DOI: 10.55648/1998-6920-2026-20-1-3-22 УДК 004.056

Методика оценки эффективности автоматизированного конвейера обнаружения фишинга на основе FastText в рамках специализированной библиотеки AutoML для анализа барьеров внедрения искусственного интеллекта в современные системы обнаружения вторжений

С. И. Штеренберг, Д. Н. Гречухин, А. С. Кривец

ФГБОУ ВО «Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича» (СПбГУТ)

Аннотация: В статье представлен подход к обнаружению фишинговых атак на основе построения и автоматической оптимизации конвейеров машинного обучения с помощью специализированной библиотеки (PhishAutoML). Актуальность проблемы обусловлена эволюцией фишинговых атак, использующих методы социальной инженерии и лексические уловки, что делает традиционные статические методы защиты неэффективными. Описаны теоретические основы векторизации текста с помощью модели FastText и ее применение в рамках AutoML-подхода на основе байесовской оптимизации, позволяющего автоматически подбирать гиперпараметры для всего конвейера. Предложенная концепция PhishAutoML применяется для построения моделей, способных выявлять фишинг на основе семантического анализа и гибко настраивать компромисс между качеством и производительностью. Представлены результаты вычислительных экспериментов: итоговые метрики качества и производительности, а также сравнительный анализ с классическим (TF-IDF) и современным (DistilBERT) подходами. Выводы подтверждают эффективность предложенного решения (достижение полноты обнаружения фишинга в 95%, что в разы превосходит альтернативные методы) и намечают направления дальнейшего развития. Однако такая интеграция в традиционные системы обнаружения и предотвращения вторжений сталкивается с серьезными рисками и фундаментальными проблемами. В данной статье рассматриваются ключевые технические, организационные и этические барьеры, препятствующие массовому внедрению решений, использующих ИИ, и предлагаются возможные пути их преодоления.

Ключевые слова искусственный интеллект, обнаружение вторжений, предотвращение вторжений, обработка естественного языка (NLP), AutoML, FastText, байесовская оптимизация.

Для цитирования: Штеренберг С. И., Гречухин Д. Н., Кривец А. С. Методика оценки эффективности автоматизированного конвейера обнаружения фишинга на основе FastText в рамках специализированной библиотеки AutoML для анализа барьеров внедрения искусственного интеллекта в современные системы обнаружения вторжений // Вестник СибГУТИ. 2026. Т. 20, № 1. С. 3–22. https://doi.org/10.55648/1998-6920-2026-20-1-3-22.

Статья поступила в редакцию 16.10.2025;
переработанный вариант – 22.10.2025;
принята к публикации 05.02.2026.

1. Введение

Современный ландшафт киберугроз характеризуется беспрецедентной сложностью, скоростью и масштабом. Традиционные сигнатурные методы обнаружения, основанные на известных шаблонах, уже не справляются с целевыми атаками, полиморфным вредоносным программным обеспечением (далее – ПО) и изощренными техниками социальной инженерии, так как злоумышленники разрабатывают шифрованные, самоизменяющиеся и мутирующие вирусы. В этом контексте искусственный интеллект (далее – ИИ), способный к самообучению и анализу огромных массивов данных, видится идеальным решением. Но между теоретическим потенциалом и практическим внедрением лежит пропасть, обусловленная рядом критических барьеров.

Обеспечение безопасности информации и данных является одной из главных целей в нынешних реалиях. Под обеспечением безопасности зачастую подразумевается использование обычных антивирусов и систем защиты. Но существуют такие типы атак, от которых не всегда получается защищаться традиционными способами. Это может привести к краже паролей, логинов и т. п., но также существует риск проникновения злоумышленника в систему для выполнения своих преступных намерений, которые могут привести к финансовым и иным потерям. Таким образом, обеспечение безопасности системы от нестандартных атак является приоритетной задачей, для которой необходимо разрабатывать новые рубежи защиты [1].

Существуют различные методы защиты безопасности: использование SIEM систем, DLP, антивирусы, фаерволы, шифрование и другие. Многие из этих методов работают по шаблону, заранее известным угрозам. Те же антивирусы работают по базам сигнатур и могут не всегда реагировать на опасность. Однако существует человеческий фактор, который активно эксплуатируется для проведения атак, вследствие чего возникает необходимость создания систем, которые будут фильтровать информацию, получаемую пользователями и предотвращать угрозу на начальной фазе [2].

Одним из перспективных направлений для решения подобных задач является использование методов машинного обучения (далее – МО), в частности обработка естественного языка (NLP). NLP уже зарекомендовало себя в задачах классификации спама и анализе тональности текста. Преимуществом данного подхода является возможность обучаться на больших объемах данных, подбирая оптимальные параметры для решения поставленной задачи, а также адаптироваться к любым изменениям, которые обычные системы могли бы пропустить. В контексте атак, использующих человеческий фактор, таких как фишинг, использование NLP позволит создать модель, которая может комплексно оценивать текст, учитывать его особенности, а не только оценивать слова, которые имеются в тексте [3].

Одним из таких типов атак является фишинг. Теперь это не просто массовые рассылки того же вредоносного ПО, но и использование социальной инженерии. Например, замена обычных букв символами и т. п. Эволюция таких атак демонстрирует смещение от массовых рассылок с примитивными текстовыми шаблонами к сложным, персонализированным сообщениям, использующим методы социальной инженерии [4]. Таким образом, задача своевременного и точного обнаружения фишинга является актуальной и нетривиальной: в потоке легитимной корреспонденции требуется выявлять злонамеренный контент, мимикрирующий под доверенные источники.

Для противодействия фишингу существует ряд традиционных подходов: использование черных списков доменов и IP-адресов, сигнатурный анализ для обнаружения известных вредоносных вложений, а также фильтрация на основе ключевых слов. Многие из этих методов ориентированы на обнаружение уже известных индикаторов компрометации. Однако в условиях, когда злоумышленники активно используют укороченные ссылки, генерируемые домены и, что наиболее важно, лексические уловки (опечатки, омоглифы, неологизмы), статических методов становится недостаточно. Возникает потребность в более гибком подходе,

способном анализировать не только форму, но и семантическое содержание сообщения для выявления скрытых признаков мошенничества.

Одним из перспективных направлений для решения подобных задач является использование методов обработки естественного языка (NLP) и машинного обучения [24]. Технологии NLP уже успешно зарекомендовали себя в смежных областях, таких как классификация спама и анализ тональности текста. Преимущество данных методов заключается в их способности обучаться на больших объемах данных, выявлять неявные закономерности и адаптироваться к новым видам текстовых атак. В контексте обнаружения фишинга это означает возможность построения моделей, которые оценивают текст комплексно, учитывая семантические и стилистические особенности, а не только наличие определенных слов [4].

Ключевым этапом в построении NLP-моделей является векторизация – преобразование текста в числовые представления. Популярные инструменты, такие как TF-IDF или Word2Vec, эффективно работают со словами, которые присутствовали в обучающем словаре. Однако их основное ограничение – проблема слов вне словаря (Out-of-Vocabulary, OOV). Модели на их основе уязвимы к словам с преднамеренными опечатками (например, P@yPal вместо PayPal), которые часто встречаются в фишинге. Для решения этой проблемы была разработана модель FastText, которая обучается на n-граммах символов и способна генерировать векторы даже для ранее неизвестных слов [5]. Несмотря на мощность самого метода, построение эффективного конвейера на его основе остается сложной задачей, требующей ручного подбора десятков гиперпараметров. Этот процесс является узким местом, замедляющим разработку и адаптацию систем защиты.

