Научная статья
УДК 519.872
https://doi.org/10.31854/1813-324X-2025-11-5-9-20
EDN:WALXIJ

Обнаружение признаков аномального поведения трафика на основе методов искусственного интеллекта

Михаил Васильевич Близнюк[1], mikebliznyuk200123@gmail.com
Василий Иванович Близнюк[2], v_bliznyuk@mail.ru
Андрей Петрович Постарнак[3] ✉, postarnak.ap@sut.ru
Александр Владичевич Болбенков[2], bolben@mail.ru
Антон Юрьевич Кибалин[2], kibalinanton@mail.ru

[1]Федеральная службы охраны Российской Федерации в Северо-Западном Федеральном округе, Санкт-Петербург, 191123, Российская Федерация
[2]Академия Федеральной службы охраны Российской Федерации, Орел, 302034, Российская Федерация
[3]Санкт-Петербургский государственный университет телекоммуникаций им. проф. М.А. Бонч-Бруевича, Санкт-Петербург, 193232, Российская Федерация

Аннотация

Актуальность: *в настоящее время для обнаружения признаков аномального поведения трафика применяется сигнатурный анализ, однако данный метод имеет свои ограничения. Учитывая недостатки сигнатурного анализа, становится ясным, что использование только этого метода может ограничить возможности обнаружения и предотвращения новых и неизвестных аномалий. Рассмотрено внедрение поведенческого анализа в дополнение к сигнатурному, чтобы обеспечить более полную и надежную защиту информационной системы.*
Цель исследования: *повышение оперативности обнаружения признаков аномального поведения трафика за счет использования методов искусственного интеллекта.*
Используемые методы: *для решения задачи обнаружения аномального поведения трафика без обучения на основании анализа принято решение о комбинировании алгоритма k-ближайших соседей и метода главных компонент.*
Результаты: *разработаны алгоритм обнаружения сетевых аномалий, программное средство «Обнаружение сетевых аномалий на основе методов искусственного интеллекта», программный стенд.*
Новизна исследования *заключается в том, что программное средство позволяет вычислять критерии обнаружения аномалий сетевого трафика за период времени меньший, чем у ранее представленных аналогов, и позволяет обнаруживать различные аномалии без предварительного обучения на готовых шаблонах аномалий.*
Практическая значимость: *полученные в работе результаты могут быть использованы для классификации аномалий сетевого трафика в информационных системах и инфраструктурах.*

Ключевые слова: *сетевые аномалии, методы искусственного интеллекта, поведенческий анализ*

Ссылка для цитирования: Близнюк М.В., Близнюк В.И., Постарнак А.П., Болбенков А.В., Кибалин А.Ю. Обнаружение признаков аномального поведения трафика на основе методов искусственного интеллекта // Труды учебных заведений связи. 2025. Т. 11. № 5. С. 9–20. DOI:10.31854/1813-324X-2025-11-5-9-20. EDN:WALXIJ

Original research
https://doi.org/10.31854/1813-324X-2025-11-5-9-20
EDN:WALXIJ

Artificial Intelligence-Based Traffic Anomaly Detection

Mikhail V. Bliznyuk[1], mikebliznyuk200123@gmail.com
Vasiliy I. Bliznyuk[2], v_bliznyuk@mail.ru
Andrey P. Postarnak[3] ✉, postarnak.ap@sut.ru
Alexandr V. Bolbenkov[2], bolben@mail.ru
Anton Yu. Kibalin[2], kibalinanton@mail.ru

[1]Federal Security Service of the Russian Federation in the North-Western Federal District, St. Petersburg, 191123, Russian Federation
[2]Academy of the Federal Guard Service of the Russian Federation, Orel, 302034, Russian Federation
[3]The Bonch-Bruevich Saint Petersburg State University of Telecommunications, St. Petersburg, 193232, Russian Federation

Annotation

Relevant. *Nowadays to detect signs of abnormal traffic behavior signature analysis is used, but this method has its limitations. Given the disadvantages of signature analysis, it becomes clear that using this method alone can limit the ability to detect and prevent new and unknown anomalies. Considered implementation of a custom analysis in addition to the signature to provide a more complete and reliable information system protection.*
The aim of the study *is to increase the efficiency of detecting signs of abnormal traffic behavior through the use of artificial intelligence methods.*
In result *the following were developed: an algorithm for detecting network anomalies, a software tool "Detection of network anomalies based on methods of artificial intelligence", a software stand.*
The novelty *of the study lies in the fact that the software allows you to calculate the criteria for detecting anomalies of network traffic in a period of time shorter than that of previously presented analogs and allows you to detect various anomalies without prior training on ready-made anomaly templates.*
The practical significance. *The results obtained in the work can be used for classification of anomalies of network traffic in information systems and infrastructures.*

Keywords: *network anomalies, artificial intelligence methods, behavioral analysis*

For citation: Bliznyuk M.V., Bliznyuk V.I., Postarnak A.P., Bolbenkov A.V., Kibalin A.Yu. Artificial Intelligence-Based Traffic Anomaly Detection. *Proceedings of Telecommunication Universities.* 2025;11(5):9–20. (in Russ.) DOI:10.31854/1813-324X-2025-11-5-9-20. EDN:WALXIJ

Введение

Современные информационные системы и коммутационные инфраструктуры сталкиваются с растущими угрозами, что требует эффективных методов их обнаружения и предотвращения [1–3]. В условиях стремительного роста числа угроз традиционные способы защиты часто оказываются недостаточными. Одним из таких методов является сигнатурный анализ, который основан на поиске известных шаблонов аномалий в сетевом трафике [3, 4]. Сигнатурный анализ работает путем сравнения входящего трафика с базой данных известных сигнатур, представляющих собой цифровые «отпечатки» ранее идентифицированных угроз [5]. Этот метод позволяет быстро и эффективно обнаруживать аномалии, которые уже зарегистрированы и классифицированы в банке данных угроз безопасности информации [6].

Однако сигнатурный анализ имеет значительные ограничения [7]. Во-первых, он не способен выявлять новые, ранее неизвестные угрозы. Появление новых видов угроз требует обновления базы

сигнатур, что создает временны́е промежутки, в течение которых системы остаются уязвимыми. Во-вторых, этот метод плохо справляется с полиморфными угрозами, которые могут изменять свои сигнатуры с целью обхода системы защиты. В-третьих, сигнатурный анализ часто требует значительных ресурсов для поддержания и обновления базы данных, что может быть проблематично для больших и динамичных сетей.

