, a significant amount of data is transmitted through networks, and one of the key tasks is the classification of this traffic to ensure security and effective network management. Methods for determining the type of content in incoming traffic are a set of algorithms and approaches that allow you to automatically determine what type of data is transmitted over the network. In the course of studying the problems of methods for determining the type of content in incoming traffic, data on network traffic is collected, a data set is selected for training the model, we consider classifier algorithms and focus on metrics for assessing classification efficiency.

[*] *Received 23 June 2023.*

The results of the study can be used to create effective systems for detecting malicious or unwanted content, filtering data, or optimizing the operation of network resources. The study of methods for determining the type of content in incoming traffic is of practical importance and can be applied in various fields, including information security, network analytics, monitoring of network resources and optimization of network processes.

Keywords: artificial intelligence, computer networks, semantic content filtering, network traffic, dataset, cybersecurity, True-traffic, False-traffic, true positive, true negative, false positive, false negative, precision, recall, F-measure

REFERENCES

1. Skurichina M., Duin R.P.W. Limited bagging, boosting and the random subspace method for linear classifiers. *Pattern Analysis and Applications*, 2002, vol. 5 (2), pp. 121–135.

2. Reva I.L., Ivanov A.V., Medvedev M.A., Ognev I.A. Sravnitel'nyi analiz sovremennykh trendov v oblasti modelei trafika setei peredachi dannykh [Comparative analysis of modern trends in the field of traffic models of data transmission networks]. *Sistemy analiza i obrabotki dannykh = Analysis and Data Processing Systems*, 2022, no. 2 (86), pp. 55–68.

3. Pustejovsky J., Stubbs A. *Natural language annotation for machine learning*. Sebastopol, CA, O'Reilly Media, 2013. 326 p.

4. Medvedev M.A., Reva I.L. Analiz podkhodov k fil'tratsii trafika i effektivnost' primeneniya chernykh i belykh spiskov [Analysis of traffic filtering approaches and the effectiveness of blacklisting and whitelisting]. *Vestnik SibGUTI = The Herald of the Siberian State University of Telecommunications and Information Science*, 2023, vol. 17 (1), pp. 107–116.

5. *UserGate Web Filter. Rukovodstvo administrator* [UserGate Web Filter. Administrator's Guide]. Available at: https://www.rosnovotech.ru/files/UGWF-administrator-manual-ru.pdf (accessed 30.11.2023).

6. Abe N. Recent developments in the theory and applications of machine learning. *Journal Japanese Society for Artificial Intelligence*, 1999, vol. 14 (5), p. 762. Available at: https://www.ai-gakkai.or.jp/en/published_books/journals_of_jsai/past_journals/in1999/vol14_no5/ (accessed 30.11.2023).

7. Arkhipova A.B., Medvedev M.A., Reutov V.V. Perspektivnost' ispol'zovaniya mashinnogo obucheniya dlya klassifikatsii setevogo trafika v tekhnologiyakh doverennogo vzaimodeistviya [The perspective of using machine learning to classify network traffic in trusted interaction technologies]. *Bezopasnost' tsifrovykh tekhnologii = Digital Technology Security*, 2022, no. 3 (106), pp. 49–61.

8. Liu J., Tian Z., Zheng R., Liu L. A distance-based method for building an encrypted malware traffic identification framework. *IEEE Access*, 2019, vol. 7, pp. 100014–100028. DOI: 10.1109/ACCESS.2019.2930717.

9. Rokach L., Maimon O., eds. *Data mining and knowledge discovery handbook*. 2nd ed. Springer, 2005. Available at: https://tanthiamhuat.files.wordpress.com/2015/04/data_mining_and_knowledge_discovery_handbook.pdf (accessed 30.11.2023).

10. Strekalov I.E., Novikov A.A., Lopatin D.V. Metody dinamicheskoi fil'tratsii veb-kontenta [Methods of web-content dynamic filtering]. *Vestnik rossiiskikh universitetov. Matematika = Russian Universities Reports. Mathematics*, 2014, vol. 19 (2), pp. 668–669. Available at: https://cyberleninka.ru/article/n/metody-dinamicheskoy-filtratsii-veb-kontenta\ (accessed 30.11.2023).

11. Bruch J., Hinz O., Reinking J. A process model for requirements engineering of digital ecosystems. *Empirical Validation in the Industrial Context. Requirements Engineering*, 2018, vol. 23 (1), pp. 49–66.

12. Babenko A.A., Bakhracheva Yu.S., Aleeva A.R. Sistema fil'tratsii nezhelatel'nykh prilozhenii internet-resursov [System for filtering unwanted applications of internet resources]. *NBI tekhnologii = NBI Technologies*, 2020, vol. 14 (4). Available at: https://cyberleninka.ru/article/n/sistema-filtratsii-nezhelatelnyh-prilozheniy-internet-resursov (accessed 30.11.2023).

13. Breiman L. Bagging predictors. *Machine Leaning*, 1996, vol. 24 (2), pp. 123–140.

14. Afuwape A.A., Xu Y., Anajemba J.H., Srivastava G. Performance evaluation of secured network traffic classification using a machine learning approach. *Computer Standards and Interfaces*, 2021, vol. 78. DOI: 10.1016/j.csi.2021.103545.

15. Friedman J.H. Stochastic gradient boosting. *Computational Statistics and Data Analysis*, 2002, vol. 38 (4), pp. 367–378. DOI: 10.1016/S0167-9473(01)00065-2.

Для цитирования:

Рева И.Л., Медведев М.А., Воронцова И.В. Исследование проблематики методик определения типа контента во входящем трафике // Системы анализа и обработки данных. – 2022. – № 4 (92). – С. . – DOI: 10.17212/2782-2001-2023-4-69-84.

For citation:

Reva I.L., Medvedev M.A., Vorontsova I.V. Issledovanie problematiki metodik opredeleniya tipa kontenta vo vkhodyashchem trafike [Study of the issues of methods for determining the type of content in incoming traffic]. *Sistemy analiza i obrabotki dannykh = Analysis and Data Processing Systems*, 2022, no. 4 (92), pp. . DOI: 10.17212/2782-2001-2023-4-69-84.