НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
март–апрель 2026 Том 26 № 2 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
March–April 2026 Vol. 26 No 2 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

doi: 10.17586/2226-1494-2026-26-2-367-377
УДК 004.9

Подход к обнаружению DGA-доменов на основе контекстного обучения больших языковых моделей

Артем Бакытжанович Менисов[1]✉, Владимир Михайлович Моргунов[2], Павел Васильевич Тимашов[3]

[1,2,3] Военно-космическая академия имени А.Ф. Можайского, Санкт-Петербург, 197198, Российская Федерация

[1] vka@mil.ru✉, https://orcid.org/0000-0002-9955-2694
[2] vka@mil.ru, https://orcid.org/0009-0008-5949-7820
[3] vka@mil.ru, https://orcid.org/0000-0001-9361-8819

Аннотация

Введение. Рассмотрена проблема обнаружения доменов, сгенерированных алгоритмами генерации доменов (Domain Generation Algorithms, DGA), которые широко используются злоумышленниками для построения устойчивых каналов управления ботнетами и скрытой коммуникации. Традиционные методы основаны на ручной инженерии признаков или специализированных нейросетевых архитектурах, что снижает их устойчивость к быстро эволюционирующим DGA-семействам. Научная новизна предлагаемого подхода заключается в применении больших языковых моделей (БЯМ) с использованием механизма их контекстной адаптации для автоматического выявления скрытых закономерностей в доменных именах и их классификации. **Метод.** Разработанный подход основывается на использовании БЯМ, которые получают в контексте примеры легитимных и сгенерированных доменов. Для повышения эффективности адаптации предложены стратегии выбора примеров (TopK, VoteK), учитывающие метрики близости и вариативность данных. Дополнительно анализируется влияние длины доменного имени и энтропии строки на устойчивость подхода. **Основные результаты.** Экспериментальная часть выполнена на наборе данных, включающем 68 DGA-семейств и подмножестве легитимных доменов Tranco. В обучающую выборку вошли 54 семейства, а тестирование проводилось на 68 семействах, включая невидимые ранее 14. Результаты показали высокую эффективность подхода: Precision = 0,93, Recall = 0,95 и F1-мера = 0,94. Подтверждена способность БЯМ обобщать закономерности на новые DGA-семейства. **Обсуждение.** По сравнению с существующими методами, предложенный подход не требует дополнительного переобучения и отличается гибкостью за счет использования контекстной адаптации. Адаптация подхода показала устойчивость к шуму и возможность выявления новых DGA-семейств, что делает ее перспективной для применения в системах кибербезопасности. В то же время выявлена чувствительность модели к длине доменных имен и необходимость балансировки контекста. Перспективными направлениями развития являются интеграция дополнительных признаков (метаданные Domain Name System (DNS), временные ряды запросов) и адаптация подхода к потоковой обработке в реальном времени.

Ключевые слова

информационная безопасность, DNS-туннелирование, алгоритмы генерации доменов, большие языковые модели, контекстная адаптация

Ссылка для цитирования: Менисов А.Б., Моргунов В.М., Тимашов П.В. Подход к обнаружению DGA-доменов на основе контекстного обучения больших языковых моделей // Научно-технический вестник информационных технологий, механики и оптики. 2026. Т. 26, № 2. С. 367–377. doi: 10.17586/2226-1494-2026-26-2-367-377

An approach to contextual example mining for DGA domain identification using large language models

Artem B. Menisov[1]✉, Vladimir M. Morgunov[2], Pavel V. Timashov[3]

[1,2,3] Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation

[1] vka@mil.ru✉, https://orcid.org/0000-0002-9955-2694
[2] vka@mil.ru, https://orcid.org/0009-0008-5949-7820
[3] vka@mil.ru, https://orcid.org/0000-0001-9361-8819

Abstract

The article addresses the problem of detecting domains generated by Domain Generation Algorithms (DGA) which are widely used by attackers to build robust botnet control channels and covert communication. Traditional methods are based on manual feature engineering or specialized neural network architectures that reduce their robustness to evolving DGA families. The scientific novelty of the proposed approach lies in the use of Large Language Models (LLM) by leveraging their contextual adaptation mechanism to identify hidden patterns in domain names and classify them. The developed approach is based on the use of LLMs which receives examples of legitimate and generated domains within the context. To improve the efficiency, example selection strategies (TopK, VoteK), various metrics of data homogeneity and variability are used. Additionally, the influence of the domain name length and entropy on the stability of the approach is analyzed. The experimental part is performed on a dataset including 68 DGA families and a subset of legitimate Tranco domains. The training set included 54 families, and testing took place on all 68 families, including previously unseen 14 families. Results showed the efficiency of the approach: precision = 0.93, recall = 0.95 and F1-measure = 0.94. The ability of LLM to generalize rules to new DGA families is confirmed. Compared with existing methods, the proposed approach does not require additional retraining and provides flexibility due to contextual adaptation. It demonstrated resistance to noise and the capability to detect new DGA families, which makes its application promising in the field of cybersecurity. At the same time, the sensitivity of the model to the length of domain names and the need for context balancing were revealed. Promising areas of development are the integration of additional features (DNS metadata, query time series) and methods for adaptation to stream processing.

Keywords

information security, DNS tunneling, domain generation algorithms, large language models, contextual adaptation

For citation: Menisov A.B., Morgunov V.M., Timashov P.V. An approach to contextual example mining for DGA domain identification using large language models. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2026, vol. 26, no. 2, pp. 367–377 (in Russian). doi: 10.17586/2226-1494-2026-26-2-367-377

Введение

Для современных информационных систем алгоритмы генерации доменов (Domain Generation Algorithms, DGA) стали серьезной угрозой [1]. DGA открывает злоумышленникам возможность перенаправлять сетевой трафик через собственный узел, подменяя легитимные записи измененными доменными именами (Domain Name System, DNS) и IP-адресами. Основной причиной такой угрозы являются недостатки в механизмах проверки подлинности DNS-участников сетевого взаимодействия и уязвимости DNS-серверов [2]. Для реализации этой угрозы создается большое количество доменных имен, которые в дальнейшем используются для скрытой утечки данных, распространения вредоносного программного обеспечения и фишинговых действий [3]. Доменные имена варьируются по структуре и длине, а для их сокрытия применяются шифрование и обфускация. Кроме того, с помощью DGA злоумышленники обходят системы безопасности, перегружая их большим объемом недавно созданных доменов и увеличивая ложные срабатывания, и создают избыточную нагрузку на систему защиты информационной инфраструктуры.

Учитывая эти трудности, своевременное выявление DGA стало критически важным компонентом современных политик кибербезопасности. Традиционные подходы, такие как статические черные списки и эвристика, оказались недостаточно эффективными для динамической природы DGA [4]. Это привело к повышению интереса к более сложным методам обнаружения, включая машинное обучение за счет использования более сложного извлечения признаков для классификации [5]. В этом контексте большие языковые модели (БЯМ), благодаря способности определять закономерности в шаблонах, имеют высокий потенциал для эффективного выявления DGA [6].