В данной статье рассматривается альтернатива и дополнение к сигнатурному анализу – поведенческий анализ. Он основан на мониторинге и изучении нормального поведения системы и выявление отклонений от этого поведения, которые могут свидетельствовать о наличии аномалий [8]. Этот подход позволяет обнаруживать подозрительную активность, даже если конкретная аномалия ранее не была зарегистрирована. Поведенческий анализ оценивает различные параметры, такие как объем трафика, типы запросов, временны́е метки и другие метрики, чтобы определить аномалии.

Одним из ключевых преимуществ поведенческого анализа является его способность функционировать без предварительного обучения. В то время, как многие современные системы обнаружения аномалий требуют предварительного обучения на больших объемах данных для создания моделей нормального поведения, поведенческий анализ может адаптироваться к новым и неизвестным угрозам в реальном времени (https://bdu.fstec.ru/threat). Это достигается за счет использования адаптивных алгоритмов, которые могут автоматически подстраиваться под изменяющиеся условия и идентифицировать аномалии на основе текущих наблюдений.

Таким образом, поведенческий анализ может значительно улучшить обнаружение новых аномалий, дополняя сигнатурный анализ и обеспечивая более высокий уровень защиты информационных систем. В условиях постоянного появления новых угроз и усложнения действий нарушителя, комбинированное использование сигнатурного и поведенческого анализов представляется наиболее эффективным подходом для обеспечения комплексной защиты информационных систем и коммутационных инфраструктур.

Анализ методов искусственного интеллекта

Анализ методов искусственного интеллекта включает в себя рассмотрение различных подходов и техник, используемых для создания систем, способных имитировать интеллектуальные способности человека.

Вот несколько основных методов искусственного интеллекта:

1) глубокое обучение (*от англ.* Deep Learning);

2) эволюционные алгоритмы (*от англ.* Evolutionary Algorithms);

3) логическое программирование (*от англ.* Logic Programming);

4) символьные методы (Symbolic AI);

5) обработка естественного языка (NLP, *аббр. от англ.* Natural Language Processing);

6) обучение с подкреплением (*от англ.* Reinforcement Learning);

7) системы экспертных знаний (*от англ.* Expert Systems);

8) машинное обучение (*от англ.* Machine Learning).

Глубокое обучение – подраздел машинного обучения, использующий искусственные нейронные сети с большим числом слоев для извлечения сложных закономерностей из данных. Глубокое обучение получило широкое распространение благодаря своей способности эффективно обрабатывать большие объемы данных и решать сложные задачи, такие как распознавание образов, естественный язык, аудиообработка и другие.

Ключевые аспекты глубокого обучения:

– искусственные нейронные сети (ANN, *аббр. от англ.* Artificial Neural Networks): основной строительный блок глубокого обучения; ANN моделируют структуру нейронной сети мозга и состоят из множества взаимосвязанных узлов, называемых нейронами; каждый нейрон принимает входные данные, выполняет вычисления и передает результаты следующему слою;

– глубокие нейронные сети (DNN, *аббр. от англ.* Deep Neural Networks) содержат несколько скрытых слоев между входными и выходными слоями; глубокие сети способны автоматически извлекать иерархии признаков из данных на различных уровнях абстракции;

– сверточные нейронные сети (CNN, *аббр. от англ.* Convolutional Neural Networks) предназначены для обработки структурированных данных, таких как изображения; CNN имеют специальные слои свертки и пулинга, которые позволяют им извлекать пространственные закономерности изображений;

– рекуррентные нейронные сети (RNN, *аббр. от англ.* Recurrent Neural Networks) обрабатывают последовательные данные, сохраняя состояние или память о предыдущих входах; RNN широко используются в задачах обработки естественного языка, временны́х рядов и других последовательностей данных;

– генеративные модели (*от англ.* Generative Models) позволяют создавать новые данные, имитируя распределение обучающих данных; примеры включают в себя генеративные состязательные

сети (GAN, *аббр. от англ.* Generative Adversarial Networks) и вариационные автокодировщики (VAE, *аббр. от англ.* Variational Autoencoders).

К преимуществам глубокого обучения можно отнести *автоматическое извлечение признаков* (глубокие модели способны автоматически извлекать иерархии признаков из данных, что устраняет необходимость вручную создавать признаки); *способность к обобщению* (могут обучаться на больших объемах данных и обобщать на новые данные, что делает их эффективными в различных прикладных областях); *масштабируемость* (с развитием вычислительных мощностей и технологий глубокое обучение может масштабироваться для обработки больших объемов данных).

Глубокое обучение находит применение во многих областях, включая компьютерное зрение, распознавание речи, естественный язык, аудиообработку, медицинскую диагностику, финансовый анализ и другие. Его возможности и эффективность продолжают расти с развитием технологий и улучшением алгоритмов.

Эволюционные алгоритмы основаны на принципах естественного отбора и эволюции, где популяция решений подвергается итеративному процессу мутации, скрещивания и отбора для достижения оптимальных или приблизительно оптимальных решений. Это класс алгоритмов оптимизации, вдохновленных механизмами естественного отбора и эволюции в природе. Они используют метафору процессов эволюции, таких как мутация, скрещивание и отбор, для выявления оптимальных решений в пространстве поиска.

Основные компоненты и принципы работы алгоритмов:

– эволюционные алгоритмы работают с *популяцией* потенциальных решений, называемых индивидами или особями; эти особи представляют собой кандидатов на оптимальное решение задачи оптимизации;

– для оценки качества каждой особи используется *функция приспособленности* (*от англ.* Fitness Function), которая определяет, насколько хорошо каждая особь решает задачу оптимизации; в зависимости от постановки задачи значение функции приспособленности максимизируется или минимизируется;

– *генетические операторы*: мутация, *от англ.* Mutation (процесс случайного изменения генетического материала особи для создания новых вариантов); скрещивание, *от англ.* Crossover (процесс комбинирования генетического материала двух особей для создания потомства, которое обладает комбинацией их характеристик); отбор, *от англ.* Selection (процесс выбора особей для создания следующего поколения на основе их приспособленности; чем лучше приспособленность, тем больше вероятность, что эта особь будет выбрана для скрещивания и / или передачи генетического материала следующему поколению);

– *процесс эволюции*: итеративно применяются генетические операторы к текущей популяции особей, чтобы создать новое поколение; этот процесс продолжается до тех пор, пока не будет достигнуто условие остановки – достижение максимального числа поколений или достижение определенного уровня приспособленности;

– существует несколько *параметров алгоритма*, которые могут влиять на эффективность поиска решений, такие как размер популяции, вероятности мутации и скрещивания, способы отбора и т. д.; настройка этих параметров может потребоваться для достижения оптимальных результатов.