Цель исследования – разработать концепцию и программную реализацию специализированной библиотеки автоматизированного машинного обучения (AutoML) для построения и оптимизации конвейеров обнаружения фишинга на основе векторизации FastText, а также провести оценку ее эффективности [25].

2. Разбор ключевых ошибок и рисков применения ИИ в IDS.

Первая проблема **«Мусор на входе – мусор на выходе»**. Эффективность любой модели ИИ на 99% зависит от качества данных, на которых она обучена. Именно здесь кроется первая и главная ошибка. Если модель обучали на данных из сети условного банка, она будет бесполезна на промышленных системах управления (industrial control systems, далее – ICS) и системах диспетчерского управления и сбора данных (supervisory control and data acquisition, далее – SCADA) с их специфическим трафиком. Для таких систем не может быть универсальной модели.

В реальных сетях атак – это крошечный процент от всего трафика. Модель, обученная на таких данных, быстро приходит к выводу, что самый простой способ достичь точности в 99,9% – это всегда предсказывать «норму», и пропускает реальные инциденты (ложноотрицательные срабатывания).

Так же проходит дрейф понятий. «Норма» – понятие нестабильное. Сетевая активность меняется с запуском новых сервисов, обновлениями и сменой бизнес-процессов. Модель, не обученная на актуальных данных, быстро устаревает и теряет точность.

Вторым риском является **проблема «черного ящика»**. Современные нейросети, особенно глубокие, часто необъяснимы. Аналитик безопасности получает от системы вердикт: «Угроза с вероятностью 99%», но не получает ответа на главный вопрос – почему?

Из-за этого происходит сложность расследования. Без понимания логики принятия решения невозможно оценить масштаб атаки, ее источник и цель. Аналитик тратит время на обратный разбор работы алгоритма вместо расследования самого инцидента.

Всё это приводит к отсутствию доверия. В безопасности нельзя слепо доверять инструменту. Если система несколько раз ошиблась и не смогла доступно объяснить причину тревоги, такая работа может привести к тому, что аналитики начнут игнорировать ее срабатывания.

Следующим риском являются **ложные срабатывания и «алертная усталость»**. Даже самая точная модель ИИ периодически ошибается, принимая легитимную активность за атаку (ложноположительные срабатывания). Поток таких ложных предупреждений вызывает у команды центра мониторинга и реагирования на инциденты информационной безопасности (Security Operations Center, далее – SOC) «алертную усталость» – состояние, когда из-за постоянного шума специалисты теряют бдительность и могут пропустить реальную, серьезную угрозу. Человеческий мозг не может долго оставаться сосредоточенным на десятках и сотнях ложных инцидентов в день.

Четвертой угрозой являются **целевые атаки на саму модель ИИ**. Это самый опасный сценарий. Злоумышленники знают о существовании систем обнаружения вторжений (intrusion detection system, далее – IDS) управляемых ИИ и атакуют непосредственно ее. К таким атакам относятся:

1. Отравление данных (Data Poisoning) – при такой атаке злоумышленники обманывают ИИ на этапе обучения, размещая в базах данных, на основе которых проводиться обучение, ложные знания. При обучении нейронной сети такими данными она будет отрабатывать так, как нужно злоумышленникам, а инженеры, разрабатывающие сеть, не смогут этого узнать вплоть до реализации атаки. Это происходит из-за того, что данных для обучения используется невероятно много и каждый массив невозможно проверить вручную перед обучением. И если атака отравлением проведена до разделения данных на обучающие и тестовые, то узнать об этом на этапе обучения будет очень сложно. Но использование такой атаки ограничено тем, что необходим непосредственный доступ к данным.
2. Атаки уклонения (Evasion Attack) – это атаки, которые изначально являлись ошибками второго рода, но сейчас под этим понятием находятся любые обманы ИИ. Целю такой атаки является создания иллюзии, которая повлияет на восприятие поступающей информации к ИИ и получить ответ, который ИИ не дал бы при нормальных условиях. Примером такой атаки могут являться способы обойти ограничения при общении с нейронной сетью ChatGPT. Чтобы обойти контент фильтр пользователи просят нейронную сеть гипотетически представить себя на месте какого-либо персонажа и попросить написать информацию, которую чат изначально не выдает по этическим или моральным соображениям разработчиков.
3. Атаки установления принадлежности (Attacks to establish affiliation) – такие атаки направленны на получения конфиденциальных данных, используемых при обучении ИИ. Злоумышленник пытается определить, использовались ли конкретные данные о ком-либо для обучения модели. Примером может стать атака на нейронные сети, которые используются в медицине. Например, если получиться узнать использовались ли данные о конкретном человеке при обучении модели, следящей за передвижениями людей с болезнью Альцгеймера, то это позволяет сделать вывод о заболевании этого человека. Так же можно вытаскивать из ИИ фотографии, используемые для обучения, что тоже является раскрытием конфиденциальных данных.

И последней проблемой ИИ является высокая стоимость и сложность внедрения. Построение эффективной AI-системы – это не «купил коробку и поставил». Это:

1. Ресурсоемкость: обучение сложных моделей требует огромных вычислительных мощностей и времени.
2. Дефицит кадров: для создания и обслуживания такой системы нужна редкая и дорогая комбинация компетенций: эксперты по кибербезопасности + эксперты в больших данных.
3. Сложность интеграции: интеграция ИИ-решения в существующую ИБ-инфраструктуру и процессы – крайне нетривиальная задача.

Так почему же ИИ до сих пор не используется повсеместно в СЗИ? Ответ становится очевидным после разбора ошибок. ИИ не используется как ядро СЗИ, потому что он ненадежен как самостоятельный страж. Риски ложных срабатываний, пропущенных атак и целевого

обхода алгоритма слишком высоки. Но это не значит, что от ИИ нужно отказаться. Это значит, что его правильное место – быть мощным инструментом в руках опытного аналитика.

3. Технические барьеры

Рассмотрим конкретно какие препятствия стоят перед внедрением ИИ. Первым барьером можно назвать **качество и количество данных**. ИИ – это инструмент, полностью зависимый от данных. Для обучения эффективных моделей требуются объемные и репрезентативные датасеты, потому что недостаточно просто собрать логи. Данные должны быть размечены (что является атакой, а что – легитимным трафиком), что требует титанических усилий экспертов-аналитиков.

Так же присутствует **проблема дисбаланса**. В нормальной сетевой активности атаки составляют менее 1% всего трафика. Модель, обученная на таких несбалансированных данных, быстро научится всегда предсказывать «норму», достигая высокую точность, но по факту она будет бесполезная на практике.

И конфиденциальность данных для обучения так же под угрозой, так как данные сетевого трафика и логи содержат чувствительную информацию. Их использование для обучения моделей вне периметра компании (например, в облачных сервисах) вызывает серьезные опасения по поводу конфиденциальности и может противоречить таким регуляторам различным регуляторам [14].

Вторым техническим барьером является проблема ложных срабатываний и ложных пропусков. Хотя ИИ и должен снизить количество ложных срабатываний по сравнению с традиционными системами, незрелые или плохо обученные модели могут иметь обратный эффект. Ошибки 1-ого рода перегружают специалистов центр мониторинга информационной безопасности (security operations center, далее – SOC-аналитиков), приводя к «усталости от предупреждений» и потенциальному игнорированию реальных угроз. А ошибки 2-ого рода хуже, потому что модель может пропустить сложную, ранее не виданную атаку, создав у команды безопасности ложное чувство спокойствия [15].