Анализ научно-технических источников

DGA направлены на создание связи с сервером управления для несанкционированного доступа и проведения компьютерных атак типа распределенного отказа в обслуживании [7]. Самый простой подход к связи с сервером — кодирование контролируемых злоумышленником доменов или IP-адресов в исполняемом файле вредоносного программного обеспечения. Однако этот подход может быть легко выявлен, и злоумышленники используют DGA [8], чтобы создать новые доменные имена для серверов управления и контроля на основе случайного начального числа доменов, полученного из открытых источников. Злоумышленники модернизировали различные стратегии генерации доменов, затрудняющие процесс их выявления. Подходы к генерации доменов можно сгруппировать следующим образом [9]: арифметические (доменные имена генерируются с использованием простых арифметических операций); хешированные (генеративный процесс может включать любой генератор случайных чисел с начальными значениями, например, MD5, SHA-1 и т. д.); списочные (под-

ходы, объединяющие слова, фразы или другие записи из словарей); перестановочные (категория DGA, основанная на перестановке частей нескольких начальных состязательных доменов; (генеративный процесс предполагает использование состязательных примеров для DGA); с использованием технологий искусственного интеллекта (использование рекуррентных и генеративных состязательных сетей для оптимизации DGA) [10].

Стоит отметить, что традиционные методы контроля сгенерированных доменов, такие как внесение в черный список, подходы статического сопоставления строк и схемы хеширования, недостаточны для устранения угроз DGA. Анализ современных подходов к выявлению DGA позволил их сгруппировать на контекстно-независимые и контекстно-зависимые подходы к обнаружению DGA [11, 12].

При контекстно-независимом выявлении обрабатываются только доменные имена без информации, связанной с устройством/пользователем и их прошлой активности. Некоторые системы могут включать дополнительную информацию, такую как ответ DNS или информацию WHOIS, оставаясь при этом контекстно-независимыми и считаясь более масштабируемыми и быстрыми, чем контекстно-зависимые методы. Таким образом, методы контекстно-независимого выявления лучше подходят для брандмауэров, работающих на уровне интернет-провайдеров. Эти методы имеют диапазон признаков, таких как *n*-граммы, энтропия и распределение символов и внедрение современных моделей нейронных сетей. В частности, сверточные и рекуррентные нейронные сети продемонстрировали существенные перспективы в выявлении сложных шаблонов в доменных именах [13, 14]. Для улучшения выявления также применяются гибридные подходы, включая ансамблевые решения [15].

Методы контекстно-зависимого выявления DGA обрабатывают индикаторы компрометации устройства/пользователя с течением времени и моделируют их с помощью алгоритмов кластеризации. Поскольку не требуется никаких предварительных знаний о конкретных DGA, эти методы позволяют обнаруживать как известные, так и открывать неизвестные DGA. В работах [16, 17] представлены решения по контекстно-зависимому выявлению в других областях кибербезопасности, предполагая, что можно создать буфер данных и использовать алгоритмы кластеризации для выявления аномалий на этом буфере. Эти подходы могут лучше подходить для брандмауэров в локальных сетях, поскольку их трудно масштабировать на более крупные сети. Модели на основе трансформеров, эффективно используемые в прикладных задачах обработки текстов естественного языка, также применяются для выявления DGA. Использование предварительно обученной модели Bidirectional Encoder Representations from Transformers демонстрирует значительные улучшения в качестве выявления DGA и устойчивости к дисбалансу классов [18]. Появление БЯМ открыло новые возможности в различных областях, включая кибербезопасность [19]. Хотя для выявления DGA БЯМ не так широко используются по сравнению с моделями бустинга или случайного леса, их способность обрабатывать сложные лингвистические шаблоны и обобщать в разных областях предполагает потенциал для идентификации и классификации доменов, сгенерированных DGA.

Несмотря на эти достижения, некоторые трудности сохраняются. Модели должны быстро адаптироваться к новым типам DGA без переобучения. При этом важно сбалансировать точность обнаружения с вычислительной эффективностью для приложений в реальном времени. Дополнительно следует стремиться к низкому уровню ложных срабатываний при высоком уровне обнаружения. Отметим недавние достижения в архитектурах и методах адаптации БЯМ, таких как дополнительная генерация поиска, тонкая настройка и контекстное обучение, которые предлагают пути для улучшения возможностей выявления DGA.

Дополнительная генерация поиска БЯМ заключается в использовании внешних источников данных и состоит из нескольких основных процессов: извлечение, генерация и дополнение вывода, а также механизм определения необходимости извлечения данных из дополнительных данных [20]. Этот подход объединяет предварительно обученную параметрическую память БЯМ с непараметрической памятью в форме векторной базы данных, созданной из внешних данных. Учитывая ограничение БЯМ на обработку длины текста, важно разбить группы DNS на более мелкие сегменты для эффективного извлечения соответствующего контекста. Предварительно в векторной базе данных извлекается наиболее схожая информация для запроса. Сам запрос преобразуется в его векторное представление с использованием той же модели представления, а метрики сходства применяются для извлечения верхних k фрагментов, релевантных запросу. Затем эти фрагменты вводятся в генератор БЯМ вместе с запросом для получения более точных и контекстно-релевантных ответов. Дополнительная генерация поиска БЯМ предлагает улучшенные возможности для обработки сложных и контекстно зависимых данных, но его недостатки, такие как высокая вычислительная сложность, зависимость от базы данных и ограничения масштабируемости, делают его менее подходящим для задач, требующих быстрой адаптации, надежности и низких затрат в условиях изменчивого ландшафта DGA.

Тонкая настройка — подход к адаптации предварительно обученной БЯМ к конкретной задаче, а также повышающий ее производительность [21]. Следуя жизненному циклу машинного обучения, процесс начинается с подготовки специализированного набора данных, содержащего помеченные примеры для поставленной задачи. Далее процесс обучения включает в себя настройку параметров модели для улучшения ее решения. Такие методы, как адаптация низкого ранга, используются для эффективной тонкой настройки модели путем интеграции обучаемых матриц ранговой декомпозиции в каждом слое. Это уменьшает количество обучаемых параметров, делая процесс более эффективным с точки зрения вычислений. Кроме того, методы квантования и дистилляции могут применяться для уменьшения размера модели и времени вывода при сохранении производительности. Стоит отметить, что тонкая настройка

обладает следующими недостатками, ограничивающими ее применение для адаптации предобученных БЯМ для выявления DGA. Во-первых, для тонкой настройки необходимы значительные вычислительные ресурсы для адаптации к специфичным данным DGA. Настроенные модели могут иметь увеличенное время вывода, особенно если их размер остается большим. Это снижает их применимость в задачах реального времени, например, в системах мониторинга трафика. Во-вторых, DGA быстро эволюционируют, создавая новые шаблоны и стратегии генерации доменов. Модели, настроенные на основе существующих данных, могут терять актуальность при появлении новых типов, требуя регулярного переобучения. В-третьих, для эффективной тонкой настройки требуется специализированный и тщательно размеченный набор данных. Если данные нерепрезентативны или имеют дисбаланс классов (например, преобладание безопасных доменов над вредоносными), модель может демонстрировать низкую точность или высокий уровень ложных срабатываний. И, наконец, тонкая настройка определенных архитектур привязывает систему выявления к конкретной технологии. Это может стать проблемой при появлении новых архитектур или методов, требующих полной замены существующих решений.

Хотя тонкая настройка БЯМ обладает потенциалом для улучшения выявления DGA, ее недостатки делают этот подход менее эффективным для задач, требующих высокой адаптивности, интерпретируемости и масштабируемости. Альтернативные подходы, такие как использование гибридных методов или контекстного обучения, могут стать более эффективными в долгосрочной перспективе.