Эволюционные алгоритмы могут применяться к широкому спектру задач оптимизации, включая поиск функций, обучение нейронных сетей, проектирование архитектуры систем и другие. Они часто используются в случаях, когда функция приспособленности не дифференцируема или, когда пространство поиска слишком велико для эффективного применения классических методов оптимизации.

Логическое программирование – подход к искусственному интеллекту, где знания и правила представлены в форме логических выражений. Примером такого языка является Prolog (*от англ.* Programming in Logic), который широко используется в академических и научных сферах, а также в различных прикладных областях. Это парадигма программирования, основанная на математической и формальной логике. Основным инструментом здесь является логический язык программирования, а также система вывода на основе логических правил и фактов. Этот метод широко используется в экспертных системах. В Prolog программа состоит из фактов и правил, а система вывода использует механизм унификации и резолюции для поиска ответов на запросы.

Основные элементы логического программирования: *факты* (утверждения о мире, которые истинны по определению; факты представляются в виде предикатов, которые описывают отношения между объектами); *правила* (логические выражения, которые определяют отношения между фактами; правила состоят из головы и тела: голова содержит цель (запрос), а тело – условия, при которых цель считается истинной); *запросы* (выражения, для которых система вывода пытается найти ответы, используя имеющиеся факты и правила; запросы представляют собой цели, которые нужно достигнуть); *унификация* (процесс сопоставления переменных и термов для нахождения значений переменных, при которых выражение становится

истинным; широко используется в системах вывода логического программирования для сопоставления запросов с фактами и правилами).

Логическое программирование хорошо подходит для решения задач, в которых знания о предметной области легко выражаются в виде логических правил и фактов. Оно часто используется в областях искусственного интеллекта, а также в экспертных системах, обработке естественного языка, анализе данных и других прикладных областях, где важна логическая инференция и обработка знаний.

Символьные методы, также известные как символьный искусственный интеллект (Symbolic AI), – подход к созданию интеллектуальных систем, который сосредоточен на символах и символьных операциях и использует символьные представления знаний и операции для решения задач, в отличие от статистических методов, таких как машинное обучение и нейронные сети.

Основные характеристики символьного искусственного интеллекта:

– *символьное представление знаний* (знания о мире представляются в виде символов и символьных выражений; эти символы могут представлять объекты, отношения, действия и другие элементы предметной области);

– *манипуляция символами* (сопоставление шаблонов, унификация, модификация символьных выражений и выполнение символьных операций, таких как логические выводы).

Символьные методы часто используют *логические выводы* для генерации новых знаний на основе имеющихся фактов и правил. Логический вывод позволяет извлекать новые знания и делать заключения на основе логических правил. Примером символьного искусственного интеллекта являются *экспертные системы*, использующие символьное представление знаний и логический вывод для моделирования экспертных знаний и принятия решений в специфической предметной области.

Для реализации методов часто используют специализированные символьные языки программирования, такие как Lisp, Prolog и другие. Эти языки обеспечивают удобные средства для работы с символами и выполнения символьных операций.

Преимущества символьного искусственного интеллекта включают ясность и интерпретируемость решений, возможность использования экспертных знаний и эффективность в задачах, где символьное представление знаний является естественным и удобным. В то же время возможны ограничения в областях, где данные сложны или неточны, и где требуется обучение на больших объемах данных. В последние десятилетия символьные методы часто комбинируются с методами машинного обучения и другими подходами для создания гибридных систем искусственного интеллекта, которые объединяют преимущества различных методов.

Обработка естественного языка – область искусственного интеллекта, занимающаяся взаимодействием между компьютерами и естественными языками, используемыми людьми для коммуникации. Она включает в себя такие задачи, как распознавание речи, синтаксический анализ, семантический анализ, машинный перевод и другие. В основе этого метода положена цель – разработать методы и системы, которые могут понимать, анализировать и генерировать текст на естественных языках, таких как английский, русский, китайский и другие.

Ключевые аспекты метода:

– *токенизация и сегментация* (процесс разделения текста на более мелкие элементы, такие как слова, фразы или предложения; включает в себя разделение текста на отдельные слова или токены, а сегментация отвечает за разделение текста на более крупные единицы, такие как предложения или абзацы);

– *морфологический анализ* (процесс анализа формы слова, который включает в себя определение частей речи, склонения, спряжения и других морфологических характеристик; помогает понять структуру и смысл слова в контексте предложения);

– *синтаксический анализ* (процесс определения структуры предложения и отношений между его элементами, такими как подлежащее, сказуемое, дополнения и т. д.; помогает понять синтаксическую структуру предложения и его семантический смысл);

– *семантический анализ* (это процесс понимания смысла текста и выявления семантических отношений между словами и фразами; включает в себя определение значения слов и конструкций, а также их взаимосвязей в контексте);

– *обработка диалогов* (понимание и генерация диалогов между компьютером и пользователем; включает в себя распознавание речи, понимание интентов пользователя, генерацию ответов и поддержку естественного и интуитивного взаимодействия);

– *машинный перевод* (автоматический перевод текста с одного языка на другой; используются различные модели и методы, включая статистические подходы, нейронные сети и трансформеры, для перевода текста между языками);

– *извлечение информации и анализ текстовых данных* (извлечение структурированной информации из неструктурированных текстовых данных; может включать извлечение сущностей, связей, фактов, событий и другой информации из текста).

