УДК 004.855.5

Анализ перспективных подходов и исследований по классификации потоков трафика для поддержания QoS методами ML в SDN-сетях

В. Ю. Деарт, В. А. Маньков, И. А. Краснова

Одной из важнейших задач, существующих в современных сетях, является поддержка качества обслуживания QoS на соответствующем уровне, что может быть достигнуто путем применения различных механизмов управления трафиком. Но для того, чтобы поддерживать параметры QoS в надлежащем состоянии, требуется знать типы трафика, проходящие по сети. С учетом высокотехнологичных и производительных сетей, таких как SDN-сети, классификация трафика обычными способами становится практически невозможной. На помощь приходят методы интеллектуального анализа данных, в т.ч. методы машинного обучения. В статье анализируются основные перспективные подходы к классификации трафика в режиме реального времени для поддержания QoS в SDN-сетях методами ML, а также представлен сравнительный обзор наиболее выдающихся работ.

Ключевые слова: Machine Learning (ML), SDN, QoS, классификация трафика.

1. Введение

С развитием телекоммуникаций появляется все большее количество новых услуг в различных сферах сетевых взаимодействий, особенно это проявляется в динамически перестраиваемых высокопроизводительных SDN-сетях (Software Defined Networking). Каждый тип услуги выставляет определенные требования по качеству обслуживания (Quality of Service, QoS) передаваемого трафика. Для поддержания условий уровня обслуживания оператору сети необходимо располагать достоверной информацией о типах предоставляемого сервиса, что в современных сетях становится затруднительным.

С другой стороны, в последнее время широкое развитие в различных сферах человеческой деятельности получили методы интеллектуального анализа данных, в частности методы машинного обучения (Machine Learning, ML).

Цель данного обзора – показать применяемые в SDN-сетях подходы к решению задач классификации трафика и поддержки QoS в режиме реального времени методами машинного обучения. Рассматриваются не только сами алгоритмы, но и проблемы, возникающие на всех этапах реализации выбранных подходов. Помимо прочего, приводятся результаты анализа наиболее перспективных работ, ведущихся в этой области.

Статья имеет следующую структуру: в разделе 2 дается описание обзоров, смежных с тематикой этого исследования, в разделе 3 показаны особенности SDN-сетей, подчеркивающие преимущества использования методов ML в SDN-сетях, в разделе 4 формулируется задача классификации трафика в режиме реального времени и дается подробный анализ существующих подходов на всех этапах классификации трафика. Раздел 5 посвящен анализу имеющихся работ и исследований в этой области. В заключении приведены общие выводы по работе, а в приложение вынесены сравнительные таблицы, включающие в себя сводную характеристику

20 самых актуальных статей по классификации трафика в режиме реального времени с поддержкой параметров QoS методами ML в SDN-сетях с описанием способов решений проблем, перечисленных в статье.

2. Анализ методов классификации трафика в современных транспортных сетях

В работе [1] дан подробный обзор подходов к задачам классификации сетевого трафика. Перечислены основные возникающие при этом проблемы, связанные с передачей трафика по сети, и перспективные способы их решения. Но в работе приводится очень мало информации относительно стремительно развивающегося направления классификации – классификации с помощью методов машинного обучения. Также не учитываются особенности построения сети, преимущества и недостатки, которые могут быть с помощью них получены. В частности, в статье нет упоминания о сетях SDN и анализа работ, посвященных классификации в таких сетях.

Авторы [2] анализируют приложения, использующие методы машинного обучения в различных сетевых технологиях, в [3–4] – классификацию трафика, в [5–6] – сетевые приложения, применяемые в SDN-сетях. Несмотря на подробное изложение самих подходов, проблемы классификации, а также трудности, возникающие при проверке собственных методов, в статьях не описываются.

В [7], помимо краткого обзора статей, рассматриваются средства моделирования SDN-сетей, использующиеся в работах. По результатам исследования авторов [7] преобладающим инструментом для моделирования SDN-сетей является Mininet [8].

В [9] дано краткое описание всех выдающихся работ, проводимых с применением методов машинного обучения в период с 1993 по 2013 гг., но не сделано никаких теоретических обобщений и про классификацию трафика методами машинного обучения написано очень мало. В статье [10] приводится описание 18 работ по классификации трафика, основанное на методах ML за 2004–2007 года, но сети SDN не вошли в список этих исследований. Более того, за 13 лет вышло довольно много новых работ и технологии ушли далеко вперед. Авторы [11] представляют обзор с применением методов искусственного интеллекта к SDN-сетям, но практически не рассматривают классификацию трафика в режиме реального времени и не затрагивают вопросы сбора базы данных для классификации.

В обзорах [12–15] описываются подходы к классификации трафика в целях безопасности, в т.ч. с целью обнаружения вторжений. Статьи [12–14] дают описание методов для традиционных сетей, а [15] – для SDN-сетей. Существует также ряд обзоров, посвященных определенным технологиям организации сети: [16–17] – беспроводные сети, [18] – гетерогенные, [19] – самоорганизующиеся и т.д.

Чаще всего в существующих обзорах подробно рассматривается какое-то одно из направлений, а остальные остаются неохваченными. Таким образом, не обнаружено обзора, подробно описывающего работы и возникающие у исследователей проблемы по классификации трафика для целей поддержания QoS в режиме реального времени методами ML в SDN-сетях. В этой статье впервые предпринята попытка систематизировать существующие в этом направлении подходы, выявить общие проблемы, проанализировать их решения на каждом этапе реализации, подчеркнув достоинства и недостатки различных способов, а также провести обзор наиболее перспективных работ по теме исследования.

3. Особенности SDN

SDN-сети (Software-Defined Networking, [20]) – способ организации сетей, при котором плоскость управления (Control Plane) отделена от плоскости передачи данных (Data Plane) и представляет собой единую структуру (SDN-контроллер), способную предоставить централизованный контроль и доступ ко всем сетевым элементам. Посредством плоскости приложений (Application Plane) администратор сети способен повлиять на обработку трафика, создавая новые приложения и взаимодействуя через API-интерфейс с контроллером.

Становление и развитие архитектуры SDN-сетей стимулирует появление все новых подходов к классификации потоков сетевого трафика на основе методов ML.

Преимущества SDN-сетей, используемые при классификации трафика:

– централизованный мониторинг сети с любых узлов;

– расширенная информация о потоках и пакетах по портам, узлам, сетям, в т.ч. информация о каждом пакете, размере, времени отправки, флагах TCP, заголовках приложений, времени прохождения пакетом определенного этапа обработки коммутатором (P4-сети, Programming Protocol-Independent Packet Processors) и т.д.;

– централизованный анализ данных без дополнительной нагрузки на сетевые элементы;

– управление сетью на основе множества факторов;

– исследование трафика в режиме реального времени;

– возможность создания собственных протоколов и приложений для мониторинга, классификации и принятия решений по управлению трафиком.

В SDN-сетях имеется возможность проводить интеллектуальную классификацию трафика на разных плоскостях. В [21] классификация трафика проводится в плоскости Data Plane и позволяет выявить Elephant-потоки (очень большие потоки, занимающие большую долю пропускной полосы). В [22] потоки классифицируются в плоскости Control Plane, а в [23–24] в плоскости Application Plane. В [25] создали свою дополнительную плоскость, в которой производится интеллектуальная обработка данных о потоках.

Основные сферы применения методов ML в SDN-сетях: классификация потоков трафика [26–43]; динамическое определение оптимального маршрута с учетом состояния соединений и сетевых элементов [25, 44]; прогнозирование параметров QoS/ QoE (Quality of Experience) на основе имеющейся статистической информации [45–48]; управление ресурсами [44, 49–51]; управление политиками безопасности, в т.ч. обнаружение вторжений [52–53]; проведение статистических исследований, предсказание занятости каких-либо каналов связи или использование определенных приложений [40, 44].

Существует классификация трафика в режиме реального времени и классификация трафика вне режима реального времени.

Классификация трафика в режиме реального времени может применяться с целью маркировки трафика и поддержки параметров QoS либо для систем обнаружения вторжений. Особенности такой классификации заключаются в следующем:

1. Системы, работающие в таком режиме, должны быть быстродействующими, т.к. результат классификации необходимо получить быстро, иначе ответ будет уже неактуален.