Ещё одним техническим барьером становится **интерпретируемость и «черный ящик»**. Большинство сложных моделей ИИ (например, глубокие нейронные сети) действуют как «черный ящик». Мы видим входные данные и результат, но не понимаем почему было принято то или иное решение. Для специалиста по безопасности критически важно не просто получить сообщение о потенциальной угрозе, а понять какие именно признаки указывают на атаку и какова тактика и техника злоумышленника. Без пояснений действий ИИ доверие к системе падает, а расследование инцидента сильно затрудняется [16].

И последний технический барьер – это **адаптивность и манипулирование входными данными**. Злоумышленники сами начинают использовать ИИ для создания атак, специально разработанных для обхода ИИ-моделей. Примером это является Состязательное машинное обучение (adversarial machine learning, далее – AML). Путем внесения незаметных для человека изменений во вредоносный код или сетевой пакет (например, изменив всего несколько байт) можно «обмануть» модель, заставив ее классифицировать атаку как легитимную активность. Такие технологии требуют от защищающих моделей постоянной адаптации и переобучения.

4. Организационные и экономические барьеры

Первым таким барьером становится **высокая стоимость внедрения и эксплуатации**. Внедрение ИИ-решений – это не только покупка лицензии. Это инвестиции в мощное аппаратное обеспечение и квалифицированные кадры. Обучение и запуск моделей требуют значительных вычислительных ресурсов, закупка которых может не оправдать затраты на себя, а нехватка компетентных специалистов на стыке кибербезопасности и больших данных – одна из главных проблем. Зарплаты таких экспертов очень высоки. Так же важна поддержка и

дообучение моделей, потому что они не могут работать вечно без обновлений. Их необходимо постоянно дообучать на новых данных, что требует постоянно вкладываемых ресурсов [17].

Следующим барьером является **нехватка экспертизы и интеграция в процессы**. Внедрение ИИ – это организационное изменение. Командам безопасности необходимо пересматривать существующие процессы реагирования на инциденты. Обучать аналитиков работать с новыми инструментами и интерпретировать их выводы. А также интегрировать ИИ-решения с уже существующей системой управления информацией и событиями безопасности (Security Information and Event Management, далее – SIEM), системой оркестрации, автоматизации и реагирования на инциденты безопасности (Security Orchestration, Automation and Response, далее – SOAR) и другими платформами, что может быть технически сложно.

Третьим организационным барьером являются **этические и нормативные барьеры**. Это вопрос ответственности за принятые решения. Кто будет нести ответственность в случае сбоя системы ИИ и пропуска масштабной атаки? Производитель программного обеспечения, компания, внедрившая его, или разработчик алгоритма? И существенный вес имеет **регуляторная неопределенность**. Отсутствие четких правовых рамок, регулирующих применение ИИ в критически важных инфраструктурах, сдерживает его внедрение, особенно в госсекторе и финансах. Этот вопрос на текущий момент уже активно прорабатывается, но всё еще требуется время на отработку регуляторики [18].

5. Пути преодоления барьеров

Несмотря на различные препятствия, отказ от внедрения ИИ – это путь к отставанию в кибергонке вооружений. Злоумышленники уже массово используют ИИ, и это сильное оружие в их руках. Создать равновесную систему безопасности возможно лишь путем использования в ответ тех же современных технологий. Решение лежит в комплексном подходе:

Основным направлением может стать **развитие объясняемого ИИ** (explainable artificial intelligence, далее – XAI): Приоритетом должно являться создание моделей, способных аргументировать свои решения на языке, понятном аналитику. Необходимо сделать ИИ более прозрачным, чтобы пользователи, разработчики и заинтересованные стороны могли проверять, соответствуют ли решения ИИ техническим, правовым и этическим стандартам. XAI – это не одна технология, а набор методов и подходов. Ключевые из них, которые можно применять в системах обнаружения и предотвращения вторжений (intrusion detection and prevention system, далее – IDS/IPS) представлены в таблице 1 [19].

Таблица. 1. Набор методов и подходов XAI, которые можно применять в IDS/IPS.

Метод/подход	Как это работает?	Пример в кибербезопасности
LIME (Local Interpretable Model-agnostic Explanations/ Локальные интерпретируемые модель-агностические объяснения)	Создает упрощенную, понятную модель (например, линейную), которая аппроксимирует поведение сложной модели («черного ящика») для конкретного отдельного предсказания.	Модель ИИ определяет сетевой пакет как вредоносный. LIME показывает: «Это решение на 80% обусловлено необычно длинным полем HTTP-заголовка и на 15% – редкой последовательностью TCP-флагов». Аналитик сразу видит, на что смотреть.
SHAP (SHapley Additive exPlanations/ аддитивные объяснения Шепли)	Основан на теории игр, чтобы справедливо распределить «вклад» каждой входной информации в итоговое решение. Показывает, насколько наличие или отсутствие конкретного признака увеличило или	Модель обнаружила аномалию в поведении пользователя. SHAP-график показывает, что главными «виновниками» стали: вход в систему в нерабочий час (+40% к риску), доступ к серверу, к которому у пользователя нет прав (+35%), ска-

	уменьшило вероятность класса.	чивание большого объема данных (+25%).
Counterfactual Explanations (Контрфактуальные объяснения)	Объяснение через противопоставление. Метод показывает минимальные изменения, которые нужно внести во входные данные, чтобы модель изменила свое решение.	«Этот исполняемый файл был классифицирован как вредоносный. Если бы он не пытался установить постоянство в реестре и не делал сетевых запросов на подозрительный IP-адрес, он был бы признан безопасным».
Attention Mechanisms (Механизмы внимания)	Внутренний механизм некоторых нейросетевых архитектур, который визуально подсвечивает части входных данных (например, слова в тексте лога или участки кода), на которые модель «смотрела» больше всего при принятии решения.	Анализируя командную строку PowerShell, модель уделила основное внимание выделенным фрагментам -EncodedCommand, IEX и base64-строке. Это прямо указывает на попытку запуска закодированного скрипта – классическая техника злоумышленников.

XAI – это не просто «приятная опция», а необходимое условие для создания надежных, доверенных и эффективных систем кибербезопасности на основе искусственного интеллекта. Это мост между raw-вычислительной мощью машин и экспертизой человека, позволяющий им работать в тандеме для отражения самых изощренных кибератак. Без XAI ИИ останется просто интересным экспериментом, но с ним он становится краеугольным камнем обороны нового поколения.

Для преодоления проблем с данными можно использовать методы **федеративного и синтетического обучения**, позволяющие обучать модели на децентрализованных данных без их непосредственного объединения (федеративное обучение), или генерировать синтетические данные для балансировки датасетов [20].

Федеративное обучение решает ключевую проблему конфиденциальности данных. Вместо объединения информации из разных источников в одном месте, модель ИИ отправляется к данным (например, к клиентам). Она обучается локально, а затем отправляет только полученные знания (обновления параметров), а не сами сырые данные, на центральный сервер для агрегации. Это позволяет создавать мощные модели, соблюдая общий регламент по защите данных (General Data Protection Regulation, далее – GDPR) и коммерческую тайну. На сегодняшний день компании, использующие поведенческий анализ трафика для поиска аномалий в сети как раз используют федеративное обучение для тренировки модели на трафике локальном сети для работы далее в этой же сети.

Синтетическое обучение борется с нехваткой и дисбалансом данных об атаках. С помощью таких методов, как генерация синтетических примеров для класса меньшинства (synthetic minority over-sampling technique, далее – SMOTE) или генеративные состязательные сети (generative adversarial network, далее – GAN), создаются искусственные, но правдоподобные данные. Например, генерируются новые варианты вредоносного трафика для расширения датасета. Это позволяет эффективно обучать модели на редких типах угроз и моделировать атаки будущего в безопасных средах.