Обработка естественного языка находит применение во многих областях, включая поиск информации, анализ социальных медиа, автоматическую обработку документов, управление знаниями, медицинское информационное моделирование, автоматизацию процессов обслуживания клиентов и многое другое. В последние годы развитие нейронных сетей и глубокого обучения привело к значительному прогрессу в области обработки естественного языка, позволяя решать более сложные задачи с большей точностью и эффективностью.

Обучение с подкреплением – метод машинного обучения, в котором агент (программа или робот) учится принимать последовательность действий в среде с целью максимизации суммарного награждения. Агент действует в среде, где он может выполнять различные действия, наблюдать состояние среды и получать награды или штрафы в зависимости от своих действий. Цель агента состоит в том, чтобы научиться принимать оптимальные действия, которые приведут к максимизации награды в долгосрочной перспективе.

Основными компонентами метода являются среда (*от англ.* Environment), состояние (*от англ.* State), функция вознаграждения (*от англ.* Reward Function), политика (*от англ.* Policy), обучение (*от англ.* Learning).

Среда – контекст, в котором действует агент. Среда может быть реальной (например, физический робот) или виртуальной (например, симуляция игрового мира). Обычно она характеризуется состояниями, действиями, возможными наградами и правилами, определяющими, какие действия возможны в каждом состоянии.

Состояние – описание текущего положения агента в среде. Состояние может быть полным, когда оно содержит всю информацию о среде, или частичным, когда агент видит только часть информации.

Действие – выбор, который агент делает в определенном состоянии с целью изменения этого состояния и получения награды. Действия могут быть дискретными (например, направление движения) или непрерывными (например, сила двигателя).

Функция вознаграждения определяет размер вознаграждения, которое агент получает за выполнение конкретного действия в определенном состоянии. Цель агента состоит в том, чтобы максимизировать суммарное награждение в течение времени.

Политика – стратегия, которую агент использует для выбора действий в каждом состоянии. Политика может быть детерминированной (определенной) или стохастической (вероятностной), и ее цель состоит в том, чтобы максимизировать суммарное награждение.

Агент использует методы *обучения* для настройки своей стратегии на основе опыта взаимодействия со средой. Обучение может происходить путем проб и ошибок (Trial and Error), используя различные методы, такие как *Q*-обучение, методы глубокого обучения и многое другое.

Применения метода обучения с подкреплением включают в себя управление роботами, автономное вождение, управление игровыми персонажами, оптимизацию финансовых портфелей, управление энергосистемами и другие. Метод представляет собой мощный инструмент для решения задач, где требуется принятие последовательности действий для достижения определенных целей в динамической среде.

Системы экспертных знаний моделируют решение проблем в определенной области, основываясь на знаниях экспертов в этой области. Эти системы принимают данные о проблеме, анализируют их с помощью правил и знаний, заложенных в систему, и предоставляют пользователю рекомендации, советы или решения на основе своего «экспертного» знания.

Ключевые аспекты систем экспертных знаний:

– *база знаний* (Knowledge Base) – центральная часть системы, которая содержит знания и правила, определяющие специализированные знания эксперта в определенной области; знания обычно представлены в виде правил «если – то», «то» (Rule-Based) или в виде фактов и предикатов (Declarative Knowledge);

– *механизм вывода* (Inference Engine) – компонент системы, который использует базу знаний для решения конкретных проблем; применяет логические правила и алгоритмы, чтобы сформулировать заключения и принять решения на основе предоставленных данных;

– *интерфейс пользователя* (интерактивная часть системы) позволяет пользователям взаимодействовать с системой; может принимать данные от пользователя, задавать вопросы для уточнения информации и предоставлять рекомендации или выводы;

– *обучение и адаптация* (некоторые системы экспертных знаний могут включать в себя возможность обучения на основе опыта или данных, что позволяет им улучшать свою производительность и адаптироваться к изменяющимся условиям).

Применение систем экспертных знаний включает в себя широкий спектр областей, в том числе медицину, финансы, инженерные системы, управление предприятием и многое другое. Например, системы поддержки принятия решений в медицине могут использоваться для диагностики заболеваний и предоставления рекомендаций по лечению на основе клинических данных пациента и медицинских знаний.

Однако следует отметить, что эти системы имеют свои ограничения. Их применение неэффективно в областях, где знания не могут быть формализованы или изменяются слишком быстро, а также в задачах, требующих контекстного понимания или нестандартных решений. В последние годы системы экспертных знаний часто комбинируются с другими методами искусственного интеллекта, такими как машинное обучение и обработка естественного языка, чтобы создавать более гибкие и мощные системы поддержки принятия решений.

Машинное обучение позволяет компьютерным системам обучаться на основе данных и опыта, вместо явного программирования. Методы машинного обучения включают в себя такие алгоритмы как нейронные сети, деревья решений, метод опорных векторов, кластеризация и другие.

Методы искусственного интеллекта, в частности машинного обучения, в последние годы стремительно развиваются и находят применение в самых различных областях от медицины и финансов до транспорта и развлечений [9]. Машинное обучение, являясь одной из ключевых технологий искусственного интеллекта, позволяет системам самостоятельно учиться и адаптироваться на основе данных, что значительно повышает их эффективность и точность. Это делает машинное обучение незаменимым инструментом в современном мире, где количество информации растет экспоненциально, а задачи становятся все более сложными [10].

Основная цель машинного обучения – разработка алгоритмов, которые могут автоматически обнаруживать закономерности в данных и использовать их для принятия решений или предсказаний. Машинное обучение включает такие методы, как контролируемое и неконтролируемое обучение, обучение с подкреплением и другие. Каждый из этих методов имеет свои особенности, преимущества и ограничения, что делает их применение специфичным для различных типов задач и данных.

Анализ существующих методов машинного обучения для решения задачи обнаружения сетевых аномалии представлен в таблице 1. На основании проведенного анализа сделан вывод, что ни один из алгоритмов не может быть применен для выполнения задачи без предварительного обучения. Для решения задачи обнаружения аномального поведения трафика без предварительного обучения выдвигается гипотеза о комбинировании алгоритма *k*-ближайших соседей (KNN, *аббр. от англ*. *k*-Nearest Neighbors) и метода главных компонент (PCA, *аббр. от англ*. Principal Component Analysis). Алгоритм KNN выбран из-за его простоты реализации и высокой точности обнаружения аномалий[1]. Метод PCA основан на многомерном анализе данных и используется для уменьшения размерности данных. PCA находит новые оси (главные компоненты), по которым данные имеют наибольшую дисперсию. Затем данные проецируются на эти главные компоненты, что позволяет сократить количество признаков, сохраняя при этом наибольшую часть их вариации. Это решение легло в основу разработанного алгоритма.