2. Классификация проводится только по n первым пакетам или t первым мс, где параметры n и t стремятся минимизировать. Невозможность просмотра всего потока накладывает определённые ограничения на матрицу признаков для классификации, в частности, в качестве признаков не могут быть выбраны длительность всего потока, количество пакетов в потоке, средние параметры по потоку.

3. Методика сбора информации о пакетах потока должна быть достаточно гибкой и способной снимать и оперативно передавать лишь необходимое количество информации, не создавая дополнительных нагрузок на сеть.

4. Результаты классификации должны быть точными, что делает невозможным применение непроверенных и слабых подходов.

Классификация трафика вне режима реального времени может применяться, например, для статистических исследований с целью выявления наиболее популярных приложений и услуг. Такие методы позволяют накапливать всю информацию о потоке, но делают невозможным их применение в режиме реального времени.

4. Задача классификации потоков трафика в режиме реального времени

4.1. Этапы классификации потоков трафика

Под классификацией потоков трафика подразумевается разделение различных образцов трафика (потоков, сессий, групп пакетов) на определенные классы (группы, категории).

Основные этапы классификации потоков трафика в режиме реального времени в сетях SDN:

1. Формирование базы данных для классификации (создание базы данных, разметка базы данных и сбор статистической информации о потоках трафика).
2. Формирование матрицы признаков (определение шаблонов трафика, построение матрицы признаков и определение классов трафика как результата работы алгоритма).
3. Определение классов с применением методов ML (методы классификации, инструменты для интеллектуального анализа данных и анализ работоспособности предложенного алгоритма).

В дальнейшем в статье последовательно и подробно описывается каждый из этапов, возникающие проблемы и известные способы их решения.

4.2. Формирование базы данных для классификации

Одной из сложнейших задач при классификации сетевого трафика, замедляющих развитие этого направления в целом, является формирование базы данных трафика, т.к. сетевой трафик – разнородный, быстроменяющийся, сложный и закрытый в целях безопасности от сторонних наблюдателей. Не существует уникальных общедоступных баз данных. Все попытки создания подобной базы сопряжены с определенными трудностями, из-за которых в результате базы получаются с существенными недостатками, ограничивающими область их применения. В итоге большинство исследователей проводят эксперименты на различных базах данных, собранных в разное время при различных условиях, и итоги их работы довольно сложно сопоставить и сравнить между собой.

Способы создания базы данных для классификации (рис. 4.1)

1. Моделирование потоков трафика в условиях лабораторного стенда (инвазивный трафик):

а). Имитация работы приложений с использованием различных генераторов трафика. Существует множество приложений и устройств, которые способны моделировать трафик (например, D-ITG-генератор, Distributed Internet Traffic Generator). В этом случае заранее известен трафик, который проходит классификацию, его легко промаркировать и выделить среди остальных потоков, что является неоспоримым преимуществом данного способа. Недостатком же является излишняя «искусственность» сетей, при которой зачастую ситуация со смоделированным трафиком очень сильно отличается от реальной работы сети. Кроме того, уровень приложений такого трафика обычно заполняется случайными либо повторяющимися значениями, что исключает возможность применения классификации трафика на основе данных прикладного уровня.

б). Программирование на сырых сокетах (raw socket) и создание отдельных пакетов на основе Python Scapy. Сырые сокеты – специальный API-интерфейс (Application Programming Interface), который позволяет принимать и передавать необработанные пакеты без добавления

в них каких-либо дополнительных заголовков. Преимуществом метода является неограниченная возможность по созданию трафика, т.к. могут быть получены абсолютно любые пакеты и трассы. Недостатками так же, как и в использовании генераторов трафика выступает не реалистичность полученных баз трафика, а также повышенные технические сложности организации эксперимента.

в). Использование открытых баз данных. В открытом доступе находится довольно большое количество открытых баз данных [54–56]. Преимуществом способа является простота использования и возможность быстрого доступа к разнообразным сервисам, которые зачастую очень сложно даже смоделировать на тестовом стенде. Основными недостатками являются отсутствие маркировки потоков, ограниченность представленных сервисов, фиксированные лабораторные условия и быстрое «устаревание» информации.

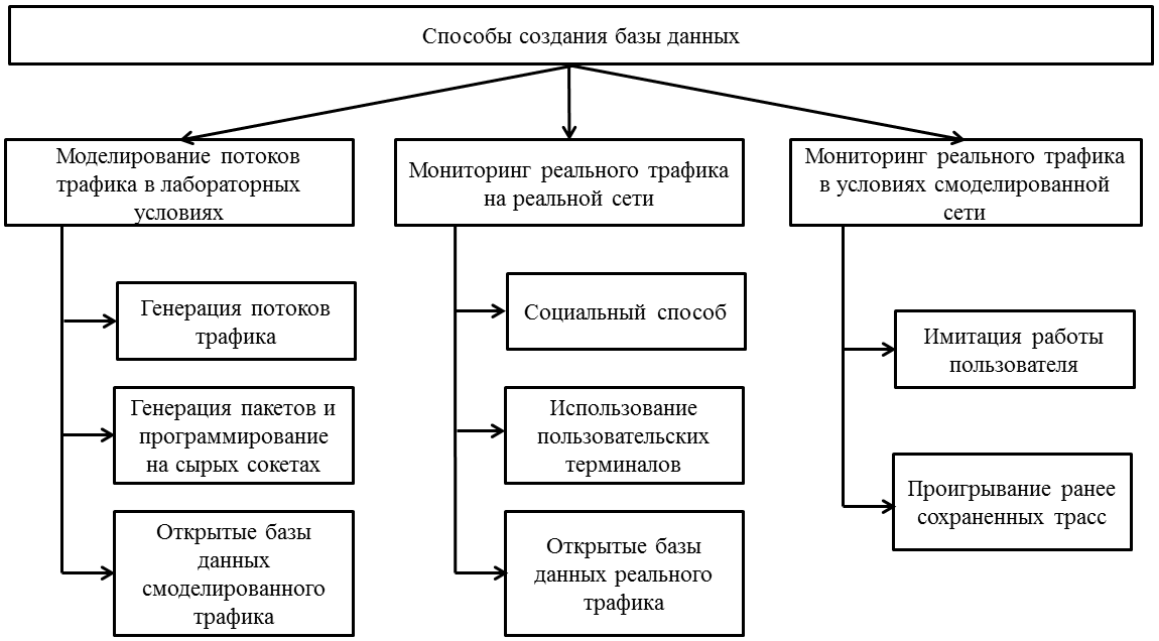

Рис. 4.1. Способы создания базы данных

2. Мониторинг реального трафика на реальной сети (неинвазивный трафик):

а). Социальный способ с привлечением партнеров, учащихся и сотрудников университетов или пользователей созданной сети; в коммерческой среде может мотивироваться получением определенных бонусов/баллов либо денежных вознаграждений клиентам. Преимуществами такой сети, безусловно, являются реалистичность и актуальность собранной базы данных, большие объемы трафика, возможность получения разнообразного вида трафика, который не всегда может быть представлен или учтен в условиях лабораторных стендов. Кроме того, такой подход не вносит дополнительных нагрузок на сеть. Недостатки метода: отсутствие маркировки как таковой; нестабильность условий – невозможность проведения экспериментов с равными условиями; сложность организации социальной структуры – привлечение большого количества людей и, как следствие, затруднения при повторной организации подобных экспериментов; политика конфиденциальности – передача трасс трафика, проходящего по сети даже с целью исследования, является небезопасной, поэтому над трассами проводятся процедуры анонимизации – удаление полезной информации и шифрование трассы, что ограничивает варианты применения методов машинного обучения.

б). Использование собственных пользовательских терминалов. В некоторых исследованиях можно встретить базы данных, полученные при помощи мониторинга собственных пользовательских терминалов, в качестве которых обычно выступают ПК и ноутбуки одного или нескольких участников эксперимента. Такой подход намного проще организовать в социальном и техническом плане, в некоторых вариантах возможна маркировка трафика, но, по срав-

нению с предыдущим способом, получается значительно меньшая по объему и худшая по качеству база данных. Кроме того, подобные эксперименты позволяют снимать трафик только в одной точке сети – на оконечном устройстве, не имея представления о том, что происходит в остальных узлах сети.

в). Использование выложенных в интернете баз данных трафика. Подобные базы обладают теми же недостатками, что и аналогичные базы данных с моделированными потоками [56].