Вместе эти подходы создают синергию: федеративное обучение собирает знания из разных источников, а синтетическое – обеспечивает сбалансированность и полноту этих данных для обучения, формируя основу для коллективного кибериммунитета.

Так же одним из путей преодоления барьеров безусловно является **гибридный подход**. Нужно не заменять, а усиливать традиционные сигнатурные и поведенческие методы с помощью ИИ. Гибридная система может использовать сильные стороны каждого подхода. Таким способом можно оптимизировать сигнатуры и при помощи ИИ разрабатывать сигнатуры заранее под неизвестные атаки [21].

Оптимизация сигнатур с помощью ИИ может начаться с автоматизации их создания и постоянного обновления. Вместо того чтобы специалистам вручную анализировать каждый

новый образец вредоносного ПО и писать под него правило, искусственный интеллект берет на себя эту тяжелую работу. Он применяет методы машинного обучения для анализа огромных массивов данных о срабатываниях систем безопасности, выявляя скрытые паттерны их поведения и группируя похожие события в кластеры. Например, обнаружив десятки различных файлов, которые выполняют одну и ту же последовательность действий – обращаются к одному управляющему серверу и затем пытаются шифровать файлы, – ИИ автоматически генерирует обобщенную поведенческую сигнатуру на всё это семейство угроз. Это кардинально ускоряет реакцию на новые атаки, превращая рутинный процесс в динамичный и непрерывный цикл самообучения системы.

Еще одно критически важное направление – это **снижение уровня ложных срабатываний**. Традиционные статические сигнатуры часто срабатывают на безобидную легитимную активность, порождая информационный шум и заставляя аналитиков тратить время на расследование несущественных инцидентов. ИИ решает эту проблему за счет интеллектуального контекстного анализа. Он оценивает каждое срабатывание правила не изолированно, а в совокупности с множеством факторов: кто является источником события, в какое время суток оно произошло, что предшествовало ему и что последовало после. На основе этого глубокого анализа система присваивает каждому предупреждению оценку риска, в результате чего аналитик видит не равномерный поток из сотен одинаковых уведомлений, а краткий список из нескольких высокоприоритетных инцидентов, которые с высокой долей вероятности являются реальными угрозами и требуют немедленного вмешательства [22].

Следующий шаг к преодолению барьеров – это **переход от реактивной защиты к проактивной**, то есть создание сигнатур на опережение. ИИ способен не только анализировать уже произошедшие атаки, но и предсказывать будущие. Он обрабатывает данные из открытых источников и разведки угроз: анализирует тренды на хакерских форумах, изучает свежие уязвимости, попавшие в общий доступ, и отслеживает новые тактики злоумышленников. На основе этих прогнозов система может генерировать гипотетические правила, предназначенные для обнаружения атак, которые еще только готовятся или находятся на ранней стадии разработки. Таким образом, безопасность перестает быть догоняющей и начинает работать на опережение, заранее выстраивая барьеры на пути вероятных угроз [23].

И также, ИИ способен внести неоценимый вклад в оптимизацию производительности всей системы безопасности. С течением времени база сигтур разрастается до десятков тысяч правил, многие из которых устарели, дублируют друг друга или крайне редко срабатывают, бесполезно потребляя вычислительные ресурсы и замедляя работу. ИИ действует как высококвалифицированный архитектор производительности: он постоянно мониторит эффективность каждого правила, определяя, какие из них действительно ловят злоумышленников, а какие лишь создают шум. На основе этого анализа система может автоматически рекомендовать отключить, объединить или упростить неэффективные правила, что приводит к значительной разгрузке системы без малейшей потери в качестве защиты, обеспечивая ее высокую скорость и отказоустойчивость. Важный пункт в том, что ИИ не сам отключает правила, а именно рекомендует это специалисту, который может оценить рассуждения и доводы по оптимизации, а потом на основе этого принять решение.

Немаловажным для преодоления барьеров являются **инвестиции в образование и автоматизацию**: Компаниям необходимо готовить или привлекать новых специалистов, а вендорам – создавать более дружелюбные и автоматизированные решения, скрывающие сложность фундаментальных моделей машинного обучения (machine learning, далее – ML-моделей) от конечного пользователя [21].

Столь же важным, как образование является **развитие методология машинного обучения** (machine learning operations, далее – MLOps): Внедрение практик MLOps позволяет автоматизировать жизненный цикл ML-моделей (развертывание, мониторинг дрейфа данных, переобучение), сделав их поддержку более эффективной. Такой подход позволит сосредоточить ресурсы и время компаний на оптимизации самих моделей и подготовку данных для их обучения.

Так же методом преодоления барьеров может стать **внедрение методологии активного обучения** (Active Learning). Этот подход кардинально меняет парадигму подготовки данных для ИИ-моделей. Вместо того чтобы аналитики вручную размечали огромные массивы данных, модель сама идентифицирует наиболее ценные для своего обучения примеры – те, в которых она менее всего уверена или которые находятся на границе принятия решений. Таким образом, эксперты-аналитики фокусируют свои усилия на разметке только этих ключевых данных, что значительно экономит время и ресурсы, позволяет быстрее «доучивать» модель на редких типах атак и эффективно бороться с дисбалансом классов в данных, делая процесс обучения более целенаправленным и рентабельным [22].

Следующим важным направлением является **применение ансамблей моделей** (Ensemble Learning) и проактивное использование ИИ для охоты за угрозами (Threat Hunting). Вместо опоры на одну сложную модель «черного ящика» используется комитет более простых и интерпретируемых алгоритмов. Их совокупное решение оказывается более точным и устойчивым к целенаправленным атакам обхода, поскольку злоумышленнику нужно скомпрометировать сразу несколько разнородных моделей. Кроме того, ИИ может выступать не в роли автоматического сторожа, а в роли интеллектуального помощника для специалистов по кибербезопасности. Анализируя данные, система может генерировать гипотезы о потенциальных скрытых угрозах, например, указывая аналитику на группу хостов, чье поведение статистически отклоняется от нормы по определенным параметрам. Это позволяет перейти от реактивного обнаружения к проактивному расследованию, где ИИ расширяет возможности человека-эксперта.

Для повышения устойчивости и доверия к системам критически важна **организация непрерывной валидации моделей** с помощью практик, аналогичных «красным командам», но на основе ИИ. Речь идет о создании внутренних процессов, при которых специальные инструменты или команды постоянно пытаются атаковать и обмануть развернутые защищающие модели, используя методы состязательного машинного обучения. Такой подход позволяет выявлять и устранять уязвимости в алгоритмах до того, как ими воспользуются реальные злоумышленники. Регулярное успешное прохождение таких испытаний значительно повышает доверие аналитиков к системе, так как ее надежность подвергается постоянной проверке в контролируемых, но близких к реальным условиях [23].

С организационной и правовой точек зрения преодолению барьеров способствует **разработка четких моделей ответственности и развитие сертификации ИИ-решений**. Компаниям и регуляторам необходимо создать прозрачные рамки, определяющие, кто несет ответственность в случае сбоя ИИ – разработчик алгоритма, интегратор системы или конечный пользователь. Параллельно развитие независимой сертификации ИИ-моделей для критической инфраструктуры по аналогии с другими стандартами безопасности позволит снизить юридические риски и стимулировать внедрение в консервативных отраслях, таких как финансы и госсектор, поскольку наличие сертификата будет свидетельствовать о соответствии модели установленным стандартам надежности и этики.