[1]Государственная регистрация программы для ЭВМ «Anomaly Analyzer» (Роспатент). 2023 г.

ТАБЛИЦА 1. Анализ методов машинного обучения

TABLE 1. Analysis of Machine Learning Methods

Критерий	Деревья решений	Метод опорных векторов (SVM)	*k*-ближайших соседей	Случайный лес	*k*-ближайших соседей с методом главных компонент
Применимость	Широкая	Широкая	Ограниченная	Широкая	Ограниченная
Сложность обучения	Средняя	Высокая	Низкая	Средняя	Низкая
Прозрачность	Средняя	Низкая	Высокая	Низкая	Средняя
Требования к данным	Низкие	Высокие	Высокие	Низкие	Высокие
Интерпретируемость	Средняя	Низкая	Высокая	Низкая	Средняя
Вычислительная сложность	Высокая	Высокая	Низкая	Высокая	Низкая
Возможность работы без обучения	Не возможна	Не возможна	Не возможна	Не возможна	Возможна

Способ обнаружения аномального поведения трафика на основе методов искусственного интеллекта

Исходными данными для функционирования предлагаемого способа является база данных трафика с параметрами различных уровней. В основе способа обнаружения аномального поведения трафика на основе методов искусственного интеллекта лежит применение двух методов: PCA и метод *k*-средних. Метод PCA применяется для предобработки данных для последующего применения метода *k*-средних. Для обработки данных перед применением метода PCA, в основе которого лежит процесс приведения данных к стандартному нормальному распределению с нулевым средним и единичной дисперсией. Это делается с помощью вычитания среднего значения и деления на стандартное отклонение для каждой функции (признака).

Для каждого признака вычисляются среднее значение (*от англ.* Mean) и стандартное отклонение (*от англ.* Standard Deviation) по всему набору данных. Преобразование данных производится с помощью вычитания среднего значения (для каждого признака), после чего полученный результат делится на стандартное отклонение. Это преобразование называется центрированием и масштабированием.

Формула для стандартизации данных выглядит следующим образом:

$$x_{std} = \frac{x - mean(x)}{std(x)}, \quad (1)$$

где x_{std} – стандартизированное значение признака x; $mean(x)$ – среднее значение признака x; $std(x)$ – стандартное отклонение признака x.

Результаты стандартизации: после преобразования все признаки имеют среднее значение, равное нулю, и стандартное отклонение, равное единице. Таким образом, данные приводятся к стандартному нормальному распределению.

Полученные в результате этапа «стандартизации» данные применяются на следующем этапе в методе главных компонент. Метод PCA является методом линейного преобразования, который используется для снижения размерности данных. Он позволяет выделить наиболее значимые признаки данных путем проецирования их на новое пространство признаков.

Пусть у нас имеется набор данных X размерности $m \times n$, где m – количество наблюдений, а n – количество признаков. Задача состоит в том, чтобы найти новые оси (главные компоненты), по которым данные будут наиболее широко распределены. Для ускорения работы метода стандартизация была проведена заранее.

После стандартизации данных мы вычисляем ковариационную матрицу Σ, которая показывает связь между всеми парами признаков:

$$\sum \frac{1}{n-1}\left((X - \mu)^T \cdot (X - \mu)\right), \quad (2)$$

где X – матрица данных размерности $m \times n$; μ – вектор средних значений признаков.

Далее мы находим собственные векторы v_i и собственные значения λ_i ковариационной матрицы Σ:

$$\sum v_i = \lambda_i v_i, \quad (3)$$

где v_i – собственный вектор; λ_i – собственное значение.

Главные компоненты выбираются в порядке убывания их соответствующих собственных значений. Таким образом, первая главная компонента соответствует собственному вектору с наибольшим собственным значением, вторая – собственному вектору со вторым по величине собственным значением, и так далее.

На следующем шаге проецируем исходные данные на новое пространство признаков, образованное главными компонентами. Это делается путем умножения исходных данных X на матрицу главных компонент V:

$$X_{pca} = X \cdot V, \quad (4)$$

где X_{pca} – данные в новом пространстве признаков; V – матрица главных компонент, содержащая собственные векторы как столбцы.

Совокупность этих вычислений вместе составляют процесс PCA, который позволяет снизить размерность данных, сохраняя при этом наибольшее количество информации.

После выполнения метода главных компонент применяем метод KNN. Для этого рассчитываем Евклидово расстояние. В алгоритме обнаружения аномалий оно используется для измерения разницы между каждым наблюдением и средним значением всех наблюдений. Это помогает определить, насколько далеко каждое наблюдение находится от среднего значения данных.

Евклидово расстояние является мерой расстояния между двумя точками $P = (p_1, p_2, \dots, p_n)$ и $Q = (q_1, q_2, \dots, q_n)$ в многомерном пространстве и определяется следующим образом:

$$d(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2}, \quad (5)$$

где $d(P, Q)$ – евклидово расстояние между точками P и Q; p_i и q_i – координаты точек P и Q по i-й оси, соответственно.

В контексте обнаружения аномалий евклидово расстояние используется для вычисления расстояния от каждой точки данных до их среднего значения. Это позволяет определить, насколько каждая точка данных отличается от общего среднего значения данных.

Для каждой точки данных X_i Евклидово расстояние вычисляется относительно среднего значения μ по следующей формуле:

$$d(X_i, \mu) = \sqrt{\sum_{j=1}^{n} (X_{ij} - \mu_j)^2}, \quad (6)$$

где X_{ij} – значение i-й точки данных по j-му признаку; μ_j – среднее значение j-го признака по всем точкам данных.

Этот процесс позволяет нам вычислить расстояние от каждой точки данных до их центра (среднего значения) и использовать это расстояние в дальнейшем для определения аномалий.

После расчета Евклидова расстояния устанавливаем пороговое значение. Пороговое значение определяется как среднее расстояние плюс несколько стандартных отклонений. Обычно используется несколько стандартных отклонений от среднего значения для определения диапазона, в пределах которого большинство наблюдений считается нормальными, и точки данных за пределами этого диапазона считаются аномалиями.