3. Мониторинг реального трафика в условиях смоделированной сети:

а). Получение потоков трафика с использованием различных приложений и программ, имитирующих работу пользователей. Ввиду сложности создания автоматизированных систем с использованием голосового трафика такие исследования включают в себя в основном Web-трафик, DNS (Domain Name Service) и FTP (File Transfer Protocol).

б). Проигрывание ранее записанных трасс трафика, полученных в реальных сетях. Например, tcpreplay позволяет проигрывать трассы по их .pcap-записям. Таким образом, зная сценарии определенных реальных трасс, можно воссоздать новую трассу в других условиях. Однако остается некоторая неопределенность при использовании такого подхода: с одной стороны, имеется сценарий поведения, которым руководствуется один из узлов, а с другой – трасса получена с одними условиями, а воспроизводится с другими, что влияет на ее параметры, например, NAT (Network Address Translation), VLAN (Virtual Local Area Network), TCP/UDP-порты и т.д.

Разметка базы данных

Помимо непосредственного сбора базы данных, необходимо проводить и ее разметку, т.е. определять принадлежность образца к определенному классу. Способы разметки базы данных (рис. 4.2):

1. Априорные способы – способы разметки трафика непосредственно во время сбора трафика или до его прохождения по сети:

а). Автоматическая разметка с использованием специальных инструментов, включающих в себя разметку по портам, разметку с помощью DPI (Deep Packet Inspection), системы определения сервиса на основе ответов DNS и т.д. Такие методы можно применять на реальных и достаточно больших сетях, но они имеют ряд недостатков: маркировка по портам достаточно простая, но в сетях с динамическим изменением портов не очень эффективна. Системы с DPI технически более сложно реализуемы и работают не во всех случаях (например, с зашифрованным трафиком).

б). Разметка перед отправкой пакетов в моделируемых трассах – ряд генераторов трафика позволяет менять поля TOS (Type of Service) / DSCP (Differentiated Services Code Point) у отправляемых пакетов. Также подобный способ может применяться при проигрывании известных трасс с изменением соответствующих полей. Другой разновидностью этого способа является разметка потоков с помощью известных запущенных процессов или с помощью специальных приложений, таких как tcptrace.

в). Разметка при формировании пакетов используется в случаях с программированием на сырых сокетах, т.е. с воссозданием работоспособного приложения либо имитации пакета с помощью, например, Scapy. Это довольно сложный и специфичный способ разметки трафика.

г). Изменение параметров интерфейса в зависимости от прохождения того или иного приложения через сетевой интерфейс в определенные моменты времени. Такой способ может применяться при условии однозначного и достоверного представления о работающих сервисах в выбранные временные промежутки.

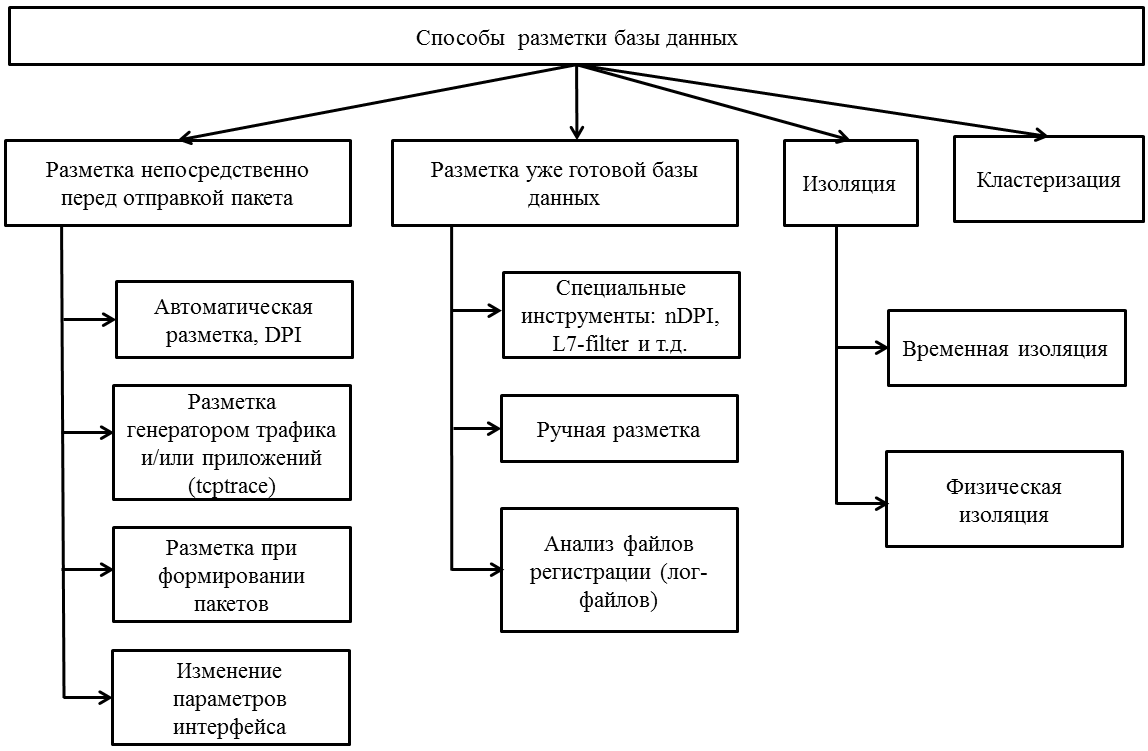

Рис. 4.2. Способы разметки базы данных

2. Апостериорные виды маркировки трафика – методы, применяемые для разметки уже готовой базы данных трафика:

а). Разметка с использованием специальных инструментов, таких как PACE, OpenDPI, NDPI, L7-filter, Libprotoident, NBAR, или инструментов, работающих на совокупности их результатов, является очень популярной среди исследователей и встречается в ряде работ [57]. С технической точки зрения методы проработаны и выверены на различных трассах, могут применяться в различных условиях, даже при получении базы данных со стороннего источника. Но при подобном методе существует серьезная проблема под названием «ground truth problem», что, по сути, является проблемой качества используемого эталона. Кроме того, что выбранные эталонные методы могут содержать некоторые ошибки при идентификации, они все имеют ограниченное количество определяемых классов приложений, что не позволяет исследователям, использующим эти инструменты, выявить какие-либо новые классы. В условиях динамически развивающихся сетей, в особенности сетей SDN, в которых любой участник сети может написать новое приложение, перечисленные методы будут давать большие ошибки, т.к. скорость обновления базы классификации не позволяет в полном объеме учитывать всё новые и новые классы. Более того, перечисленные инструменты могут отделить одно приложение от другого, например, WhatsApp и Telegram, но не в состоянии разделить между собой мелкозернистые типы услуг (отдельные сервисы), такие как передача сообщений в Telegram от голосового вызова в Telegram, которые должны иметь различный приоритет по качеству обслуживания. Также перечисленные инструменты хорошо работают на сравнительно небольших базах данных, а при анализе достаточно объемного трафика требуют больших вычислительных мощностей.

б). Ручная разметка, если известны использованные виды приложений – трудоемкий процесс, вероятность ошибочной маркировки при котором очень высока.

в). Разметка по файлам регистрации (логи) генераторов трафика, например, D-ITG. Метод хорошо подходит при использовании генераторов трафика, но его сложно применить при работе с реальными видами трафика.

3. Разметка с помощью изоляции. Точные и достаточно простые для реализации на моделируемом стенде способы, но практически нереализуемые на реальных сетях:

а). Временная изоляция – в определенный момент времени работает только один класс трафика, в другой момент времени работает другой.

б). Физическая изоляция – разделение классов трафика по разным источникам, портам и т.д. Один сервис работает только по одному порту, другой – по другому и т.д.

4. Разметка с помощью кластеризации. Кластеризация – способ обучения без учителя, способный разбить имеющуюся группу объектов по определенным признакам на классы. Т.к. сама по себе является неточной, для верификации других методов обычно не применяется.

Сбор статистической информации о потоках трафика

В случаях, когда база данных трафика поступает не из внешнего источника, важным моментом является выбор метода сбора трафика из сети. Необходимо соблюдать баланс и находить оптимальное соотношение между частотой обновления информации о потоке, получаемыми параметрами потока и дополнительной нагрузкой на контроллер, сетевые элементы и саму сеть.