Контекстное обучение представляет собой подход, при котором БЯМ адаптируются к новым задачам без необходимости значительного переобучения [22]. Данный подход имеет большую прикладную значимость, так как позволяет быстро адаптироваться к широкому спектру задач, просто предоставляя примеры в контекст модели. Процесс начинается с проектирования дополнительных примеров, вставки их в контекст модели вместе с новыми данными, что позволяет БЯМ распознавать закономерности. Эта способность дает возможность предобученным БЯМ эффективно выполнять задачи и проводить параллели, демонстрируя универсальность и потенциал контекстного обучения в различных приложениях. Опираясь на эти подходы к адаптации, можно использовать современные предварительно обученные БЯМ для решения задач в области выявления DGA.

В настоящей работе под контекстной адаптацией понимается не обучение модели в параметрическом смысле, а использование заранее подготовленного набора демонстрационных примеров, позволяющих сместить внимание модели в релевантную область латентного пространства. Эффективность контекстной адаптации напрямую определяется взаимосвязью между масштабом модели, ее способностью к селективной активации внутренних представлений и достигнутыми метриками классификации.

Под возможностями модели понимаются архитектурные и вычислительные характеристики, определяющие ее способность удерживать и обрабатывать сложные зависимости: число параметров нейронной сети; размер контекстного окна; устойчивость механизмов внимания к разреженным структурам; стабильность логитного распределения при низкой энтропии входов; активация групп внутренних признаков; подавление нерелевантных траекторий внимания; стабильность решения при добавлении шумовых/длинных доменов; чувствительность к структуре контекста и порядку примеров.

Перечисленные характеристики определяют объем доступного и эффективно используемого латентного пространства модели, а также выступают механизмом выбора подпространства внутри общей латентной структуры модели.

С практической точки зрения наблюдаются следующие особенности:

— для малых моделей контекст действует как жесткий эвристический якорь, тогда как крупные модели используют более гибкие и многомерные паттерны внимания, что позволяет им точнее локализовать «кластер аномальности», характерный для DGA-доменов;
— при недостаточной мощности модели наблюдается эффект «контекстного насыщения», когда добавление примеров деградирует результат из-за конкурирующих фокусов внимания, в то время как более крупные модели демонстрируют плавный рост метрик качества.

Формализованная постановка задачи выявления DGA

Формально, учитывая текст запроса x и набор возможных ответов $Y = \{y_1, y_2, \ldots, y_m\}$ предварительно обученная БЯМ M выдает наиболее возможный ответ $\hat{y}$. Пусть доменное имя представляет собой текст запроса x, а набором возможных ответов Y будут значения типов DGA, обусловленных демонстрационным набором[1]:

$$C = \{I, s(x_1, y_1), \ldots, s(x_k, y_k)\},$$

где C — типы DGA; I — инструкция (системный промт); $s(x_k, y_k)$ — набор, содержащий k демонстрационных примеров и написанный на естественном языке в соответствии с задачей.

Вероятность ответа y_i получается из функции оценки f для всей входной последовательности:

$$P(y_i|x_i) = f(y_i, C, x_i).$$

[1] Разметка пар (x, y) осуществляется предварительно на основе общепринятых открытых источников DGA-семейств и легитимных доменов Tranco. Для DGA-доменов используется пометка на основании источника/генератора; для нормальных — включение в списки доверенных доменов. Дополнительно проводится фильтрация случайных и синтетических доменов, встречающихся в легитимном трафике, чтобы исключить ошибочную маркировку.

Результатом является предсказанная метка $\hat{y}$, определяющая ответ от предобученной БЯМ с наибольшей вероятностью:

$$\hat{y} = \arg\max P(y_i|x_i).$$

Таким образом, целью работы является нахождение оптимального семейства DGA $c \in C$, который может максимизировать выявление DGA.

Ключевым является то, что научная новизна работы заключается не в применении БЯМ к задаче выявления DGA-доменов как таковой, а в подходе к формированию набора демонстрационных примеров (x, y), обеспечивающем целенаправленное смещение модели в релевантную область латентного пространства. В отличие от фиксированных или случайно выбранных примеров, используемых в большинстве существующих исследований, предлагаемый подход обеспечивает систематическую оптимизацию состава контекста.

Набор демонстрационных пар обладает следующими свойствами:

— в набор включаются примеры, отражающие широкий спектр генеративных механизмов DGA (хешевые, арифметические, словарные, гибридные и перестановочные схемы). Это позволяет модели учитывать не один признак (например, энтропию), а совокупность латентных закономерностей;
— контекст содержит как высокоэнтропийные домены, так и псевдо-лексические последовательности, имитирующие человеческий язык. Такой баланс предотвращает переобучение модели на тривиальные признаки;
— оптимальность набора подтверждается тем, что модель сохраняет высокое качество классификации на семействах, отсутствующих в обучающем множестве. Это демонстрирует способность сформированного контекста направлять модель к выявлению обобщенных закономерностей.

Таким образом, набор (x, y) представляет собой не произвольный список примеров, а тщательно отобранный, структурно разнообразный и метрически взвешенный контекст, который обеспечивает максимальное качество классификации и обобщающую способность БЯМ без дополнительного обучения.

Описание подхода

Предложенный подход к выявлению доменов, сгенерированных алгоритмами DGA в условиях DNS-туннелирования, основан на контекстной адаптации БЯМ. Основная идея заключается в разработке инструкций, которые позволяют эффективно использовать примеры для обучения модели $(x_1, y_1), \ldots, (x_k, y_k)$ распознавать DGA-домены. В отличие от стандартных задач, где достаточно объединения примеров с фиксированным шаблоном, в задаче выявления DGA это оказывается затруднительным, так как она требует сложных рассуждений и гибкой интерпретации входных данных. Шаблоны инструкций состоят из трех основных компонентов: описание задачи, генерация примеров и образец выходных данных. Описание задачи и образец выходных данных редко видоизменяются и формируются специалистами в зависимости от смежных систем и протоколов взаимодействия с БЯМ. Генерация примеров является самым сложным компонентом и требует адаптации под динамику появления новых DGA, для которого необходимо решить три этапа: формирование набора примеров; сокращение пространства поиска и выбор и ранжирование примеров.

Этап 1. Формирование набора примеров. Для адаптации БЯМ создается набор данных, включающий множество доменных имен $D_{train} = D_{sec} \cup D_{dga}$, где D_{sec} — множество легитимных доменов; D_{dga} — множество доменов, сгенерированных различными DGA-семействами. В условиях контекстного обучения используются инструкции, включающие компоненты: описание задачи, набор примеров и образец выходных данных:

$$I = \left(sys\left\{(x_i, y_i)\right\}_{i-1}^{k}, format\right),$$

где sys — описание задачи; $\left\{(x_i, y_i)\right\}_{i-1}^{k}$ — множество примеров доменов с метками; $format$ — шаблон выходных данных.

Этап 2. Сокращение пространства поиска. Основной вызов в контекстной адаптации заключается в подборе множества примеров $\left\{(x_i, y_i)\right\}_{i-1}^{k}$, чтобы минимизировать неопределенность БЯМ. Этап сводится к комбинаторной оптимизации, где полный перебор невозможен. Для ее решения в подходе была предложена двухуровневая схема: отбор ограниченного множества примеров; упорядочивание выбранных примеров.

Формально задача выбора сводится к построению подмножества $S \subset D_{train}$, которое соответствует критерию:

$$S = \arg\min_{x_i \in D_{train}} \subset sim(V(x), V(x_i)),$$

где $V(x)$ — векторное представление домена; $sim()$ — метрика близости.