Пороговое значение вычисляется по формуле:

$$\text{Порговое значение} = \text{Среднее значение} + \\ + k \times \text{Стандартное отклонение}, \quad (7)$$

где Среднее расстояние – среднее значение всех расстояний между точками данных и их центром; k – коэффициент, который определяет, насколько далеко от среднего значения лежат аномальные точки данных (чем больше значение k, тем более строгим будет критерий для определения аномалий); Стандартное отклонение – мера разброса точек данных относительно их среднего значения.

Чаще всего используется k = 3; это означает, что пороговое значение равно среднему расстоянию плюс три стандартных отклонения. Это пороговое значение затем используется для определения того, какие точки данных считаются аномалиями: если расстояние от точки данных до среднего превышает пороговое значение, то точка считается аномалией.

Алгоритм обнаружения сетевых аномалий на основе методов искусственного интеллекта

На рисунке 1 представлен разработанный алгоритм для обнаружения аномалий в информационных системах и коммутационных инфраструктурах.

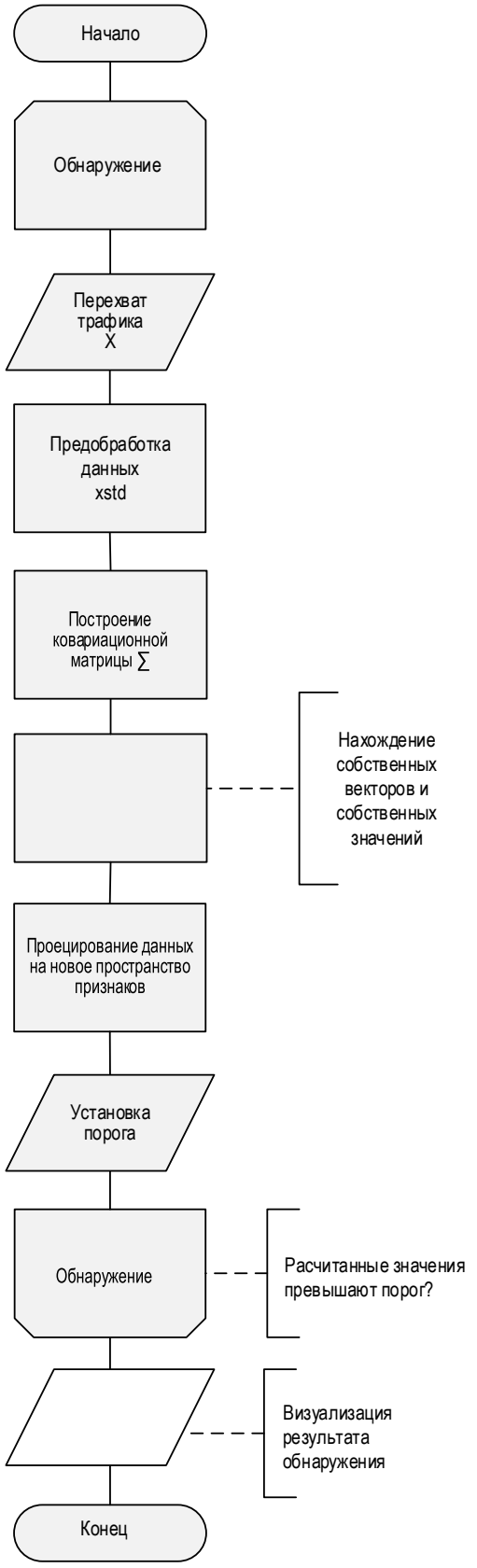


Рис. 1. Алгоритм обнаружения аномалий сетевого трафика

Fig. 1. Network Traffic Anomaly Detection Algorithm

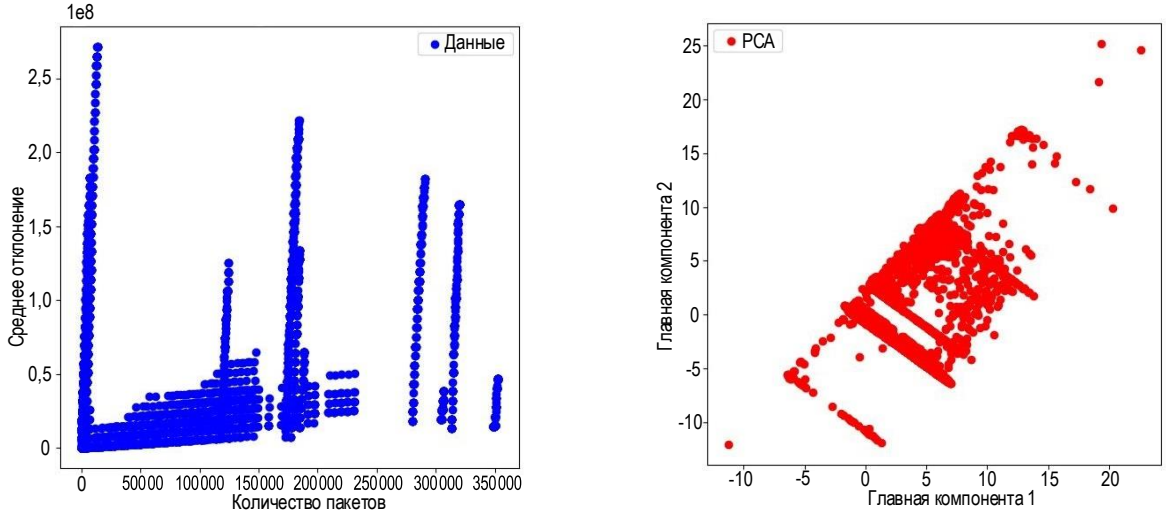
Разработанный алгоритм предназначен для обнаружения признаков аномального поведения трафика в информационных системах и коммутационных инфраструктурах, состоящей из телекоммуникационного оборудования, автоматизированного рабочего места (АРМ) пользователя, системы мониторинга сетевого трафика, развернутой на АРМ администратора.