Основные способы сбора информации о потоках, применяемые в исследованиях:

1. Мониторинг моделируемых потоков при помощи файлов регистрации (лог-файлов) генератора. Подход возможен при использовании соответствующих генераторов трафиков. Довольно прост в использовании, не несет дополнительной нагрузки на сетевые элементы, сеть и контроллер, но информация о потоках является очень ограниченной.

2. Инструменты мониторинга, связанные с получением информации о пакетах. Наиболее популярными являются протоколы tcpdump, libpcap, NetFlow, SNMP (Simple Network Management Protocol) и т.д. Они дают очень подробную информацию о пакетах, способны группировать их в потоки, работают с различными фильтрами, но, тем не менее, содержат очень много избыточной информации, которая затрудняет их использование в режиме реального времени, т.к. требуют предобработки на узле либо маленькой частоты обновления информации. Кроме того, не все устройства поддерживают те или иные способы мониторинга.

3. Инструменты мониторинга, связанные с получением информации о потоках. Например, встроенные агенты мониторинга контроллеров ODL (OpenDayLight) и ONOS (Open Network Operating System) [24]. Также применяются различные инструменты, построенные на возможностях протокола OpenFlow [58–59]. Обычно такие средства не вносят больших нагрузок на сеть и позволяют регулировать частоту сбора информации, например, PayLess, но предлагают ограниченный набор данных по потокам без возможности получения подробной информации о конкретных пакетах, что сужает сферу применяемых с ними методов ML.

4. Собственные адаптивные инструменты мониторинга, основанные на возможностях SDN-сетей. Развивающиеся SDN-сети позволяют не только создавать собственные приложения с использованием API-интерфейса, но и задавать логику обработки пакета на коммутаторе (p4-сети), благодаря чему открываются неограниченные возможности, связанные со сбором статистической информации о пакетах с любой частотой сбора, любой информацией, в т.ч. временем прохождения определенных этапов обработки коммутатора, информацией о каждом пакете или только об определенных полях определенных пакетов, отсчет которых может проводиться не только с помощью различных фильтров по портам, приложениям, IP-адресам, но и по порядковому номеру пакета или при соблюдении определенных условий, например, при накоплении определенного количества байт с потока. Благодаря такой архитектуре метод открывает возможности выполнения предварительных расчетов на узле или нересурсоёмкой проверки по DPI для отдельных пакетов, не внося при этом никаких дополнительных нагрузок на сеть, сетевые элементы и контроллер [60].

4.3. Формирование матрицы признаков

Из полученной базы данных формируется матрица признаков, в которой для каждого представленного образца (шаблона) трафика рассчитывается ряд параметров (признаков) и ставится в соответствие маркер класса.

Шаблоны трафика

В качества образцов трафика выступают сгруппированные по каким-либо общим свойствам пакеты. Наиболее популярные примеры шаблонов трафика:

1. Односторонний UDP- или TCP-поток, идентифицируемый общими значениями 5-Tuple-параметров (IP-адресов источника и назначения, портов источника и назначения и номера транспортного протокола).

2. Двухсторонний UDP- или TCP-поток, идентифицируемый общими значениями 5-Tuple-параметров. Рассмотрение в качестве шаблонов двухсторонних потоков, с одной стороны, расширяет набор признаков, а с другой – усложняет процесс классификации, т.к. потоки довольно часто бывают ассиметричными как по работе сервиса, так и по используемым сетевым маршрутам.

3. Односторонняя или двухсторонняя сессия, состоящая из разных потоков, образованных с целью создания и поддержания определенного сервиса в рамках одного сеанса. Например, голосовой вызов с помощью приложения Telegram сопровождается работой протокола DNS, RTP (Real-Time Transport Protocol), RTCP (Real-Time Transport Control Protocol) и т.д.

Матрица признаков

Для каждого из образцов трафика составляется вектор признаков (в некоторых статьях – атрибутов), который подбирается в зависимости от того, что принимается за шаблон трафика и каким образом происходит обработка пакетов. Признаки условно можно поделить на следующие категории:

1. Базовые статистические данные одностороннего потока. К ним относятся: время прихода *n*-го пакета на интерфейс, размер *n*-го пакета, записанные для каждого из *n* пакетов, общее количество пакетов. Тип протокола, записанный в поле «порт» источника и назначения TCP/UDP-пакета, не относят к признакам, т.к. он может значительно улучшить результаты классификации в случаях, когда протоколы соответствуют используемым номерам, и позволить допустить множество ошибок при динамически изменяющихся портах.

2. Расширенные статистические данные одностороннего потока. Эти данные получены на основе базовых путем расчета статистических характеристик, например: среднее время прибытия пакета на интерфейс, средний размер пакета, минимальные и максимальные параметры, СКО, средняя пропускная способность потока на интерфейсе, параметр Херста и т.д.

3. Данные заголовков протокола TCP. При исследовании TCP-сессий в качестве атрибутов часто используют такие параметры, как общее количество пакетов с флагами SYN, ACK, ACK-PSH (отправленных или полученных), размер окна источника и назначения в пакетах с флагами SYN и т.д.

4. Расширенные статистические характеристики двухстороннего потока: направление передачи, отношение количества переданных/полученных пакетов, байт и т.д. Также могут быть использованы различные агрегированные сущности при формировании групп пакетов. Так, например, в работах [57] использовались «порции» данных, состоящие из двух TCP-запросов и одного ACK-ответа, а в работе [32] – «раунды», состоящие из всех пакетов на уровне приложений, отправляющихся в одну сторону и подтверждаемых другой стороной.

5. Глубокий анализ пакета – в качестве признаков могут применяться любые данные на уровне приложений, записанные в поле полезной нагрузки и рассчитанные на их основе. Чаще всего трафик в таком случае представляет собой некий «след» или «сигнатуру». Сложность заключается в получении непосредственно самих данных, как описано в предыдущем пункте статьи, и расширяемости метода применительно к разным протоколам.

Основные проблемы, возникающие при обработке данных

Одним из вопросов, возникающих у исследователей, является обработка TCP-сессий. Протокол TCP работает с поддержкой установления соединения и отправляет служебные пакеты (SYN, SYN-ACK, ACK) в начале сессии. Для того чтобы они не оказывали влияния на классификацию, во многих работах информацию о первых трех пакетах TCP-сессии убирают из базы данных. Другие же, напротив, используют информацию, получаемую с помощью TCP-флагов,

и формируют шаблоны трафика на уровне не TCP-сессии, а вышележащих протоколов (HTTP, MSN и т.д.).

В разных сетях действует разный размер MTU (Maximum Transmission Unit, максимальная единица передачи), которая, в свою очередь, влияет на наличие фрагментации трафика. Для того чтобы фрагментация не влияла на результаты классификации трафика, некоторые исследователи убирают данные TCP- и UDP-заголовков, а размеры фрагментированных частей полезной нагрузки соединяют друг с другом.

При формировании потоков может оказаться, что некоторых пакеты потока были потеряны. В таких случаях из базы данных удаляют шаблоны потоков с потерянными пакетами, пренебрегают потерями либо описывают потерянные пакеты по средним характеристикам других пакетов/шаблонов трафика.

В случаях некачественного сбора базы данных возможно появление дублирующей информации – обычно повторные записи удаляются из общей базы данных.

При сборе данных на реальной сети можно заметить, что некоторые приложения используются намного чаще других, так что их потоков в выборке оказывается в тысячи раз больше, чем всех остальных, а некоторые – намного реже и встречаются всего лишь по несколько экземпляров. Редкие, по сравнению с другими, виды трафика удаляются, т.к. невозможно построить классификатор, обучаясь всего на нескольких примерах. Слишком частые потоки тоже частично удаляются, потому что иначе их большой количественный перевес может послужить причиной дисбаланса образцов.

Также важно обращать внимание на количество признаков в матрице классификации. Для повышения быстродействия классификаторов рекомендуется сокращать избыточные малоинформативные признаки. Информативность признаков можно определить с помощью инструментов ML и способами понижения размерности.

Классы трафика

Наиболее популярные классы, которые используют в качестве результатов классификации с целью поддержки QoS в сетях SDN:

1. Типы приложений: TeamViewer, Skype, YouTube, Gmail и т.д.
2. Тип протокола: FTP, SSH (Secure Shell), DNS и т.д.
3. Категории сервисов: WEB, почта, голос, данные и т.д.
4. Тип характера трафика: Elephant- и Mice-потоки (очень большие и очень маленькие потоки соответственно), интерактивный/не интерактивный трафик и т.д.
5. Тип определенного сервиса в пределах приложения: Skype-голос, Skype-сообщение и т.д.