Этап 3. Выбор и ранжирование примеров. Пусть определены пары примеров $X = \left\{(x_i, y_i)\right\}_{i-1}^{k}$. Подмножество $S \subset X$ (например, для $|S| = m$) определяется одним из механизмов: TopK (определяются m ближайших к тестовому x по метрике близости); VoteK (итеративно добавляются близкие, но за избыточно похожие на уже выбранные следует штраф).

При ранжировании из подмножества S выбирается k примеров и выполняется поиск перестановки $c \in C$ (где C — множество всех возможных семейств DGA) по следующему критерию:

$$c^* = \arg\min_{c \in C} L_\theta(y|c, x) + L(\theta),$$
$$L_\theta(y|c, x) \approx -\mathrm{E}_{p(y_i|c,x)} \log_2 p(y_i|c, x), \qquad (1)$$

где E — средняя логарифмическая ошибка; c — упорядоченный список примеров, помещенных в контекст промта перед тестовым входом x; $L(\theta)$ — длина системного промта, где параметры θ фиксированы для всех c. В результате оптимизация сводится к минимизации $L_\theta(y|c, x)$.

$L_\theta(y|c, x)$ — представление домена, применяемое для прогнозирования метки y для данного тестового примера x, если в качестве распределения используется модель с параметрами θ, и, если модель видит контекст c. Формально $L_\theta(y|c, x) = -\log_2 p(y_i|c, x)$, но поскольку при ранжировании не известна истинная метка y, ее заменяют на ожидание по некоторому распределению (выражение (1)).

$-E_{p(y_i|c,x)}\log_2 p(y_i|c, x)$ — математическое ожидание, где случайная величина y_i имеет распределение самой модели, т. е. того, что метка объекта x (например, «DGA» или «Normal») равна y_i, если модель рассматривает данный контекст c.

Пример расчета. Пусть распределение модели: $p(DGA|c,x) = 0{,}9$, $p(Normal|c,x) = 0{,}1$.

Тогда $-E_p\log_2 p = -(0{,}9\log_2 0{,}9 + 0{,}1\log_2 0{,}1) = 0{,}469$ — чем меньше, тем более уверенной является модель.

Таким образом, для БЯМ формируется контекст с примерами, которые одновременно релевантны и разнообразны, что критично при встрече с новыми DGA-семействами.

Экспериментальное исследование

Для оценки предложенного подхода были проведены эксперименты на специально сформированном наборе данных, включающем 68 семейств DGA-доменов и легитимные домены из списка Tranco. Эксперименты направлены на проверку: точности выявления DGA-доменов; способности модели к обобщению на новые DGA-семейства; влияния стратегии выбора и ранжирования примеров на итоговый результат.

Исходные данные. Для обучения и тестирования набор данных, включающий 68 типов DGA-доменов, из которых 54 типа использованы для обучения, и безопасные домены, извлеченные из набора данных Tranco[1], что обеспечивает актуальность и репрезентативность легитимного трафика. 14 семейств (Conficker B, Gozi, DirCrypt, Murofet, Banjori, Symmi, Ramnit, SupremeBot, TheMoon, Virut, Gootloader, DarkShell, Rovnix, Bedep) были полностью исключены из процесса формирования контекста и применялись только для тестирования как неизвестные модели. Сформированный набор данных обеспечивает репрезентативное покрытие DGA-семейств, охватывая различные алгоритмы и стратегии, используемые различными семействами вредоносных программ (рис. 1).

Для обеспечения репрезентативности демонстрационных примеров и предотвращения смещения модели на основании единичных признаков (например, длины строки или частоты символов) в выборку целенаправленно включаются доменные имена, различающиеся по статистическим и структурным свойствам. Такой подход позволяет исключить формирование у модели упрощенной эвристики «длинный/хаотичный домен = DGA» и способствует выявлению более глубоких закономерностей, присущих алгоритмически генерируемым строкам. Это достигается следующим.

1. В контекст включаются домены как с высокой символьной энтропией (случайные и псевдослучайные последовательности, характерные для хешированных и математических DGA), так и с пониженной энтропией (словарные, составные и гибридные DGA, имитирующие человеческую лексику). Таким образом, исключается смещение в сторону высоко-

[1] [Электронный ресурс]. Режим доступа: https://tranco-list.eu/ (дата обращения: 12.10.2025).

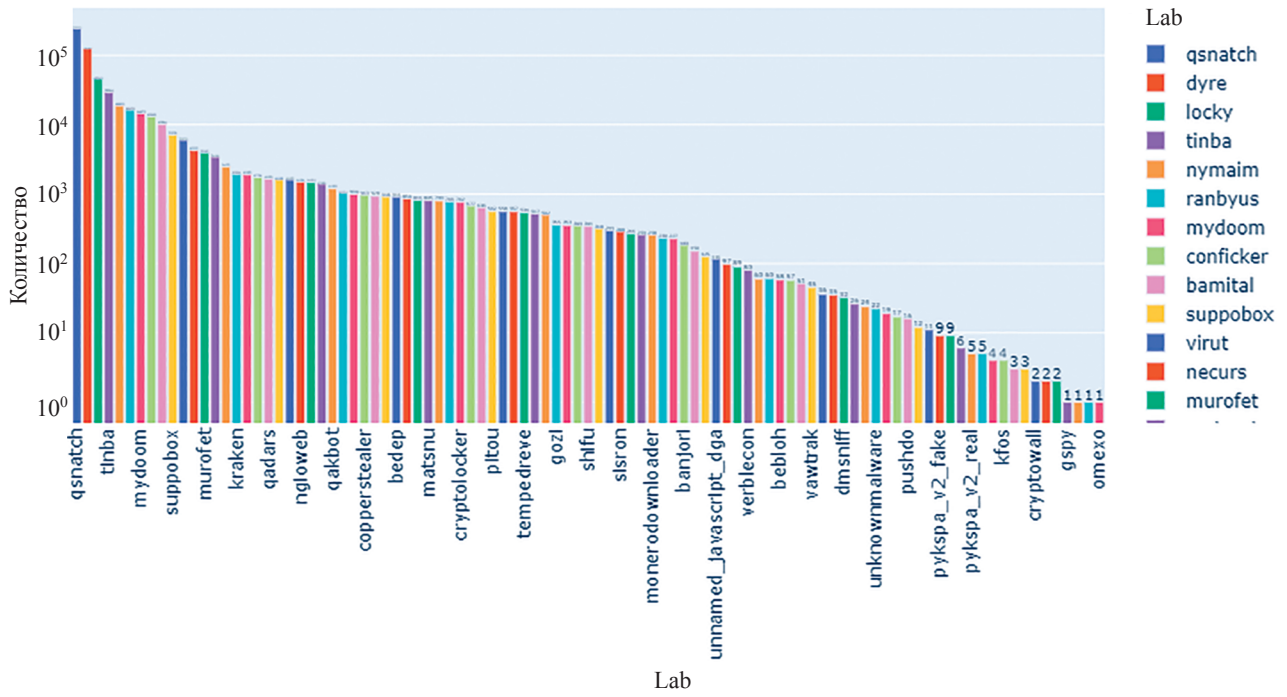

Рис. 1. Распределение доменных имен по классам DGA в исходных данных

Fig. 1. Distribution of domain names by DGA classes in the original data

энтропийных генераторов, которые наиболее очевидны для классификаторов.
2. В выборку включаются домены разных длиновых диапазонов, поскольку DGA могут генерировать как компактные (6–10 символов), так и длинные (более 20 символов) строки. Поддержание сбалансированного распределения предотвращает доминирование признака длины в процессе контекстного смещения модели.
3. Учитываются лингвистические и морфологические свойства (наличие би-/три-грамм, встречающихся в естественном языке, частота гласных и согласных букв, присутствие разделителей (например, в словарных DGA-доменах могут содержаться сегменты вида «news-update-portal.com»).
4. Для выявления не только в случайных последовательностях, но и в доменах, маскирующихся под человеческий язык доменов, в контекст также включаются комбинированные признаки гибридных доменов (например, словарная основа и случайный хвост), доменов, имитирующих бренды или сервисы (фишинговые DGA), а также оборотные строки и перестановочные схемы.