Процесс перехвата трафика происходит в заданных интервалах времени с выводом отчета в случаях обнаружения аномалий. Процесс поиска аномалий является непрерывным и прекращается в случае остановки работы программного средства обнаружения аномалий. Поиск аномалий основан на методе KNN. Предварительно производится предобработка входных данных, получаемых с помощью протокола Netflow, методом PCA. Такой подход позволил с высокой точностью обнаруживать различные аномалии в сети.

Так как аномалия является подозрением на событие безопасности в сети, следующим этапом является описание каждой аномалии с занесением данного факта в специальный журнал событий. Разработанный алгоритм способен лишь выявить сам факт появления аномалии, но не обладает расширенными механизмами для анализа и полноценной системой принятия решений, которое определяет событие как событие безопасности. Поэтому данные об аномалии необходимо передать на анализ в SIEM-систему.

Результаты эксперимента

Результат верификации предложенного способа обнаружения аномалий сетевого трафика представлены на рисунках 2 и 3. Верификация основана на исследовании трафика из выборки известных аномалий, которая подвергается анализу согласно разработанному алгоритму. Работа алгоритма начинается с нормализации и уменьшения размера данных на основе PCA, результаты которого представлены на рисунке 2.



a) b)

Рис. 2. Графики обнаружения аномалий до (a) и после (b) обработки методом PCA

Fig. 2. Anomaly Detection Plots before (a) and after (b) PCA Treatment

Далее начинается поиск аномалий на основе метода KNN. Автоматически устанавливается пороговое значение, все что его превышает определяется как аномалия. Индексы аномальных наблюдений фиксируются.

Для оценки эффективности методики был проведен натурный эксперимент. Заранее была подготовлена база данных перехваченного трафика с различными сетевыми аномалиями. Эксперимент проводился на двух программных средствах: Anomaly Analyzer [11] и Обнаружение сетевых аномалий на основе методов искусственного интеллекта (разработанное программное средство).

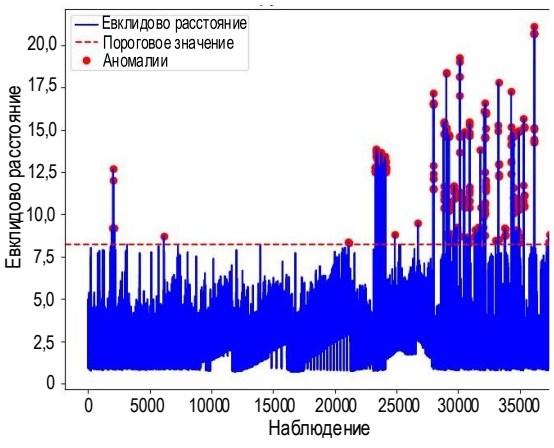


Рис. 3. Обнаружение аномалий

Fig. 3. Anomaly Detection

В ходе эксперимента выявлено (рисунок 4):

– время работы программного средства уменьшилось с 4,3 до 2,4 сек;

– количество ложноположительных срабатываний увеличилось с 252 до 464;

– количество ложноотрицательных срабатываний уменьшилось с 352 до 0.

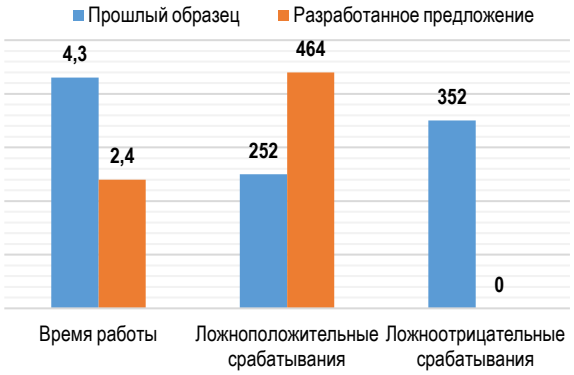


Рис. 4. Оценка эффективности методики обнаружения аномалий сетевого трафика

Fig. 4. Evaluation of the Effectiveness of the Detection Method Network Traffic Anomalies

Данные эксперимента (см. рисунок 4) показали повышение оперативности решения задачи обнаружения аномалий сетевого трафика при использовании разработанной методики: время работы программы снизилось на 1,9 сек.

Введем коэффициент повышения оперативности, равный отношению времени работы разработанного программного средства ко времени работы прошлого образца:

$$\text{КПО} = \frac{2{,}4 \cdot 100}{4{,}3} = 55{,}8\ \%.$$

Вероятность ложноположительных срабатываний можно определить как отношение ложноположительных срабатываний к общему числу срабатываний:

$$P_{\text{ЛС}} = \frac{464 \cdot 100}{956} = 48{,}5\ \%.$$

Вероятность ложноотрицательных срабатываний вычисляется как отношение ложноотрицательных срабатываний к общему числу срабатываний. В ходе эксперимента выявлено, что ложноотрицательные срабатывания отсутствуют, следовательно, их вероятность равна 0.

Таким образом, натурный эксперимент показал, что выдвинутая гипотеза подтверждается.

Заключение

В статье предложен способ обнаружения сетевых аномалий на информационных системах и коммутационных инфраструктурах на основе методов искусственного интеллекта. Разработан алгоритм и программное средство обнаружения сетевых аномалий. Данный способ позволяет повысить оперативность обнаружения сетевых аномалий, а также позволяет обнаруживать новые аномалии, которые не подлежат обнаружению методами сигнатурного анализа, при задействовании меньшего количества пакетов.

Дальнейшим направлением развития исследования является применение методов искусственного интеллекта для классификации аномалий сетевого трафика. Нейронная сеть позволит эффективнее решать задачу обнаружения и классификации сетевых аномалий.