4.4. Определение классов с применением методов ML

Наиболее распространенные алгоритмы классификации сетевого трафика методами ML

Алгоритмы классификации трафика методами ML, встречающиеся в современных исследованиях, можно разделить на чистые методы ML и методы ML с применением дополнительных алгоритмов.

В качестве дополнительных методов используются: проверка пакетов по DPI, информация, полученная с помощью DNS-запросов, проверка по портам общеизвестных протоколов (SSH, SNMP) и др. Чаще всего такие способы позволяют выявить часть общеизвестных потоков и упростить задачу классификации методами ML.

Наиболее популярные алгоритмы ML можно представить несколькими группами [61]:

1. Классическое обучение с учителем – наиболее распространенные методы, готовые построить модель по готовым шаблонам трафика. Достаточно точны и просты в использовании. Основной недостаток – сложность внедрения нового потока, для этого требуется получить шаблоны трафика нового потока и переобучить модель. К методам обучения с учителем относят: Support Vector Machine (SVM), Decision Tree (DT), Наивный Байес и др.
2. Классическое обучение без учителя позволяет обучать модель даже в отсутствие шаблонов трафика. Основные недостатки: низкая точность алгоритма и необходимость сбора

определенной базы данных, на основании которой строятся кластеры. Примеры: Linear discriminant analysis (LDA), Expectation-Maximization (EM), Gaussian Mixture Model (GMM), k-means и др.

3. Обучение с подкреплением. Данные для обучения модели не требуются, но требуется подкрепление – т.е. ответ на вопрос, правильно ли алгоритм классифицировал пришедший поток. Достоинством является возможность добавления новых классов. Примеры: Q-Learning, Deep Q-Learning (DQN), State–action–reward–state–action (SARSA).

4. Нейросети (NN, Neural Network) и глубокое обучение работают в несколько слоев с заданием весов на каждом уровне. Обладают высокой точностью и позволяют решить даже самые сложные задачи, но требуют больших обучающих наборов данных и больших вычислительных мощностей. Примеры: Multilayer Perceptron (MLP), Radial Basis Function (RBF), AutoEncoder Neural Network (ANN), Convolutional NN (CNN) и т.д.

5. Ансамблевые методы – способы, при которых несколько алгоритмов классических алгоритмов работают вместе. Примеры: Random Forest (RF), Stochastic Gradient Boosting (SGB) и Extreme Gradient Boosting (EGB) и т.д.

Инструменты для классификации

Основные инструменты, применяемые для классификации сетевого трафика в различных исследованиях:

1. Платформы WEKA и RapidMiner – интеллектуальные приложения, позволяющие внедрять их в собственные проекты. Отличаются простотой использования, достаточно неплохи при проверке какой-либо небольшой гипотезы, начальной стадии изучения методов ML, но затруднительны при внесении каких-либо значимых изменений в данные, алгоритмы и их комбинации. Способности исследователя в классификации, по сути, ограничиваются их возможностями.

2. Языки программирования R или более популярный Python (библиотека Scikit-learn). Отличаются наличием объемных справочных документаций, по сравнению с WEKA и RapidMiner имеют большое разнообразие алгоритмов, позволяют внедрять алгоритмы непосредственно в проект: т.к. большое количество устройств работает на Python, то созданные на нем алгоритмы ими легко поддерживаются. Python позволяет взаимодействовать с Ansible, потребляет значительно меньше ресурсов и работает быстрее. Сложность заключается в более высоком пороге вхождения исследователя, т.к. требуется определенный уровень понимания языков программирования.

3. Statgraphics, Statistica, MATLAB и др. – сильные инструменты для интеллектуального анализа данных, но не работают в режиме реального времени и для больших данных требуют больших вычислительных мощностей. Могут использоваться для проверки каких-либо гипотез или оценки результата работоспособности созданных кодов.

Анализ работоспособности алгоритма

Любой созданный подход, использующий методы ML, может оцениваться с двух сторон:

1. Параметры алгоритма, влияющие на скорость работы всей системы. К ним можно отнести:

- среднее количество пакетов потока, необходимое для его классификации;
- среднее время существования потока, необходимое для его классификации;
- скорость построения модели;
- минимальное количество потоков одного класса, необходимое для построения модели;
- возможность дообучения, т.е. добавления новых классов в уже существующую модель;
- сложность внедрения алгоритма в сеть.

2. Оценка результатов работы алгоритма.

Для проверки работоспособности разработанных алгоритмов с привлечением учителя первоначальную выборку делят на две части – обучающую и тестовую. На обучающей выборке строят модель для классификации, а на тестовой – проверяют ее работоспособность. Обычно размеры обучающей выборки намного превышают размеры тестовой. Зачастую стремятся к

тому, чтобы шаблоны трафика попадали в тестовую и обучающую выборки в одинаковых процентных соотношениях среди различных классов. Также могут применяться методы кросс-валидации для независимой проверки работы алгоритма.

Так же как и в других сферах применения методов ML, при классификации трафика может возникать проблема переобучения, т.е. когда модель слишком хорошо работает на обучающей выборке, но плохо на тестовой или на реальной сети. В таких случаях рекомендуют изменить настройки классификатора или же добавить дополнительные ограничения.

Для оценки результатов работы алгоритма для тестовой выборки рассчитывается ряд параметров:

– матрица несоответствий (Confusion matrix), в которой указываются ошибки классификатора 1-го и 2-го рода;

– доля правильных ответов (Accuracy) – доля шаблонов трафика, классифицированных верно;

– полнота (Recall) – доля шаблонов класса от общего числа классов;

– точность (Precision) – доля шаблонов класса от числа шаблонов, отмеченных алгоритмом;

– F1-мера – среднее гармоническое между полнотой и точностью.

5. Обзор перспективных исследований по классификации потоков трафика для поддержания QoS методами ML в SDN-сетях

В приложении 1 приведена таблица с анализом 20 наиболее перспективных исследований, проводимых в России и за рубежом, по классификации потоков трафика для поддержания QoS методами ML в SDN-сетях. Статьи опубликованы в период 2012–2020 и индексируются в международных наукометрических базах данных, таких как WoS, Scopus, Google Scholar и т.д.

Таблица подробно описывает, какие методы и инструменты используют авторы исследований. В описании, как и в самой статье, приводится характеристика работ с трех позиций:

1. Формирование базы данных: способ создания базы данных, ее состав, способы сбора статистических данных и методы их маркировки.

2. Формирование матрицы признаков: какие признаки являются наиболее важными по мнению авторов и что именно в работе подразумевается под классами.

3. Классификация трафика: методы классификации, включающие в себя методы ML и дополнительные способы, используемые в работе исследователями. Несмотря на наличие в каждой из работ оценки результатов классификации, в сравнительную таблицу они не вынесены, т.к. авторы данной статьи считают некорректным сравнение результатов, полученных на различных статистических наборах данных и в различных условиях. Матрица несоответствий, точность классификации и т.д. являются хорошими показателями для сравнения алгоритмов и подходов, выполненных в рамках одной работы или одних и тех же условиях, но т.к. большинство исследований по ряду причин невозможно в точности повторить, подтвердить или опровергнуть, то они не представляют научного интереса при обзоре статей. Можно лишь отметить, что во всех перечисленных работах результаты классификации оказались очень точными и высокими.

При моделировании SDN-сетей чаще всего применяют сетевой эмулятор Mininet, используя в качестве контроллера ODL, ONOS или Floodlight. В [22] HP VAN SDN-контроллер установили на реальной сети, а в [26, 60] для сбора информации о потоках трафика был перепрограммирован непосредственно сам алгоритм обработки пакетов в P4-коммуаторах.

Как показано в таблице, во многих случаях авторы статей используют уже готовый набор данных, выложенный в интернете или снятый на реальной сети, и снимают с себя задачи мониторинга потоков и их маркировки, либо используют готовые инструменты типа Wireshark, tcpdump, tcptrace. В большинстве случаев даже не даются какие-либо данные о времени построения модели или минимальном количестве пакетов потока, необходимых для его класси-

фикации. При этом делаются лишь предположения о работе трафика в режиме реального времени, которые не подтверждаются никакими исследованиями, при этом принимаются некоторые допущения. Например, в качестве рассматриваемых параметров принимается длительность всего потока, время окончания сессии и т.д., не учитывая, что классификация выбранного потока для поддержания QoS совершенно точно теряет свою актуальность, как только такой поток завершается.