С точки зрения архитектуры эффективным путем является **отказ от поиска универсального инструмента в пользу принципа глубокой эшелонированной обороны с интегрированными ИИ-модулями на каждом уровне**. Это означает внедрение не одной мегамодели, отвечающей за все, а набора узкоспециализированных и более простых моделей: одна для анализа сетевого трафика, другая для поведения пользователей, третья – для оконечных хостов. Такой подход снижает общую сложность системы, делает ее более устойчивой, поскольку компрометация одного модуля не выводит из строя всю защиту, и упрощает обслуживание и объяснимость решений каждого отдельного компонента. Дополняет эту архитектуру использование периферийного искусственного интеллекта, когда анализ данных и принятие решений происходят непосредственно на шлюзах или защищаемых узлах, а не в центральном облаке. Это решает проблемы конфиденциальности, так как сырые данные не покидают периметр, снижает задержки реакции до реального времени, что критично для промышленных систем, и разгружает сетевые каналы от передачи всей телеметрии [20].

Наконец, для ускорения развития коллективного кибериммунитета перспективным выглядит **создание специализированных платформ и сообществ для обмена не данными, а уже обученными моделями или их компонентами**. По аналогии с обменом индикаторами компрометации, компании и вендоры могли бы обмениваться предобученными моделями, прошедшими валидацию. Это позволило бы организациям, особенно с небольшим объемом собственных данных, быстро развертывать эффективные системы, беря за основу модель, обученную на агрегированном опыте множества участников сообщества, и дообучая ее под свою специфику, тем самым коллективно повышая уровень защиты против новейших угроз.

6. Концепция и разработка методики

Ключевой задачей при применении машинного обучения к текстовым данным является их преобразование в числовой формат, или векторизация. Классические методы, такие как «мешок слов» (Bag-of-Words) или TF-IDF, представляют текст в виде разреженных векторов высокой размерности, где каждая составляющая соответствует определенному слову из словаря [6]. Но как бы просты не были эти методы, они обладают двумя существенными ограничениями: не учитывает семантическую близость схожих по смыслу слов (например, слова «атака» и «нападение» для них являются полностью различными) и не способны работать со словами, отсутствующими в обучающем словаре (проблема Out-of-Vocabulary, OOV) [7]. Последнее ограничение значительно снижает успех обнаружения фишинга, что часто используется злоумышленниками посредством намеренной модификации слов.

Чтобы ликвидировать данный недостаток, были разработаны методы, формирующие плотные векторные представления (embeddings), которые отображают слова в многомерное семантическое пространство, где близкие по смыслу слова имеют близкие векторы [8]. Одним из таких методов является модель FastText, предложенная исследователями из Facebook AI Research. Тогда как предшествующие модели, такие как Word2Vec, работали со словами как с отдельными единицами, FastText представляет каждое слово как совокупность символьных n-грамм. Например, для слова атака и n=3 будут сгенерированы триграммы <ат, ата, так, ака, ка>, где < и > – специальные символы начала и конца слова. Вектор слова в модели FastText вычисляется как сумма векторов всех его символьных n-грамм, что описывается формулой (1)[9]:

$$v(w) = \sum_{g} G_w Z_g , \tag{1}$$

где:

$v(w)$ – итоговый вектор слова w;

G_w – множество всех символьных n-грамм, составляющих слово w;

Z_g – вектор, соответствующий n-грамме g, который обучается в ходе работы алгоритма.

Благодаря такому подходу FastText способен генерировать осмысленные векторы даже для слов, которые отсутствовали в словаре или были модифицированы, так как они с высокой вероятностью будут состоять из уже известных модели n-грамм. Это свойство делает FastText более эффективным для анализа фишинговых текстов, богатых лексическими искажениями.

Однако наличие эффективного метода векторизации не гарантирует необходимой производительности всей системы обнаружения. На конечный результат влияет множество взаимосвязанных параметров, формирующих конвейер обработки данных (pipeline): от параметров предобработки текста (например, необходимость лемматизации или удаления стоп-слов) до гиперпараметров самой модели FastText (размерность векторов, размер контекстного окна, количество эпох обучения) и последующего классификатора (например, параметр регуляризации C для логистической регрессии) [6]. Ручной подбор этих параметров метода-

ми поиска по сетке (Grid Search) или случайного поиска (Random Search) является трудоемкой и долгой задачей, требующей экспоненциального роста вычислений при увеличении числа параметров.

Чтобы избавиться от этой задачи, в данной работе предлагается концепция специализированной библиотеки автоматизированного машинного обучения (AutoML) PhishAutoML. AutoML представляет собой парадигму автоматизации всего процесса применения машинного обучения, от подготовки данных до развертывания моделей [25, 26]. В основе нашей библиотеки лежит метод байесовской оптимизации, который позволяет находить наилучший набор в пространстве гиперпараметров, минимизируя количество необходимых вычислений [10]. Алгоритм итеративно строит вероятностную суррогатную модель (чаще всего гауссовский процесс) зависимости качества конвейера от его параметров и использует функцию приобретения (acquisition function) для выбора следующей, наиболее перспективной комбинации параметров для проверки [11].

Ключевым элементом предложенной концепции является комплексная целевая функция, которая позволяет получить высокий показатель не только с точки зрения точности, но и производительности. Данная функция формализуется следующим образом, согласно уравнению (2)[12]:

$$O = (1-\lambda)(1-F1) + \lambda * T_{inv}, \quad (2)$$

где:

O– значение целевой функции, которое минимизируется оптимизатором;

F1 – F1-мера, метрика, гармонически усредняющая точность (precision) и полноту (recall) классификатора;

T_{inv}– инвертированная пропускная способность (время обработки одного сообщения), характеризующая производительность модели;

λ – коэффициент компромисса (от 0 до 1), задаваемый пользователем.

При λ=0 оптимизация направлена исключительно на максимизацию F1-меры, а при λ=1 – на минимизацию времени обработки.

Таким образом, предложенная методика объединяет семантическую мощность модели FastText для устойчивого анализа текста и гибкость AutoML-подхода для автоматического построения и настройки всего конвейера обнаружения фишинга под конкретные практические задачи. Вся архитектура реализована в виде программной библиотеки на языке Python с использованием фреймворков scikit-learn для построения конвейеров, scikit-optimize для реализации байесовской оптимизации и официальной библиотеки fasttext [27].

7. Процесс автоматической оптимизации конвейера

В создаваемой библиотеке PhishAutoML процесс автоматического подбора гиперпараметров представляет собой повторяющуюся процедуру, которая управляется алгоритмом байесовской оптимизации. В отличие от методов полного или случайного перебора, байесовская оптимизация является наиболее эффективным подходом для такого типа задач, где оценка целевой функции является вычислительно дорогостоящей и ресурсоемкой операцией. Для работы алгоритма заранее выбирается количество итераций (n_iter), и на каждом шаге t последовательно выполняются четко выделенные инструкции.

Выбор следующей комбинации гиперпараметров.

На основе истории уже проведенных испытаний Dt={(h1,O1),...,(ht,Ot)}, где hi – вектор гиперпараметров, а Oi – соответствующее ему значение целевой функции, строится вероятностная суррогатная модель. В качестве такой модели выступает, как правило, гауссовский процесс, который, помимо предсказания ожидаемого значения целевой функции для любой новой точки в пространстве параметров, оценивает неопределенность этого предсказаниях [13].

Далее используется функция приобретения α (acquisition function), используемая для выбора следующей точки для вычислений. Эта функция формализует компромисс между исследованием (exploration) – проверкой малоизученных областей пространства с высокой неопределенностью, и эксплуатацией (exploitation) – уточнением вблизи уже известных хороших решений. В нашем случае используется одна из стандартных функций приобретения, Ожидаемое Улучшение (Expected Improvement). Она подразумевает, что следующий шаг выбора комбинации гиперпараметров ht+1 описывается как поиск аргумента, который максимизирует функцию приобретения:

$$h_{\tau+1} = \arg\max_h \alpha(h \mid D_\tau), \tag{3}$$

где:

$h_{\tau+1}$ – вектор гиперпараметров, выбираемый для тестирования на следующей итерации.