Список источников

1. Dainotti A., Benson K., King A., Claffy K.C., Kallitsis M., Glatz E., et al. Estimating Internet Address Space Usage Through Passive Measurements // ACM SIGCOMM Computer Communication Review. 2011. Vol. 41. Iss. 2. PP. 30–37. DOI:10.1145/2567561.2567568

2. Lazarevic A., Kumar V. Feature Bagging for Outlier Detection // Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD '05, Chicago, USA, 21–24 August 2005). New York: Association for Computing Machinery, 2005. PP. 157–166. DOI:10.1145/1081870.1081891

3. Talukder M.A., Islam M.M., Uddin M.A., Hasan K.F., Sharmin S., Alyami S.A. Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction // Journal of Big Data. 2024. Vol. 11. P. 11. DOI:10.1186/s40537-024-00886-w

4. Шабуров А.С., Никитин А.С. Модель обнаружения компьютерных атак на объекты критической информационной инфраструктуры // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. 2019. № 29. С. 104–117. EDN:ZBKJTN

5. Бугорский М.А., Каплин М.А., Остроцкий С.В., Казакова О.В., Селин В.И. Особенности использования объектов критической информационной инфраструктуры с современной системой обнаружения вторжений // Sciences of Europe. 2021. № 66-1(66). С. 42–46. EDN:SXGMHB. DOI:10.24412/3162-2364-2021-66-1-42-46

6. Семенов В.В., Арустамов С.А. Выявление рисков нарушений информационной безопасности киберфизических систем на основе анализа цифровых сигналов // Научно-технический вестник информационных технологии, механики и оптики. 2020. Т. 20. № 5. С. 770–772. DOI:10.17586/2226-1494-2020-20-5-770-772. EDN:BHITPY

7. Mirkovic J., Prier G., Reiher P. Attacking DDoS at the Source // Proceedings of the 10th IEEE International Conference on Network Protocols (Paris, France, 12–15 November 2002). IEEE, 2002. PP. 312–321. DOI:10.1109/ICNP.2002.1181418

8. Ahmed M., Mahmood A.N., Hu J. A survey of network anomaly detection techniques // Journal of Network and Computer Applications. 2016. Vol. 60. PP. 19–31. DOI:10.1016/j.jnca.2015.11.016

9. Alali A., Yousef M. A Survey on Intrusion Detection Systems (IDS) Using Machine Learning Algorithms // Journal of Xi'an Shiyou University. 2022. Vol. 18. Iss. 6. PP. 183–197.
10. Chandola V., Banerjee A., Kumar V. Anomaly Detection: A Survey // ACM Computing Surveys. 2009. Vol. 41. Iss. 3. PP. 1–58. DOI:10.1145/1541880.1541882. EDN:MYREHF
11. Jordan M.I., Mitchell T.M. Machine learning: Trends, perspectives, and prospects. Science. 2015. Vol. 349. Iss. 6245. PP. 255–260. DOI:10.1126/science.aaa8415

References

1. Dainotti A., Benson K., King A., Claffy K.C., Kallitsis M., Glatz E., et al. Estimating Internet Address Space Usage Through Passive Measurements. *ACM SIGCOMM Computer Communication Review*. 2011;41(2):30–37. DOI:10.1145/2567561.2567568
2. Lazarevic A., Kumar V. Feature Bagging for Outlier Detection. *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, KDD '05, 21–24 August 2005, Chicago, USA*. New York: Association for Computing Machinery; 2005. p.157–166. DOI:10.1145/1081870.1081891
3. Talukder M.A., Islam M.M., Uddin M.A., Hasan K.F., Sharmin S., Alyami S.A. Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction. *Journal of Big Data*. 2024;11:11. DOI:10.1186/s40537-024-00886-w
4. Shaburov A.S., Nikitin A.S. The model for detecting computer attacks on objects of critical information infrastructure. *Bulletin of Perm National Research Polytechnic University. Electrical engineering, information technologies, control systems*. 2019;29:104–117. (in Russ.) EDN:ZBKJTN
5. Bugorsky M., Kaplin M., Ostrotsky S., Kazakova O., Selin V. Features of using critical information infrastructure facilities with a modern intrusion detection system. *Sciences of Europe*. 2021;66-1(66):42–46. (in Russ.) DOI:10.24412/3162-2364-2021-66-1-42-46. EDN:SXGMHB
6. Semenov V.V., Arustamov S.A. Risk identification of security information violations in cyber-physical systems based on analysis of digital signals. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*. 2020;20(5):770–772. (in Russ.) DOI:10.17586/2226-1494-2020-20-5-770-772. EDN:BHITPY
7. Mirkovic J., Prier G., Reiher P. Attacking DDoS at the Source. *Proceedings of the 10th IEEE International Conference on Network Protocols, 12–15 November 2002, Paris, France*. IEEE; 2002. p.312–321. DOI:10.1109/ICNP.2002.1181418
8. Ahmed M., Mahmood A.N., Hu J. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*. 2016;60:19–31. DOI:10.1016/j.jnca.2015.11.016
9. Alali A., Yousef M. A Survey on Intrusion Detection Systems (IDS) Using Machine Learning Algorithms. *Journal of Xi'an Shiyou University.* 2022:18(6):183–197.
10. Chandola V., Banerjee A., Kumar V. Anomaly Detection: A Survey. *ACM Computing Surveys*. 2009;41(3):1–58. DOI:10.1145/1541880.1541882. EDN:MYREHF
11. Jordan M.I., Mitchell T.M. Machine learning: Trends, perspectives, and prospects. *Science*. 2015;349(6245):255–260. DOI:10.1126/science.aaa8415

Статья поступила в редакцию 24.07.2025; одобрена после рецензирования 02.09.2025; принята к публикации 05.09.2025.
The article was submitted 24.07.2025; approved after reviewing 02.09.2025; accepted for publication 05.09.2025.

Информация об авторах:

БЛИЗНЮК Михаил Васильевич — сотрудник Управления специальной связи и информации Федеральной службы охраны Российской Федерации в Северо-Западном федеральном округе
https://orcid.org/0009-0003-5285-2942

БЛИЗНЮК Василий Иванович — кандидат технических наук, сотрудник Академии Федеральной службы охраны Российской Федерации
https://orcid.org/0009-0005-8085-0738

ПОСТАРНАК Андрей Петрович — инженер-исследователь научно-исследовательской и испытательной лаборатория инновационных инфокоммуникаций ПАО «Ростелеком» Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича
https://orcid.org/0009-0001-5779-2948

БОЛБЕНКОВ Александр Владичевич — кандидат технических наук, сотрудник Академии Федеральной службы охраны Российской Федерации
https://orcid.org/0009-0000-3858-6981

КИБАЛИН Антон Юрьевич — сотрудник Академии Федеральной службы охраны Российской Федерации
https://orcid.org/0009-0006-2247-2799

Авторы сообщают об отсутствии конфликтов интересов.
The authors declare no conflicts of interests.