Кроме того, в большинстве статей процессом маркировки трафика пренебрегают, и результатами работы становятся в основном укрупненные категории трафика, полученные с помощью автоматических средств разметки потоков. Это не дает возможности небольшим категориям или приложениям трафика классифицироваться правильно.

Помимо прочего, работы в основном сосредоточены на классификации web-приложений и приложений передачи трафика, и можно наблюдать очень небольшое количество статей, посвященных классификации голосового трафика, смоделированного в собственной сети. Между тем именно голосовой трафик является наиболее чувствительным к задержкам и джиттеру в сети.

Таким образом, большое количество работ в области классификации трафика методами ML с поддержкой параметров QoS показывают высокую степень заинтересованности научного сообщества в разработке данного исследовательского направления и подтверждают актуальность данной темы. Тем не менее, несмотря на большой проделанный объем работы, в этой области остается еще очень много нерешенных вопросов и неучтенных моментов, которые призывают ученых всего мира к дальнейшей разработке данной темы.

6. Заключение

В статье впервые систематизируются существующие подходы к классификации потоков трафика с поддержкой QoS в SDN-сетях методами ML, анализируются общие проблемы, с которыми сталкиваются исследователи на разных этапах разработки новых алгоритмов. Показаны особенности задачи, не позволяющие целиком и полностью применять методики, разработанные в других направлениях классификации. Также приводится сравнительный анализ 20 наиболее перспективных работ, выполненных по теме исследования.

На основании проведенного обзора можно сделать следующие выводы:

– для классификации трафика методами ML в SDN-сетях было создано множество различных методов, отличающихся друг от друга сложностью реализации, скоростью работы алгоритма, точностью полученных результатов и областью применения;

– в любой работе по классификации трафика можно выделить три этапа: формирование базы данных, формирование матрицы признаков и определение классов с помощью алгоритмов ML. Формирование базы данных, включающее в себя сбор статистической информации о трафике, создание и разметку базы данных, является самым сложным этапом, в особенности, маркировка мелкогранулярного трафика, такого как отдельный вид сервиса конкретного приложения;

– наиболее простым оказалось организовать сбор трафика web-приложений, передачи данных и т.д., а сложнее всего – сбор голосового трафика;

– отсутствие уникальных доступных баз данных трафика значительно замедляет темпы развития исследований по данной теме. Существующие базы данных содержат множество недостатков, вследствие чего область их применения ограничена;

– основная нерешенная проблема маркировки трафика – «ground truth problem», т.е. проблема качества используемого эталона;

– методы машинного обучения для классификации трафика работают довольно успешно. В большинстве работ наилучшие результаты показывает алгоритм случайного леса (Random Forest);

– в большинстве применяемых методов полученные модели классификации не имеют возможности добавления новых классов в режиме реального времени.

Приложение 1

Таблица 1. Сравнительная характеристика основных работ по классификации потоков трафика для поддержания QoS методами ML в SDN-сетях

№	Статья	Формирование базы данных	Формирование матрицы признаков: признаки и классы	Методы классификации
1	[26]	Генератор D-ITG, всего 1000 потоков. Мониторинг собственным методом [85]. Маркировка DSCP	По каждому из первых 10 пакетов: длина пакета, время между приходом каждого из двух пакетов из одного потока, минимальные, максимальные, средние и суммарные параметры; скорость поступления пакетов, 5 приложений	DT, Наивный Байес, SVM
2	[27]	Базы реального трафика исследовательской группы BRG in UPC, 3377 потоков. Маркировка DPI	Параметр Херста и среднее значение длины пакетов, поступающих в одно и в другое направление, порт источника и назначения, медиана длины пакетов от источника к назначению, минимальная длина пакета от назначения к источнику, количество пакетов и т.д., 6 категорий	SVM, K-means
3	[28]	Реальный трафик с индивидуального ПК, 2000 потоков, сбор трафика Wireshark	Минимальное, максимальное, среднее значение, дисперсия и общее число пакетов, среднее число пакетов в секунду, размер пакета, продолжительность, 8 категорий	MLP, RBF, C4.5, Наивный Байес, Belief Network
4	[23]	Реальный трафик с индивидуального ПК, мониторинг libpcap++	4 категории	Наивный Баейс, DT, LDA, NN, SVM
5	[29]	Реальный трафик со смартфонов, маркировка специальным приложением на Python	IP назначения, порт назначения, порт источника, СКО, дисперсия, максимум и медиана длины пакетов от источника к назначению и в обратном направлении, размеры окон в пакетах SYN, отношение количества ACK-PSH к общему количеству TCP-флагов, общее число пакетов, классификация приложений	RF совместно с DPI и классификацией по портам
6	[30]	Открытые базы данных (Wireshark, Tstat)	Направление, протокол, количество пакетов, длительность потока, среднее значение, СКО размера пакетов, скорость передачи пакетов, минимальный и максимальный размер пакетов, 2 категории	н/д

№	Статья	Формирование базы данных	Формирование матрицы признаков: признаки и классы	Методы классификации
7	[31]	Генерация трафика с помощью приложений, 6525 потоков, маркировка на запущенных процессах	142 признака: количество пакетов, размеры пакетов, время передачи, время между прибытием пакетов, направление передачи, пропускная способность, размеры окон и статистические данные (итого, минимум, максимум, медиана, среднее, дисперсия и соотношение обоих направлений) и т.д., классификация приложений	APPR [32] совместно с информацией, полученной с помощью DNS
8	[32]	Реальный трафик в сети университета и имитация работы пользовательских приложений, разметка через MSN и DNS	Размер передаваемого трафика прикладного уровня, пропускная способность, время прибытия и время отклика на разных раундах установления соединения (протокол MSN), номер порта сервера, тип протокола (TCP или UDP), флаг первого отправителя данных (клиент или сервер) и информация о размере, классификация 9 категорий	C4.5, PART, Наивный Байес, zeroR и oneR
9	[33]	Реальный трафик с 4-х индивидуальных ПК, маркировка за счет физической изоляции, мониторинг через FlowStat	Тип протокола, порт назначения, время сессии, количество пакетов: всего, отправлено, получено, отношение количества отправленных к количеству полученных, пропускная способность: пиковая, средняя, отклонения, классификация 4 категорий	J4.8, PART, RIPPER, Наивный Байес
10	[34]	Реальный трафик с индивидуального ПК, 60 потоков, мониторинг: Wireshark	Отношение количества полученных и переданных данных, энтропия информации о размере пакетов переданных и скорость передачи пакетов, классификация по 5 играм	SVM
11	[35]	Реальный трафик сайта Esnet, мониторинг: Netflow	Размер (в байтах) передачи данных и длительность потока, классификация на Elephant и Mice	GMM / EM
12	[22]	Реальный трафик с производственной сети OpenFlow. Сбор собственным методом, физическая изоляция на порту	Длительность потока, количество переданных пакетов и байт, классификация по приложениям	RF, SGB и EGB
13	[36]	Открытый набор данных Мура, сбор и маркировка через tcptrace	Длительность потока, время поступления пакета и количество пакетов в потоке, классификация по 7 категориям	Глубокое обучение и SVM
14	[37]	Отрытый набор данных ISCX VPN-nonVPN [58]	12 категорий (VPN/nonVPN)	ANN, CNN

Таблица 1. Сравнительная характеристика основных работ по классификации потоков трафика для поддержания QoS методами ML в SDN-сетях (продолжение)