$\alpha(h \mid D_\tau)$ – значение функции приобретения, рассчитанное на основе уже имеющихся данных D_τ.

Построение и обучение конвейера.

С использованием выбранного набора гиперпараметров динамически собирается экземпляр конвейера scikit-learn.Pipeline. Данный конвейер является последовательностью трансформеров и оценщика, и состоит из трех настраиваемых этапов: предобработку текста (DataPreprocessor), векторизацию (FastTextVectorizer) и классификацию (LogisticRegression). Каждый компонент инициализируется с параметрами из вектора ht+1 [13].

Оценка эффективности.

Сконструированный конвейер оценивается на полном обучающем наборе данных (X_train, y_train) с помощью процедуры стратифицированной k-кратной кросс-валидации (в данной работе k=3). Метод позволяет получить робастную оценку обобщающей способности модели, при этом снижая риск переобучения на конкретное разбиение данных. Использование стратификации является критически важным, так как оно гарантирует сохранение исходного соотношения классов («фишинг» / «легитимное») в каждой из k подвыборок. Данный подход позволяет предотвратить ситуации, когда в одну из текстовых подвыборок могут попасть примеры только одного типа, что сделало бы расчет некоторых метрик невыполнимым.

Для каждого из k разбиений вычисляется F1-мера. Итоговое значение $F1_{avg}$усредняется по всем разбиениям:

$$F1_{avg} = \frac{1}{k}\sum_{i=1}^{k} F1_i, \tag{4}$$

где:

$F1_i$ – значение F1-меры, полученное на i-м разбиении.

Сама F1-мера является гармоническим средним точности (Precision) и полноты (Recall) и вычисляется по формуле:

$$F1 = 2 * \frac{\mathrm{Pr}ecision - \mathrm{Re}call}{P\,\mathrm{r}ecision + \mathrm{Re}call}, \tag{5}$$

Аналогичным образом рассчитывается средняя инвертированная пропускная способность Tinv. Полученные усредненные значения F1avg и Tinv подставляются в комплексную целевую функцию (2) для вычисления итоговой оценки Ot+1 для данной комбинации гиперпараметров [13].

Обновление суррогатной модели.

Полученная на шаге 3 пара «вектор гиперпараметров – значение целевой функции» (ht+1,Ot+1) добавляется в историю испытаний, формируя новый набор данных для следующей итерации:

$$D_{\tau+1} = D_\tau \cup. \tag{6}$$

На основе этого расширенного набора данных $D_{\tau+1}$ происходит переобучение суррогатной модели. Таким образом, с каждой итерацией «опыт» оптимизатора о структуре целевой функции становятся все точнее.

Цикл оптимизации (шаги 1-4) повторяется n_iter раз. По завершении всех итераций определяется комбинация гиперпараметров h∗, которая показала наилучшее (минимальное) значение целевой функции за весь процесс поиска. Финальным результатом является созданный итоговый экземпляр конвеера с найденными лучшими гиперпараметрами h∗, который обучается единожды на всем объеме обучающих данных. Затем этот обученный экземпляр сохраняется в отдельный файл и может быть использован для дальнейшего обучения на новом наборе данных или для тестирования системы [13].

8. Сравнительный анализ эффективности предложенного решения

Чтобы исследование было объективным, предложенное решение было сравнено с двумя другими моделями разного уровня: классической (TF-IDF + логистическая регрессия) и современной (доообученный трансформер DistilBERT). Все три подхода были обучены на одинаковом наборе данных phishing_corpus.csv, а итоговое тестирование производилось на независимом тестовом наборе phishing_legit_dataset_KD_10000.csv, содержащем 10 000 примеров (см. табл. 2).

Таблица. 2. Сравнительная таблица используемых моделей МО

Модель	F1-score (Фишинг)	Точность (Precision)	Полнота (Recall)	Время обучения	Скорость предсказания (сообщ/сек)
PhishAutoML + FastText	0.94	0.93	0.95	~4 часа	~950 сообщ/сек
TF-IDF + LogReg	0.88	1.00	0.78	~5 минут	~1200 сообщ/сек
Fine-tuned DistilBERT	0.93	0.99	0.88	~20 часов	~20 сообщ/сек

Классическая модель, основанная на векторизации TF-IDF, продемонстрировала самую низкую общую эффективность. Ключевой особенностью ее работы является идеальная точность (Precision = 1.00) для класса «Фишинг». Это означает, что, если модель помечает письмо как фишинговое, она не ошибается. Однако такой результат достигается за счет крайне низкой полноты (Recall = 0.78).

Анализ матрицы ошибок показывает, что модель не допустила ни одного ложноположительного срабатывания, но при этом пропустила 1314 фишинговых атак из 6000. С точки зрения информационной безопасности, такой высокий уровень пропущенных угроз является неприемлемым, так как основная задача системы – минимизировать риски для пользователя. Таким образом, несмотря на высокую скорость работы, данный подход не может считаться надежным.

Модель на основе дообученной архитектуры DistilBERT показала значительно более высокое качество по сравнению с TF-IDF, достигнув общей точности в 91.97%. Точность обнаружения фишинга также оказалась очень высокой, что свидетельствует о малом количестве ложных тревог.

Однако главным недостатком данного подхода является сравнительно невысокая полнота. Модель пропустила 743 фишинговые атаки, что, хотя и лучше, чем у TF-IDF, все еще является значимым риском. Кроме того, как показывают замеры, производительность трансформерной архитектуры крайне низка (~20 сообщ/сек), что делает ее практическое применение в высоконагруженных системах фильтрации электронной почты затруднительным и ресурсоемким.

Разработанная модель PhishAutoML на основе FastText и автоматической оптимизации продемонстрировала наиболее сбалансированные и практически ценные результаты. До-

стигнув общей точности в 92.30% и F1-меры для класса «Фишинг» в 94%, она превосходит классический подход и не уступает по качеству современной трансформерной модели.

Также полученные результаты демонстрируют, что использование PhishAutoML модели способно частично компенсировать ограничения внедрения искусственного интеллекта в системы безопасности как корпоративных сетей, так и при персональном использовании модели. Автоматизация оптимизации гиперпараметров снижает технические барьеры, а высокая полнота обнаружения (Recall = 0,95) повышает доверие специалистов информационной безопасности, уменьшая количество ложных тревог.

Ключевым преимуществом предложенного решения является самый высокий показатель полноты среди всех исследуемых моделей. Анализ матрицы ошибок показывает, что модель пропустила всего 303 фишинговые атаки. Это в 4.3 раза меньше, чем у TF-IDF, и в 2.5 раза меньше, чем у DistilBERT.

Да, достижение такой высокой полноты сопряжено с компромиссом в виде увеличения числа ложноположительных срабатываний. Однако в задачах обеспечения безопасности пропуск реальной угрозы (False Negative) является значительно более критичной ошибкой, чем ложная тревога (False Positive). Таким образом, предложенная модель делает наиболее адекватный и безопасный для конечного пользователя компромисс. В сочетании с высокой производительностью это доказывает практическую ценность и эффективность разработанного подхода.

9. Заключение

Проведенный анализ в этой статье показывает, что внедрение искусственного интеллекта в современные системы обнаружения и предотвращения вторжений сопряжено с комплексом взаимосвязанных барьеров. Технические вызовы, такие как проблема «черного ящика», качество данных и уязвимость к состязательным атакам, тесно переплетаются с организационно-экономическими трудностями – высокой стоимостью, дефицитом кадров и нормативной неопределенностью. Эти препятствия делают невозможным рассмотрение ИИ в качестве панацеи или полностью автономного «стража» кибербезопасности.