Таким образом, управление энтропией и структурными характеристиками входных данных позволяет сформировать демонстрационный контекст, который стимулирует модель к обучению на комплексных латентных паттернах, а не на поверхностных признаках. Это обеспечивает более высокий уровень обобщения, в том числе на ранее неизвестных DGA-семействах.

Методология проведения эксперимента. В качестве базовой модели использовалась Gemma 3 (4 млрд параметров)[1] без дообучения, что позволяет оценить эффективность именно предложенного механизма контекстной адаптации. Эксперимент выполнялся с детерминированными параметрами БЯМ:
— *temperature* = 0, *top_p* = 1, *top_k* = 0 (жадная декодировка);
— *max_new_tokens* = 4–8 (достаточно для доменного имени);
— stop-токены — перевод строки и двойной перенос.

Процесс эксперимента состоял из следующих шагов.

Шаг 1. Формирование инструкций с использованием различных механизмов выбора и ранжирования примеров:
— базовый шаблон инструкции (без адаптации);
— ближайших по векторным представлениям примеров (TopK);
— с диверсифицированным выбором примеров на основе штрафа за сходство (VoteK);
— комбинация механизмов выбора (TopK/VoteK) и модуля ранжирования на основе принципа минимальной длины описания домена.

Шаг 2. Сравнение результатов адаптации БЯМ с тестовой выборкой. Качество выявления DGA оценивается по следующим метрикам:
— общая точность (ACC) — доля правильно классифицированных доменов;
— точность выявления DGA (Precision) — доля истинных DGA среди предсказанных;
— полнота выявления DGA (Recall) — доля выявленных DGA среди всех фактических;
— F1-мера — гармоническое среднее между Precision и Recall.

Полученные результаты. Каждый запуск БЯМ повторялся для базового шаблона и трех фиксированных формированиях контекста (в табл. 1 приведены средние значения по всем запускам).

Задача: определить класс домена (DGA или Normal).

Примеры:
1) rxqzw.com → DGA
2) sdqwe.net → DGA
3) shop.com → Normal
Тест:
xqzdw.info →

Для оценки универсальности предложенного подхода формирования контекстного набора дополнительно была проведена серия экспериментов с использованием других БЯМ, отличающихся архитектурой, принципами обучения и объемом обучающих данных. В эксперимент были включены как проприетарные решения (DeepSeek, обладающие широким языковым охватом и развитым латентным представлением), так и открытые предобученные модели трансформерного и нетрансформерного типов (LLaMA-3, представляющая классическую архитектуру с обширным корпусом; Mistral Small 3, характеризующаяся компактностью и оптимизированными механизмами внимания; а также RWKV-7B, использующая альтернативный механизм работы с контекстом без классического механизма внимания).

Для всех моделей применены одинаковые демонстрационные примеры и формат инструкции, что позволило оценить именно способность модели к контекстному смещению (табл. 2), а не влияние тонкой настройки или параметров обучения.

[1] [Электронный ресурс]. Режим доступа: https://huggingface.co/google/gemma-3-4b-it (дата обращения: 12.10.2025).

Таблица 1. Качество выявления доменов для базовой модели
Table 1. Domain identification quality for the base model

Подход	ACC	Precision	Recall	F1-мера
Базовый шаблон	0,81	0,79	0,83	0,81
TopK	0,87	0,85	0,88	0,86
VoteK	0,89	0,87	0,90	0,88
Подход адаптации	**0,94**	**0,93**	**0,95**	**0,94**

Таблица 2. Качество выявления доменов
Table 2. Domain identification quality

Модель	Тип	Параметры, млрд	F1-мера	Особенности поведения
DeepSeek	проприетарный	671	**0,94**	стабильное обобщение, точный учет паттернов
LLaMA-3	Ollama	70	0,89	склонность к энтропийной эвристике
Mistral Small 3	Ollama	24	0,88	высокая точность на хеш-DGA, падение на словарных семействах
RWKV-7B	Ollama	7	0,83	хуже перенос лексических закономерностей

Таким образом, качество модели влияет на абсолютные метрики, однако разработанный подход выбора контекстных примеров остается эффективным и масштабируемым на разные архитектуры, что подтверждает его универсальность.

Обсуждение

Одним из ключевых факторов, оказывающих влияние на эффективность применения разработанного подхода для выявления доменов, сгенерированных алгоритмами DGA, является длина доменного имени (рис. 2).

Анализ экспериментальных результатов и особенностей работы БЯМ позволяет выделить несколько направлений, в которых параметр длины доменных имен может как усиливать, так и ослаблять способность модели к корректной классификации.

Сгенерированные алгоритмами DGA доменные имена, как правило, отличаются чрезмерной длиной или атипичной структурой. Их характерные особенности включают случайные подстроки, отсутствие семантической интерпретации и высокую энтропию. Это облегчает задачу различения DGA-доменов от легитимных, но одновременно вносит значительное количество шума в представление данных, так как бессмысленные символы могут быть восприняты моделью как нерелевантный контекст (рис. 3).

На основании полученных результатов можно выделить основные риски решения проблемы длины доменных имен, связанные с: дисбалансом данных в контексте (однотипная длина примеров); состязательным шумом, препятствующим семантической интерпретации; переоценкой признака «длина» в ущерб другим характеристикам (паттерны символов, статистика запросов).

1. Длина доменного имени напрямую затрагивает контекстное окно БЯМ. Поскольку размер окна ограничен, длинные домены могут вытеснять более информативные признаки (например, статистику запросов или метаданные), снижая качество анализа.
2. Доменные имена токенизируются, и чем больше длина, тем выше вероятность «размывания» полезных закономерностей. С одной стороны, высокая энтропия последовательности может быть индикатором DGA; с другой — избыточное разнообразие символов повышает риск ложных срабатываний, особенно в случаях, когда легитимные домены содержат редкие или нестандартные символы.
3. Контекстная адаптация через предоставленные примеры особенно чувствительна к дисбалансу по длине доменов. Если контекст преимущественно