№	Статья	Формирование базы данных	Формирование матрицы признаков: признаки и классы	Методы классификации
15	[38]	Реальный трафик сети Университета Каруни, мониторинг Tcpdump, 3500 потоков	Количество пакетов, размер пакета, время между поступлениями, длительность потока, процент IP-пакетов с определенными размерами и процент потоков с определенными размерами и длительностью пакетов, порты уровня транспортного протокола, размеры окон TCP, классификация по 7 категориям	C4.5, Наивный Байес, k-NN, RFB, сети Байеса
16	[39]	Открытый набор данных Мура, сбор и маркировка через tcptrace	Порт сервера, минимальный и максимальный размер TCP-сегмента, отправленного от клиента к серверу, размер окна, количество пакетов с флагом PUSH, классификация по 7 категориям	SVM и k-means кластеризация
17	[40]	Реальный трафик, воспроизводимый на лабораторной установке с помощью tcpreply	нет данных	глубокое обучение
18	[41]	Трафик, полученный на тестовом стенде, мониторинг инструментами Floodlight, маркировка ручная	Протокол транспортного уровня, порты источника и назначения, среднее число пакетов в потоке, средняя пропускная способность в пакетах/с, байт/с, классификация по 5 категориям	NN с feedforward, MLP, NARX Наивный Байес
19	[42]	Реальный трафик с индивидуального ПК, маркировка DPI	Интервал между временем поступления пакетов, длина пакета, номер порта источника, транспортный протокол, параметр Херста, классификация по 4 категориям	DPI и стеккинг SVM, Наивный Байес и K-NN
20	[43]	Реальный трафик с индивидуального ПК, маркировка ручная и tcpdump	Минимальное, максимальное, среднее значение, СКО для размеров окон, классификация по 4 приложениям	J.48, SVM+SVO, RF, Наивный Байес, Ibk

Литература

1. *Гетьман А. И., Маркин Ю. В., Евстропов Е. Ф., Обыденков Д. О.* Обзор задач и методов их решения в области классификации сетевого трафика // Труды ИСП РАН. 2017. Т. 29, В. 3. С. 117–150. DOI: 10.15514/ISPRAS-2017-29(3)-8.
2. *Boutaba R., Salahuddin M. A., Limam N., Ayoubi S., Shahriar N., Solano F. E., Rendon O. M.* A comprehensive survey on machine learning for networking: evolution, applications and research opportunities // Journal of Internet Services and Applications. 2018. № 9. P. 1–99. https://doi.org/10.1186/s13174-018-0087-2.
3. *Harkut Dr Dinesh.* An Overview of Network Traffic Classification Methods. 2015.
4. *Шелухин О. И., Ерохин С. Д., Ванюшина А. В.* Классификация IP-трафика методами машинного обучения / под ред. О. И. Шелухина. М.: Горячая линия – Телеком, 2018. 282 с.
5. *Xie J. et al.* A Survey of Machine Learning Techniques Applied to Software Defined Networking (SDN): Research Issues and Challenges // IEEE Communications Surveys & Tutorials. 2018. P. 393–430. https://doi.org/10.1109/comst.2018.2866942.
6. *Zhao Y., Li Y., Zhang X., Geng G., Zhang W. and Sun Y.* A Survey of Networking Applications Applying the Software Defined Networking Concept Based on Machine Learning // IEEE Access. 2019. V. 7. P. 95397–95417. https://doi.org/10.1109/ACCESS.2019.2928564.
7. *Mohammed A. R., Mohammed S. A. and Shirmohammadi S.* Machine Learning and Deep Learning Based Traffic Classification and Prediction in Software Defined Networking // Proc. 2019 IEEE International Symposium on Measurements & Networking (M&N). P. 1–6. https://doi.org/10.1109/IWMN.2019.8805044.
8. Mininet: An Instant Virtual Network on your Laptop (or other PC). URL: http://mininet.org/ (дата обращения: 01.02.2020).
9. *Singhal P., Mathur R., Vyas H.* State of the Art Review of Network Traffic Classification based on Machine Learning Approach // Proc. International Conference on Recent Trends in Engineering & Technology, 2013. P. 12–15
10. *Nguyen T., Grenville A.* A survey of techniques for internet traffic classification using machine learning // IEEE Communications Surveys and Tutorials. 2008. V. 10. P. 56–76.
11. *Patcha A. and Park J.-M.* An overview of anomaly detection techniques: Existing solutions and latest technological trends // Computer Networks. 2007. V. 51, № 12. P. 3448–3470.
12. *Latah M., Toker L.* Artificial Intelligence Enabled Software Defined Networking: A Comprehensive Overview // IET Networks. 2018. V. 8. https://doi.org/10.1049/iet-net.2018.5082.
13. *Buczak A. L. and Guven E.* A survey of data mining and machine learning methods for cyber security intrusion detection // IEEE Communications Surveys & Tutorials. 2016. V. 18, № 2. P. 1153–1176.
14. *Hodo E., Bellekens X. J., Hamilton A. W., Tachtatzis C., and Atkinson R. C.* Shallow and Deep Networks Intrusion Detection System: A Taxonomy and Survey // ArXiv. 2017. abs/1701.02145.
15. *Herrera A. J., Camargo J. E.* A survey on machine learning applications for software defined network security // LNCS 11605. 2019. P. 70–96
16. *Zhou X., Sun M., Li G. Y., and Juang B.-H.* Machine learning and cognitive technology for intelligent wireless networks // ArXiv. 2017. abs/1710.11240.
17. *Chen M., Challita U., Saad W., Yin C., and Debbah M.* Machine learning for wireless networks with artificial intelligence: A tutorial on neural networks // arXiv preprint arXiv:1710.02913, 2017.
18. *Wang X., Li X., and Leung V. C. M.* Artificial intelligence-based techniques for emerging heterogeneous network: State of the arts, opportunities, and challenges // IEEE Access. 2015. V. 3. P. 1379–1391.
19. *Klaine P. V., Imran M. A., Onireti O., and Souza R. D.* A survey of machine learning techniques applied to self-organizing cellular networks // IEEE Communications Surveys & Tutorials. 2017. № 99. P. 1–1.

20. *Haleplidis E., Pentikousis K., Denazis S. et al.* Software-Defined Networking (SDN): Layers and Architecture Terminology, 2015. URL: https://tools.ietf.org/html/rfc7426 (дата обращения: 01.02.2020).
21. *MVB da Silva, AS Jacobs, RJ Pfitscher, LZ Granville.* IDEAFIX: Identifying elephant flows in P4-based IXP networks // Proc. 2018 IEEE Global Communications Conference (GLOBECOM). P. 1–6.
22. *Ghulam Mohi-Ud-Din, Liu Zhi Qiang, Zhang Jiangbin.* Data learning and traffic classification by Machine Learning // Proc. Academicsera 18th International Conference, Sydney, Australia, 2018.
23. *Gomes R.L., Madeira M. E. R.*: A traffic classification agent for virtual networks based on QoS classes // IEEE Latin Am. Trans. 2012. V. 10, № 3. P. 1734–1741.
24. *Troia S., Martin N., Rodriguez A., Hernandez J. A., et al.* Machine-learning-assisted routing in SDN-based optical networks // Proc. 44th European Conference on Optical Communication (ECOC), Rome, September 2018. https://doi.org/10.1109/ECOC.2018.8535437.
25. *Pham Q.T., Hadjadj-Aoul Y., Outtagarts A.* Deep Reinforcement Learning based QoS-aware Routing in Knowledge-defined networking // Proc. 14th EAI International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (Qshine), Dec 2018, Ho Chi Minh City, Vietnam. P. 1–13.
26. *Маньков В. А., Краснова И. А.* Классификация потоков трафика SDN-сетей методами машинного обучения в режиме реального времени» // Труды международной научно-технической конференции «Информационные технологии и математическое моделирование систем», 2019. С. 65–68. https://doi.org/10.36581/CITP.2019.31.51.016.
27. *Lin S. Wang, Luo Min.* A Framework for QoS-aware Traffic Classification Using Semi-supervised Machine Learning in SDNs // Proc. 2016 IEEE International Conference on Services Computing (SCC). P. 760–765. https://doi.org/10.1109/SCC.2016.
28. *Singh K. Agrawal, S. Sohi B.* A Near Real-time IP Traffic Classification Using Machine Learning. International Journal of Intelligent Systems and Applications. 2013. V. 5. P. 83–93. https://doi.org/10.5815/ijisa.2013.03.09.
29. *Iwai T., and Nakao A.* Adaptive mobile application identification through in-network machine learning // Proc. 18th Asia-Pacific Network Operations and Management Symposium (APNOMS), 2016. P. 1–6. https://doi.org/10.1109/APNOMS.2016.7737226.
30. *Saqib N. A., Shakeel Y., Khan M. A., Mehmood H., Zia M.* An effective empirical approach to VoIP traffic classification // Turkish Journal of Electrical Engineering & Computer Sciences. 2017. V. 25, № 2. P. 888–900.
31. *Huang N., Li C., Li C., Chen C., Chen C., and Hsu I.* Application identification system for SDN QoS based on machine learning and DNS responses // Proc. 19th Asia-Pacific Network Operations and Management Symposium (APNOMS), 2017. P. 407–410. https://doi.org/10.1109/APNOMS.2017.8094160.
32. *Huang N., Jai G., Chao H., Tzang Y., and Chang H.* Application traffic classification at the early stage by characterizing application rounds // Inf. Sci. 2013. V. 232. P. 130–142. https://doi.org/10.1016/j.ins.2012.12.039.
33. *Anantavrasilp I., and Scholer T.* Automatic flow classification using machine learning // Proc. 15th International Conference on Software, Telecommunications and Computer Networks, 2007. P. 1–6. https://doi.org/10.1109/SOFTCOM.2007.4446129.
34. *Dong Y., Zhang M., Zhou R.* Classification of Network Game Traffic Using Machine Learning. In: Yuan H., Geng J., Liu C., Bian F., Surapunt T. (eds) Geo-Spatial Knowledge and Intelligence. GSKI 2017. Communications in Computer and Information Science. V. 848. Springer, Singapore. https://doi.org/10.1007/978-981-13-0893-2_15.
35. *Chhabra A., Kiran M.* Classifying Elephant and Mice Flows, in High-Speed Scientific Networks // Proc. 4th International Workshop on Innovating the Network for Data Intensive Science (INDIS), 2017.