Для разработанной методики оценки эффективности автоматического конвейера обнаружения фишинга на основе FastText в рамках специализированной библиотеки присутствуют следующие ограничения:

– неполная семантическая обработка в FastText влияющая на структурную схожесть и легитимность;
– прямая зависимость от качества данных;
– ограниченная гибкость в дообучении AutoMl;
– недостаток прозрачности и контроля в AutoMl;
– существующие риски в переобучении и дрейфе в AutoMl с зависимостью от размерности данных;
– наличие класс-имбаланса в задачах обнаружения фишинга, что может снижать recall;
– методика улучшает характеристики Precision в AutoMl, но не гарантирует защиту от состязательных атак.

Однако отказ от интеграции ИИ равносилен добровольному отказу от участия в кибергонке вооружений, где злоумышленники уже активно используют эти технологии. Успешное преодоление барьеров лежит не в поиске универсального решения, а в стратегическом, комплексном подходе. Ключ к успеху – создание гибридных, объяснимых и человекоцентричных систем безопасности. В такой парадигме искусственный интеллект выступает не в роли замены эксперта-аналитика, а в качестве его мощного ассистента, усиливающего аналитические способности, автоматизирующего рутинные операции и обеспечивающего проактивную защиту.

Перспективы развития (методы ведения которого описаны в статье) связаны с активным внедрением методологии Explainable AI (XAI) для повышения доверия, использованием федеративного и синтетического обучения для решения проблем с данными, а также с переходом к архитектуре глубокой эшелонированной обороны с распределенными ИИ-модулями. Столь же важны и организационные изменения: развитие практик MLOps, инвестиции в образование и формирование четких правовых рамок.

Таким образом, будущее систем IDS/IPS видится не в замене человека алгоритмом, а в их эффективном симбиозе. Преодоление существующих барьеров – это коллективная и непрерывная задача для исследователей, вендоров и практиков информационной безопасности, от решения которой зависит построение устойчивого киберпространства в условиях стремительно эволюционирующих угроз.

В рамках исследования была исследована актуальная научно-практическая задача повышения эффективности обнаружения фишинговых атак путем применения методов автоматизированного машинного обучения. Была разработана и экспериментально проверена концепция программной библиотеки PhishAutoML, которая применятся для автоматического построения и оптимизации конвейеров обработки текстовых данных.

Центральным элементом предложенного подхода является комбинация векторизации текста с помощью модели FastText, устойчивой к лексическим искажениям, и байесовской оптимизации для интеллектуального подбора гиперпараметров всего конвейера. В ходе работы была предложена и реализована комплексная целевая функция, позволяющая гибко управлять компромиссом между качеством обнаружения угроз и производительностью модели, что является важным требованием для практического внедрения.

Были выявлены преимущества и недостатки предложенного решения. Модель, которая была сгенерирована PhishAutoML, не только показала высокое качество в сравнении с методом на основе TF-IDF, но и показала себя как более эффективное решение в сравнении с современной моделью-трансформером DistilBERT. Основным результатом стало достижение наивысшего показателя полноты обнаружения (Recall = 95%), что свидетельствует о минимальном количестве пропущенных угроз. Таким образом, можно утверждать, что автоматизированный подбор гиперпараметров для конвейера на основе FastText позволяет создавать более сбалансированные и надежные модели, что делает PhishAutoML важным шагом на пути к преодолению барьеров внедрения искусственного интеллекта в системы безопасности.

9.1. Перспективы дальнейшего развития

Несмотря на достигнутые результаты, данное исследование открывает ряд перспективных направлений для дальнейшей модернизации:

1. В текущей реализации AutoML-оптимизация охватывает ключевые параметры конвейера. В будущем целесообразно расширить пространство поиска, включив в него выбор типа классификатора (например, градиентный бустинг, SVM), а также более тонкие параметры предобработки текста и самой модели FastText.
2. Исследование можно расширить, добавив в AutoML-конвейер возможность выбора между различными методами векторизации, такими как Word2Vec, GloVe или даже более «легкими» версиями трансформерных моделей.
3. Как показал анализ, исключительно текстовые модели уязвимы к атакам, не имеющим явных лексических аномалий. Перспективным направлением является обогащение вектора признаков дополнительными метаданными: информацией об отправителе, структурными характеристиками URL-адресов и данными о репутации доменов.

Литература

1. *Петренко С. А.*, *Ступин Д. Д.* Национальная система раннего предупреждения о компьютерном нападении / под ред. С. Ф. Боева. 2-е изд. Иннополис: Афина, 2018. 448 с.
2. *Лаврова Д. С.* Методологическое и математическое обеспечение для SIEM-систем в интернете вещей: дис. ... канд. техн. наук: 05.13.19. СПб., 2016. 179 с.
3. *Уорр К.* Надежность нейронных сетей: укрепляем устойчивость ИИ к обману. СПб.: Питер, 2021. 272 с.
4. *Корниенко А. А., Никитин А. Б., Диасамидзе С. В., Кузьменкова Е. Ю.* Моделирование компьютерных атак на распределенную информационную систему // Известия Петербургского университета путей сообщения. 2018. Т. 15. № 4. С. 613–628.
5. *Петренко С. А.* О разработке безопасного частного облака // Международный форум KAZAN DIGITAL WEEK – 2020. Казань, 21–24 сентября 2020 г. Ч. 2. Казань: Научный центр безопасности жизнедеятельности, 2020. С. 63–70.
6. *Лаврова Д. С., Зегжда Д. П., Зайцева Е. А.* Моделирование сетевой инфраструктуры сложных объектов для решения задачи противодействия кибератакам // Вопросы кибербезопасности. 2019. №2(30) С. 13–20.
7. *Зегжда Д. П., Лаврова Д. С., Павленко Е. Ю.* Управление динамической инфраструктурой сложных систем в условиях целенаправленных кибератак // Известия РАН. Теория и системы управления, 2020, № 3, с. 50–63.
8. *Котенко И. В., Паращук И. Б.* Модель системы управления информацией и событиями безопасности // Вестник Астраханского государственного технического университета. Серия: Управление, вычислительная техника и информатика. 2020. № 2. С. 84–94.
9. *Корниенко А. А., Корниенко С. В., Карпов Д. Ю.* Методика детализированного оценивания значимости систем организации движения поездов как объектов критической информационной инфраструктуры // Бюллетень результатов научных исследований. 2020. № 2. С. 5–19.
10. *Данилин Г. В., Соколов С. С., Ныpков А. П., Кныш Т. П.* Мультисервисные сети: методы повышения защищенности данных в условиях сетевых атак // XXI век: итоги прошлого и проблемы настоящего плюс. 2020. Т. 9. № 2 (50). С. 158–163.
11. *Саенко И. Б., Котенко И. В., Аль-Барри М. Х.* Применение искусственных нейронных сетей для выявления аномального поведения пользователей центров обработки данных // Вопросы кибербезопасности. 2022. № 2 (48). С. 87–97.
12. *Зеленин Д. Д.* Разработка модели управления рисками для корпоративной сети, атакуемой вредоносным программным обеспечением типа сетевой червь // Вестник Воронежского института высоких технологий. 2020. № 2 (33). С. 137–142.
13. *Семыкина Н. А.* Математическая модель защиты в информационной системе иерархической структуры // Южно-Сибирский научный вестник. 2021. № 4 (38). С. 75–80.
14. *Грызунов В. В.* Адаптивное управление доступностью ресурсов геоинформационной системы критического применения в условиях деструктивных воздействий : дис. ... док. тех. наук : 2.3.6. СПб, 2022. 395 с.
15. *Липатников В. А.* Распознавание вторжений нарушителя при управлении кибербезопасностью инфраструктуры интегрированной организации на основе нейро-нечетких сетей и когнитивного моделирования // Актуальные проблемы инфотелекоммуникаций в науке и образовании : материалы конф. – Санкт-Петербург : СПбГУТ, 2018.
16. *Ахо А. В., Лам М. С., Сети Р., Ульман Д. Д.* Компиляторы: принципы, технологии и инструменты. 2-е изд. М.: Вильямс, 2008. 1184 с.