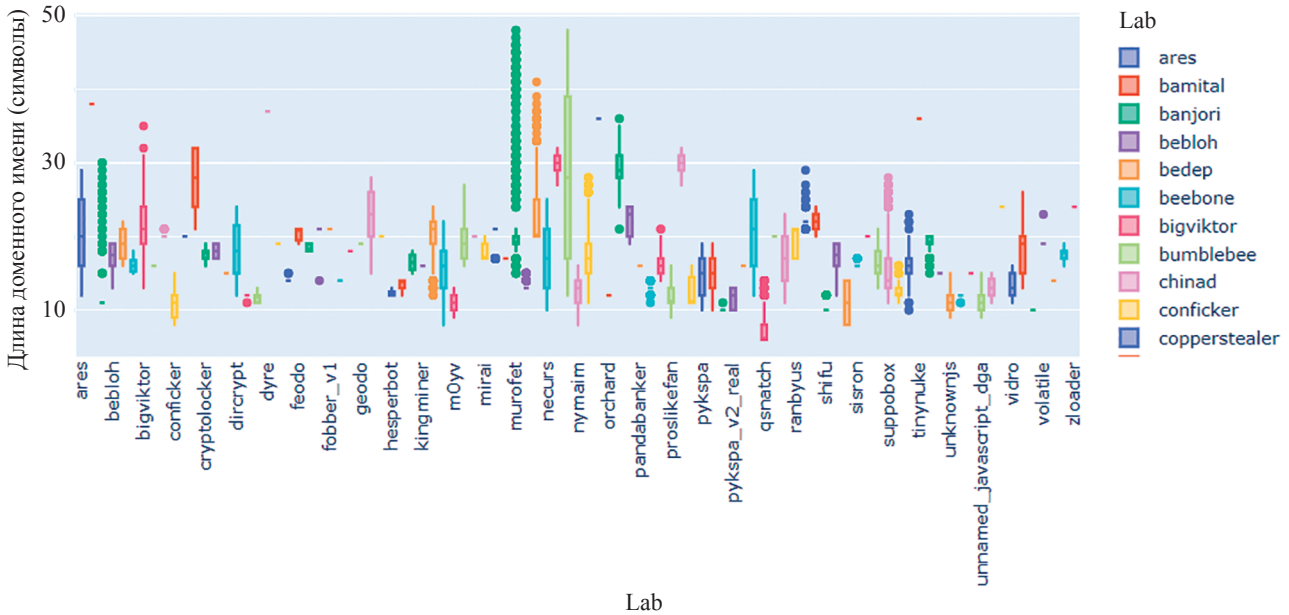

Рис. 2. Распределение длин доменных имен по классам
Fig. 2. Distribution of domain name lengths by class

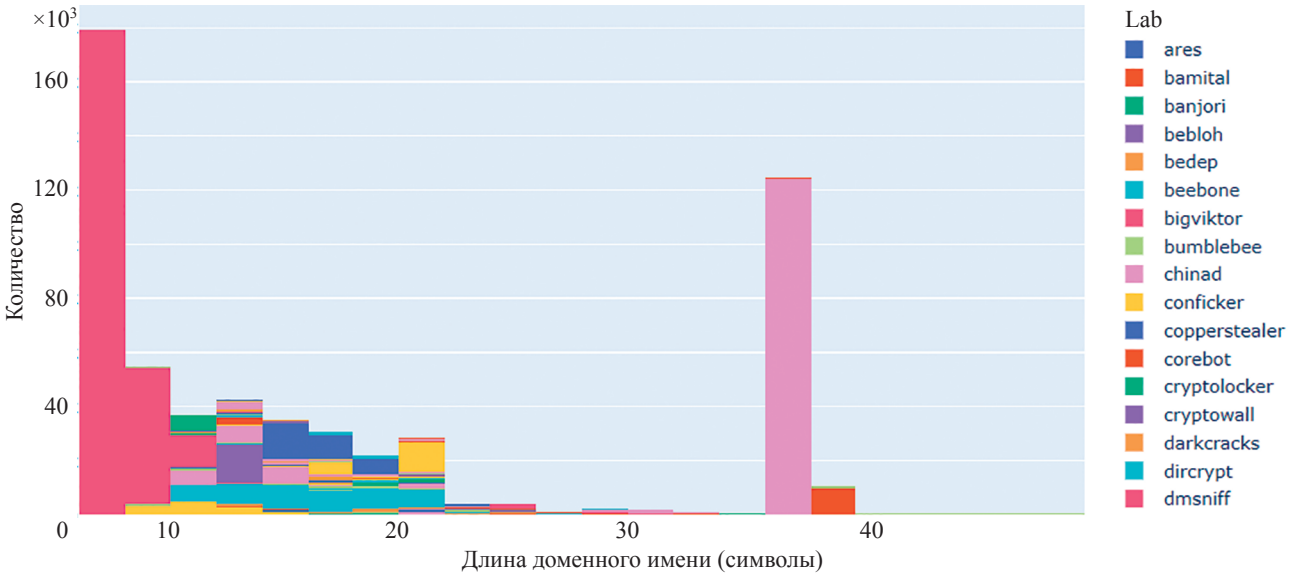

Рис. 3. Гистограмма длин доменных имен

Fig. 3. Histogram of domain name lengths

формируется из длинных DGA-доменов, модель склонна формировать эвристику «длина = подозрительность», что приводит к снижению обобщающей способности. Напротив, сбалансированное представление как коротких, так и длинных доменов в обучающих примерах позволяет БЯМ выявлять более комплексные закономерности — такие как специфические комбинации символов или структурные особенности генеративных схем.

Особое внимание следует уделить генеративным DGA, для которых длина не фиксирована и меняется динамически. В таких случаях избыточная концентрация на длине приводит к переобучению и модель правильно классифицирует длинные домены, но теряет способность к выявлению коротких экземпляров.

Таким образом, обсуждение показало, что длина доменного имени является важным, но далеко не единственным фактором, определяющим успешность выявления DGA-доменов с помощью БЯМ. Сбалансированная интеграция этого признака с другими характеристиками повышает устойчивость подхода к шуму и обеспечивает более высокую обобщающую способность модели.

Предлагаемый подход опирается на механизм внимания трансформерных архитектур, что накладывает ряд фундаментальных ограничений. Вычислительная сложность формирования матрицы внимания растет квадратично относительно длины входной последовательности $O(n^2)$. Таким образом, увеличение числа демонстрационных примеров в контексте приводит к росту нагрузки на вычислительные ресурсы и увеличению времени отклика модели. В связи с этим в работе используется ограниченный размер контекста ($n < 40$), позволяющий сохранять баланс между информативностью примеров и вычислительной эффективностью.

При увеличении контекста возможно формирование разреженных матриц внимания, что потенциально снижает стабильность и качество модели, особенно при наличии неоднородных или дублирующих паттернов доменных имен. В рассматриваемом подходе это нивелируется за счет отбора наиболее релевантных и разнообразных примеров, а также контроля их порядка в контексте.

При потоковой обработке DNS-трафика возможна ситуация переполнения доступного контекстного окна модели. Для таких сценариев предполагается использование стратегии скользящего окна с адаптивным обновлением набора примеров и исключением устаревших, что позволяет предотвращать деградацию внимания и обеспечивает устойчивость вывода.

Трансформерные модели обладают фиксированным размером контекстного пространства, который является архитектурно-зависимым и может различаться для разных семейств БЯМ. Масштабирование подхода на модели с более ограниченным контекстом или работа в условиях высоконагруженных систем требует дальнейшего изучения и возможной интеграции механизмов внимания и внешней памяти.

Заключение

В работе предложен и экспериментально подтвержден подход к выявлению доменов туннелей Domain Name System (DNS), сгенерированных алгоритмами генерации доменов (Domain Generation Algorithms, DGA), основанный на контекстной адаптации больших языковых моделей (БЯМ). В отличие от традиционных методов, требующих ручной инженерии признаков или дополнительного обучения, разработанный механизм использует встроенную способность БЯМ к выявлению скрытых закономерностей в текстовых последовательностях и к контекстной адаптации.