36. *Zhang C. Wang, X. Li F. He, Q. Huang Min.* Deep learning-based network application classification for SDN // Transactions on Emerging Telecommunications Technologies. 2017. V. 29. https://doi.org/10.1002/ett.3302.
37. *Lotfollahi M., Jafari Siavoshani M., Shirali Hossein Zade R. et al.* Deep Packet: A Novel Approach For Encrypted Traffic Classifcation Using Deep Learning // Soft Computing. 2020. V. 24. https://doi.org/10.1007/s00500-019-04030-2.
38. *Jamuna A., Vinodh Ewards S. E.* Efficient Flow based Network Traffic Classification using Machine Learning // International Journal of Engineering Research and Applications (IJERA). 2013. V. 3, Is. 2. P. 1324–1328.
39. *Fan Z., Liu R.* Investigation of machine learning based network traffic classification // Proc. International Symposium on Wireless Communication Systems (ISWCS), 2017. P. 1–6. https://doi.org/10.1109/ISWCS.2017.8108090.
40. *Mestres A., Rodríguez-Natal A., Carner J., Barlet-Ros P., Alarcón E., Solé M., et al.* Knowledge-Defined Networking // ArXiv. 2017. abs/1606.06222.
41. *Parsaei M. R., Sobouti M. J., and Javidan R.* Network Traffic Classification using Machine Learning Techniques over Software Defined Networks // International Journal of Advanced Computer Science and Applications. 2017. V. 8.
42. *Changhe Y., Lan J., Xie J., Hu Y.* QoS-aware traffic classification architecture using machine learning and deep packet inspection in SDNs // Proc. International congress of information and communication technology, 2018. P. 1209–1216.
43. *Middleton S., Modafferi S.* Scalable Classification of QoS for Real-Time Interactive Applications from IP Traffic Measurements // Computer Networks. 2016. V. 107. P. 121–132.
44. *Alharbi F*. SDN-based mechanisms for provisioning quality of service to selected network flows // Theses and Dissertations: Computer Science. 2018. V. 72.
45. *Aroussi S., and Mellouk A.* Survey on machine learning-based QoE-QoS correlation models // Proc. International Conference on Computing, Management and Telecommunications (ComManTel), 2014. P. 200–204.
46. *Alharbi F.* SDN – Survey on Machine Learning-based QoE-QoS Correlation Models. An adaptive machine learning-based QoE approach in SDN context for video-streaming services // Theses and Dissertations: Computer Science. 2018. V. 72.
47. *Caicedo O. M.* Evalution de QoS usando tecnicas de machine learning. Universidad del Cauca, 2019. P. 19.
48. *Letaifa A. B., Maher G., and Mouna S.* ML based QoE enhancement in SDN context: Video streaming case // Proc. 13th International Wireless Communications and Mobile Computing Conference (IWCMC), 2017. P. 103–108.
49. *Raumer D., Schwaighofer L., and Carle G.* MonSamp: A distributed SDN application for QoS monitoring // Proc. Federated Conference on Computer Science and Information Systems, 2014. P. 961–968.
50. *Huang N., Liao I., Liu H., Wu S., and Chou C.* A dynamic QoS management system with flow classification platform for software-defined networks // Proc. 8th International Conference on Ubi-Media Computing (UMEDIA), 2015.
51. *Sapio A., Canini M., Ho C., Nelson J., Kalnis P., Kim C., Krishnamurthy A., Moshref M., et al.* Scaling Distributed Machine Learning with In-Network Aggregation // ArXiv, 2019. abs/1903.06701.
52. *Zhang J., Chen C., Xiang Y., Zhou W., and Vasilakos, A.V.* (2013). An Effective Network Traffic Classification Method with Unknown Flow Detection // IEEE Transactions on Network and Service Management. 2013. V. 10. P. 133–147.
53. *Bakker J. N., Ng B., Seah W. K., and Pekár A.* Traffic Classification with Machine Learning in a Live Network // Proc. IFIP/IEEE Symposium on Integrated Network and Service Management (IM), 2019. P. 488–493.
54. CAIDA Data – Overview of Datasets, Monitors, and Reports. URL: https://www.caida.org/data/overview/ (дата обращения: 01.02.2020).

55. SecRepo.com – Samples of Security Related Data. URL: http://www.secrepo.com/ (дата обращения: 01.02.2020).
56. ISCX Information Centre of Excellence for Tech Innovation. URL: http://www.iscx.ca/datasets/ (дата обращения: 01.02.2020).
57. Машинное обучение вместо DPI. Строим классификатор трафика URL: https://habr.com/ru/post/304926/ (дата обращения: 01.02.2020).
58. *Маньков В. А., Краснова И. А.* Алгоритм динамической классификации потоков в мультисервисной SDN-сети // T-Comm: Телекоммуникации и транспорт. 2017. Т. 11, № 12. С. 37–42.
59. *Маньков В. А., Краснова И. А.* Задача управления трафиком с динамическим определением QoS в мультисервисных SDN сетях // Сборник трудов XI Международной отраслевой научно-технической конференции «Технологии информационного общества», 15–16 марта 2017 г., МТУСИ. С. 67–68.
60. *Mankov V. A., Krasnova I. A.* Collection of Individual Packet Statistical Information in a Flow Based on P4-switch // Proc. Advances in Intelligent Systems and Computing. 2020. V. 1127. https://doi.org/10.1007/978-3-030-39216-1_11.
61. Машинное обучение для людей: Разбираемся простыми словами. URL: https://vas3k.ru/blog/machine_learning/ (дата обращения: 01.02.2020).

Статья поступила в редакцию 07.05.2020;
переработанный вариант – 20.08.2020.

Деарт Владимир Юрьевич
к.т.н., профессор кафедры сетей связи и систем коммутации МТУСИ (111024, Москва, ул. Авиамоторная, 8а), e-mail: vdeart@mail.ru.

Маньков Владимир Александрович
ведущий преподаватель учебного центра Нокиа (111024, Москва, ул. Авиамоторная, 8а), e-mail: vladimir.mankov@gmail.com.

Краснова Ирина Артуровна
аспирант кафедры сетей связи и систем коммутации МТУСИ (111024, Москва, ул. Авиамоторная, 8а), e-mail: irina_krasnova-angel@mail.ru.

Analysis of promising approaches and research on traffic flow classification for maintaining QoS by ML methods in SDN networks

V. Yu. Deart, V. A. Mankov, I. A. Krasnova

One of the most important tasks that exist in modern networks is to maintain the Quality-of-Service QoS at the appropriate level which can be achieved by applying various traffic management mechanisms. In order to maintain the QoS parameters in the proper state, you need to know the types of traffic passing through the network. Given high-tech and high-performance networks such as SDN networks, traffic classification by conventional methods becomes almost impossible. Data mining methods, including Machine Learning methods, come to the rescue.
The article analyzes the main promising approaches to real-time traffic classification for maintaining QoS in SDN networks by ML methods as well as provides a comparative overview of the most outstanding works in this field.

Keywords: Machine Learning (ML), SDN, QoS, traffic classification.