Проведенные эксперименты показали, что включение релевантных и сбалансированных примеров в контекст значительно повышает эффективность выявления DGA-доменов, обеспечивая высокие пока-

затели общей точности (равной 0,94), Recall = 0,95 и F1-меры = 0,94. Особенно важным результатом стало подтверждение способности модели к обобщению на новые DGA-семейства, ранее не встречавшиеся в обучающих данных, что подчеркивает практическую ценность предложенного подхода в условиях быстро эволюционирующих угроз.

Отдельное внимание уделялось анализу влияния длины доменных имен на работу БЯМ. Результаты показали, что чрезмерная длина и высокая энтропия строк действительно являются индикаторами DGA, однако использование этого признака в отрыве от других может приводить к смещению и ложным срабатываниям. Включение разнообразных по длине доменных имен в контекст позволяет компенсировать этот эффект и повысить устойчивость подхода к шуму.

Таким образом, разработанный подход сочетает преимущества гибкости БЯМ и строгих принципов оптимизации выбора примеров, что делает его перспективным инструментом для задач кибербезопасности, связанных с выявлением DGA. В дальнейшем планируется исследовать: интеграцию дополнительных источников контекста (метаданные DNS, временные ряды запросов); использование гибридных стратегий выбора примеров (обучаемых механизмов ранжирования); адаптацию подхода для потоковой обработки в реальном времени и масштабных сетевых инфраструктур.

Предложенный подход открывает новые возможности для применения БЯМ в области информационной безопасности, формируя основу для более интеллектуальных и устойчивых систем обнаружения вторжений.

Литература

1. Hassaoui M., Hanini M., Kafhali S.E. Data science in cybersecurity to detect malware-based domain generation algorithm: improvement, challenges, and prospects // Journal of Computational and Cognitive Engineering. 2024. V. 3. N 3. P. 213–225. https://doi.org/10.47852/bonviewJCCE42022875
2. Albluwi A., Albalawi U., Elfaki A.O. A DNS threat awareness practical framework using knowledge graph // Journal of Information Science and Engineering. 2025. V. 41. P. 1239–1261.
3. Arora A., Shantanu. A review on application of GANs in cybersecurity domain // IETE Technical Review. 2022. V. 39. N 2. P. 433–441. https://doi.org/10.1080/02564602.2020.1854058
4. Patsakis C., Casino F. Exploiting statistical and structural features for the detection of Domain Generation Algorithms // Journal of Information Security and Applications. 2021. V. 58. P. 102725. https://doi.org/10.1016/j.jisa.2020.102725
5. Kolte S., Jare A., Babar V., Kadam S., Tekade P., Salunke D. A machine learning-based framework for real-time DNS threat detection and mitigation using ensemble models and advanced security mechanisms // Proc. of the International Conference on Electronics, AI and Computing (EAIC). 2025. P. 1–6. https://doi.org/10.1109/EAIC66483.2025.11101638
6. Pelayo-Benedet T., Rodríguez R.J., Gañán C.H. Poster: Exploring the zero-shot potential of large language models for detecting algorithmically generated domains // Lecture Notes in Computer Science. 2025. V. 15748. P. 86–92. https://doi.org/10.1007/978-3-031-97623-0_5
7. Alorainy W.S. Echoes from the void: detecting DNS tunneling with blackhole features in encrypted scenarios with high accuracy // IEEE Access. 2025. V. 13. P. 138551–138567. https://doi.org/10.1109/ACCESS.2025.3595455
8. Sharma N., Swarnkar M. DLAZE: Detecting DNS tunnels using lightweight and accurate method for zero-day exploits // IEEE Transactions on Network and Service Management. 2025. V. 22. N 3. P. 2343–2353. https://doi.org/10.1109/TNSM.2025.3541234
9. Fu Y., Yu L., Hambolu O., Ozcelik I., Husain B., Sun J., et al. Stealthy domain generation algorithms // IEEE Transactions on Information Forensics and Security. 2017. V. 12. N 6. P. 1430–1443. https://doi.org/10.1109/TIFS.2017.2668361
10. Cao Y., Li S., Liu Y., Yan Z., Dai Y., Yu P., Sun L. A survey of AI-Generated Content (AIGC) // ACM Computing Surveys. 2025. V. 57. N 5. P. 1–38. https://doi.org/10.1145/3704262
11. De Bernardi G., Gaggero G.B., Patrone F., Zappatore S., Marchese M., Mongell M. Rule-based eXplainable autoencoder for DNS tunneling detection // Computers. 2025. V. 14. N 9. P. 375. https://doi.org/10.3390/computers14090375
12. Bykov N., Chernyshov Y. Detecting DNS tunnels using machine learning // Proc. of the IEEE Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT). 2024. P. 92–94. https://doi.org/10.1109/usbereit61901.2024.10584043

References

1. Hassaoui M., Hanini M., Kafhali S.E. Data science in cybersecurity to detect malware-based domain generation algorithm: improvement, challenges, and prospects. *Journal of Computational and Cognitive Engineering*, 2024, vol. 3, no. 3, pp. 213–225. https://doi.org/10.47852/bonviewJCCE42022875
2. Albluwi A., Albalawi U., Elfaki A.O. A DNS threat awareness practical framework using knowledge graph. *Journal of Information Science and Engineering*, 2025, vol. 41, pp. 1239–1261.
3. Arora A., Shantanu. A review on application of GANs in cybersecurity domain. *IETE Technical Review*, 2022, vol. 39, no. 2, pp. 433–441. https://doi.org/10.1080/02564602.2020.1854058
4. Patsakis C., Casino F. Exploiting statistical and structural features for the detection of Domain Generation Algorithms. *Journal of Information Security and Applications*, 2021, vol. 58, pp. 102725. https://doi.org/10.1016/j.jisa.2020.102725
5. Kolte S., Jare A., Babar V., Kadam S., Tekade P., Salunke D. A machine learning-based framework for real-time DNS threat detection and mitigation using ensemble models and advanced security mechanisms. *Proc. of the International Conference on Electronics, AI and Computing (EAIC)*, 2025, pp. 1–6. https://doi.org/10.1109/EAIC66483.2025.11101638
6. Pelayo-Benedet T., Rodríguez R.J., Gañán C.H. Poster: Exploring the zero-shot potential of large language models for detecting algorithmically generated domains. *Lecture Notes in Computer Science*, 2025, vol. 15748, pp. 86–92. https://doi.org/10.1007/978-3-031-97623-0_5
7. Alorainy W.S. Echoes from the void: detecting DNS tunneling with blackhole features in encrypted scenarios with high accuracy. *IEEE Access*, 2025, vol. 13, pp. 138551–138567. https://doi.org/10.1109/ACCESS.2025.3595455
8. Sharma N., Swarnkar M. DLAZE: Detecting DNS tunnels using lightweight and accurate method for zero-day exploits. *IEEE Transactions on Network and Service Management*, 2025, vol. 22, no. 3, pp. 2343–2353. https://doi.org/10.1109/TNSM.2025.3541234
9. Fu Y., Yu L., Hambolu O., Ozcelik I., Husain B., Sun J., et al. Stealthy domain generation algorithms. *IEEE Transactions on Information Forensics and Security*, 2017, vol. 12, no. 6, pp. 1430–1443. https://doi.org/10.1109/TIFS.2017.2668361
10. Cao Y., Li S., Liu Y., Yan Z., Dai Y., Yu P., Sun L. A survey of AI-Generated Content (AIGC). *ACM Computing Surveys*, 2025, vol. 57, no. 5, pp. 1–38. https://doi.org/10.1145/3704262
11. De Bernardi G., Gaggero G.B., Patrone F., Zappatore S., Marchese M., Mongell M. Rule-based eXplainable autoencoder for DNS tunneling detection. *Computers*, 2025, vol. 14, no. 9, pp. 375. https://doi.org/10.3390/computers14090375
12. Bykov N., Chernyshov Y. Detecting DNS tunnels using machine learning. *Proc. of the IEEE Ural-Siberian Conference on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)*, 2024, pp. 92–94. https://doi.org/10.1109/usbereit61901.2024.10584043

13. Namgung J., Son S., Moon Y.-S. Efficient deep learning models for DGA domain detection // Security and Communication Networks. 2021. V. 2021. N 1. P. 8887881. https://doi.org/10.1155/2021/8887881
14. Zhou S., Lin L., Yuan J., Wang F., Ling Z., Cui J. CNN-based DGA detection with high coverage // Proc. of the IEEE International Conference on Intelligence and Security Informatics (ISI). 2019. P. 62–67. https://doi.org/10.1109/isi.2019.8823200
15. Vu X.H., Hoang X.D., Chu T.H.H. A novel model based on ensemble learning for detecting DGA botnets // Proc. of the 14th International Conference on Knowledge and Systems Engineering (KSE). 2022. P. 1–6. https://doi.org/10.1109/kse56063.2022.9953792
16. Tapsoba A.R., Ouédraogo T.F., Zongo W.B.S. Analysis of plaintext features in DoH traffic for DGA domains detection // Lecture Notes in Networks and Systems. 2024. V. 932. P. 127–138. https://doi.org/10.1007/978-3-031-54235-0_12
17. Harishkumar S., Bhuvaneswaran R.S. Enhanced DGA detection in Botnet traffic: leveraging N-Gram, topic modeling, and attention BiLSTM // Peer-to-Peer Networking and Applications. 2025. V. 18. N 1. P. 55. https://doi.org/10.1007/s12083-024-01822-8
18. Tian Y., Li Z. Dom-Bert: Detecting malicious domains with pre-training model // Lecture Notes in Computer Science. 2024. V. 14537. P. 133–158. https://doi.org/10.1007/978-3-031-56249-5_6
19. Zhang J., Bu H., Wen H., Liu Y., Fei H., Xi R., et al. When LLMs meet cybersecurity: a systematic literature review // Cybersecurity. 2025. V. 8. N 1. P. 55. https://doi.org/10.1186/s42400-025-00361-w
20. Arslan M., Ghanem H., Munawar S., Cruz C. A survey on RAG with LLMs // Procedia Computer Science. 2024. V. 246. P. 3781–3790. https://doi.org/10.1016/j.procs.2024.09.178
21. Wu X.-K., Chen M., Li W., Wang R., Lu L., Liu J., et al. LLM Fine-tuning: concepts, opportunities, and challenges // Big Data and Cognitive Computing. 2025. V. 9. N 4. P. 87. https://doi.org/10.3390/bdcc9040087
22. Highmore C. In-context learning in large language models: a comprehensive survey // Preprints.org. 2024. 11 p. https://doi.org/10.20944/preprints202407.0926.v1

13. Namgung J., Son S., Moon Y.-S. Efficient deep learning models for DGA domain detection. *Security and Communication Networks*, 2021, vol. 2021, no. 1, pp. 8887881. https://doi.org/10.1155/2021/8887881
14. Zhou S., Lin L., Yuan J., Wang F., Ling Z., Cui J. CNN-based DGA detection with high coverage. *Proc. of the IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2019, pp. 62–67. https://doi.org/10.1109/isi.2019.8823200
15. Vu X.H., Hoang X.D., Chu T.H.H. A novel model based on ensemble learning for detecting DGA botnets. *Proc. of the 14th International Conference on Knowledge and Systems Engineering (KSE)*, 2022, pp. 1–6. https://doi.org/10.1109/kse56063.2022.9953792
16. Tapsoba A.R., Ouédraogo T.F., Zongo W.B.S. Analysis of plaintext features in DoH traffic for DGA domains detection. *Lecture Notes in Networks and Systems*, 2024, vol. 932, pp. 127–138. https://doi.org/10.1007/978-3-031-54235-0_12
17. Harishkumar S., Bhuvaneswaran R.S. Enhanced DGA detection in Botnet traffic: leveraging N-Gram, topic modeling, and attention BiLSTM. *Peer-to-Peer Networking and Applications*, 2025, vol. 18, no. 1, pp. 55. https://doi.org/10.1007/s12083-024-01822-8
18. Tian Y., Li Z. Dom-Bert: Detecting malicious domains with pre-training model. *Lecture Notes in Computer Science*, 2024, vol. 14537, pp. 133–158. https://doi.org/10.1007/978-3-031-56249-5_6
19. Zhang J., Bu H., Wen H., Liu Y., Fei H., Xi R., et al. When LLMs meet cybersecurity: a systematic literature review. *Cybersecurity*, 2025, vol. 8, no. 1, pp. 55. https://doi.org/10.1186/s42400-025-00361-w
20. Arslan M., Ghanem H., Munawar S., Cruz C. A survey on RAG with LLMs. *Procedia Computer Science*, 2024, vol. 246, pp. 3781–3790. https://doi.org/10.1016/j.procs.2024.09.178
21. Wu X.-K., Chen M., Li W., Wang R., Lu L., Liu J., et al. LLM Fine-tuning: concepts, opportunities, and challenges. *Big Data and Cognitive Computing*, 2025, vol. 9, no. 4, pp. 87. https://doi.org/10.3390/bdcc9040087
22. Highmore C. In-context learning in large language models: a comprehensive survey. *Preprints.org*, 2024, 11 p. https://doi.org/10.20944/preprints202407.0926.v1

Авторы

Менисов Артем Бакытжанович — доктор технических наук, старший преподаватель, Военно-космическая академия имени А.Ф. Можайского, Санкт-Петербург, 197198, Российская Федерация, SC 57220815185, https://orcid.org/0000-0002-9955-2694, vka@mil.ru
Моргунов Владимир Михайлович — кандидат технических наук, доцент, доцент, Военно-космическая академия имени А.Ф. Можайского, Санкт-Петербург, 197198, Российская Федерация, https://orcid.org/0009-0008-5949-7820, vka@mil.ru
Тимашов Павел Васильевич — кандидат технических наук, преподаватель, Военно-космическая академия имени А.Ф. Можайского, Санкт-Петербург, 197198, Российская Федерация, https://orcid.org/0000-0001-9361-8819, vka@mil.ru

Authors

Artem B. Menisov — D.Sc., Senior Lecturer, Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation, SC 57220815185, https://orcid.org/0000-0002-9955-2694, vka@mil.ru

Vladimir M. Morgunov — PhD, Associate Professor, Associate Professor, Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation, https://orcid.org/0009-0008-5949-7820, vka@mil.ru
Pavel V. Timashov — PhD, Lecturer, Mozhaisky Military Aerospace Academy, Saint Petersburg, 197198, Russian Federation, https://orcid.org/0000-0001-9361-8819, vka@mil.ru

Статья поступила в редакцию 12.09.2025
Одобрена после рецензирования 02.12.2025
Принята к печати 21.03.2026

Received 12.09.2025
Approved after reviewing 02.12.2025
Accepted 21.03.2026