17. *Виткова Л. А.* Модель вредоносной информации и ее распространителя в социальных сетях // Защита информации. Инсайд. 2020. № 3(93). С. 66–72.
18. *Красов А. В.* Обеспечение безопасности передачи multicast-трафика в IP-сетях // Защита информации. Инсайд. 2017. № 3(75). С. 34–42.
19. *Сахаров Д. В.* Использование математических методов прогнозирования для оценки нагрузки на вычислительную мощность IoT-сети // Вестник Санкт-Петербургского университета Государственной противопожарной службы МЧС России. 2020. № 2. С. 86–94.
20. *Абрамова Т. В.* Обнаружение аномалий и нейтрализация угроз в распределенных автоматизированных системах управления на основе мониторинга сетевых информационных потоков : дис. ... канд. тех. наук : 2.3.6. Оренбург, 2024. 235 с.
21. *Шелухин О. И.* Обнаружение аномальных выбросов телекоммуникационного трафика методами дискретного вейвлет-анализа // Электромагнитные волны и электронные системы. 2012. Т. 17. №. 2. С. 15–26.
22. *Шелухин О. И.* Интеллектуальные технологии информационной безопасности: учебное пособие для вузов. М.: Горячая линия – Телеком, 2023. 384 с.
23. *Зегжда Д. П.* Кибербезопасность цифровой индустрии. Теория и практика функциональной устойчивости к кибератакам. М.: Горячая линия – Телеком, 2022. 560 с.
24. *Alaloye H. H.* Utilizing NLP to Optimize Municipal Services Delivery Using a Novel Municipal Arabic Dataset // International Journal of Advanced Computer Science and Applications. 2025. Vol. 16, No. 2.
25. Программа взаимодействия с библиотекой AutoML для тестирования нейронных сетей на уязвимости : свид. о гос. регистрации программы для ЭВМ № 2025617273 Рос. Федерация / Штеренберг С. И., заявл. 10.03.2025; опубл. 25.03.2025.
26. *Соболевский В. А.* Использование технологий Automl для автоматического анализа временных рядов // Авиакосмическое приборостроение. 2024. № 10. С. 10–18.
27. *Загагова А. М.* Использование push-уведомлений в фишинг-атаках // Информационные технологии в современном мире: материалы конф. Екатеринбург: Гуманитарный университет, 2025.

Штеренберг Станислав Игоревич

к.т.н., доцент кафедры защищенных систем связи, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича (СПбГУТ, 193232, Санкт-Петербург, пр. Большевиков д.22, к.1), тел. +7 931 337 98 61, e-mail: stas.shterenberg.89@mail.ru, ORCID ID: 0000-0002-4216-6370.

Гречухин Дмитрий Николаевич

студент института магистратуры, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича (СПбГУТ, 193232, Санкт-Петербург, пр. Большевиков д.22, к.1), тел. +7 (812) 305-12-55, доб. 1255, e-mail: darkokos1@mail.ru.

Кривец Андрей Сергеевич

студент института магистратуры, Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича (СПбГУТ, 193232, Санкт-Петербург, пр. Большевиков д.22, к.1), тел. +7 (812) 305-12-55, доб. 1255, e-mail: krivets_2002@mail.ru.

Авторы прочитали и одобрили окончательный вариант рукописи.

Авторы заявляют об отсутствии конфликта интересов.

Вклад соавторов: *Каждый автор внес равную долю участия как во все этапы проводимого теоретического исследования, так и при написании разделов данной статьи.*

A methodology for evaluating the effectiveness of an automated phishing detection pipeline based on FastText within the framework of a specialized AutoML library for analyzing barriers to the introduction of artificial intelligence into modern intrusion detection systems

Stanislav I. Shterenberg, Dmitry N. Grechukhin, Andrey S. Krivets

St. Petersburg State University of Telecommunications named after
Prof. M. A. Bonch-Bruyevich (SPbSUT)

Abstract: The article presents an approach to phishing attack detection based on the construction and automatic optimization of machine learning pipelines using a specialized library (PhishAutoML). The urgency of the problem is due to the evolution of phishing attacks that use social engineering methods and lexical tricks, which makes traditional static protection methods ineffective. The theoretical foundations of text vectorization using the FastText model and its application within an AutoML approach based on Bayesian optimization, which allows for the automatic selection of hyperparameters for the entire pipeline, are described. The proposed PhishAutoML concept is used to build models capable of detecting phishing based on semantic analysis and flexibly configuring the trade-off between quality and performance. The results of computational experiments are presented: final metrics of quality and performance, as well as a comparative analysis with classical (TF-IDF) and modern (DistilBERT) approaches. The conclusions confirm the effectiveness of the proposed solution (achieving a phishing detection recall of 95%, which is several times higher than alternative methods) and outline directions for its further development. However, integrating AI into traditional intrusion detection and prevention systems poses significant risks and challenges. This article explores the key technical, organizational, and ethical barriers that hinder the widespread adoption of AI-powered solutions and suggests potential solutions to overcome them.

Keywords: Artificial Intelligence, Intrusion Detection, Intrusion Prevention, natural language processing (NLP), machine learning, AutoML, FastText, Bayesian optimization.

For citation: Shterenberg S. I., Grechukhin D. N., Krivets A. S. A methodology for evaluating the effectiveness of an automated phishing detection pipeline based on FastText within the framework of a specialized AutoML library for analyzing barriers to the introduction of artificial intelligence into modern intrusion detection systems [Paper Preparation Manual for Vestnik SibGUTI]. *Vestnik SibGUTI*, 2026, vol. 20, no.1, pp. 3–22. https://doi.org/10.55648/1998-6920-2026-20-1-3-22.

The article was submitted: 04.12.2025;
revised version: 15.12.2025;
accepted for publication 05.02.2026.

References

1. Petrenko S. A., Stupin D. D. *Nacional'naya sistema rannego preduprezhdeniya o komp'yuternom napadenii: nauchnaya monografiya.* [National Early Warning System for Computer Attacks: A Scientific Monograph]. Afina, 2018. 448 p.
2. Lavrova D. S. *Metodologicheskoe i matematicheskoe obespechenie dlya SIEM-sistem v internete veshchej.* [Methodological and Mathematical Support for SIEM Systems in the Internet of Things]. Ph. D. thesis. Saint Petersburg, 2016, 179 p.
3. Uorr K. *Nadezhnost' nejronnyh setej: ukreplyaem ustojchivost' II k obmanu* [The reliability of neural networks: strengthening AI's resistance to deception]. Spb, Piter, 2021. 272 p.
4. Kornienko A. A. Modelirovanie komp'yuternyh atak na raspredelennuyu informa-cionnuyu sistemu [Modeling of Computer Attacks on a Distributed Information System]. *Izvestiya Peterburgskogo universiteta putej soobshcheniya,* 2018, vol. 15, no. 4, pp. 613–628.

5. Petrenko S. A. O razrabotke bezopasnogo chastnogo oblaka [About developing a secure private cloud]: *Sbornik materialov mezhdunarodnogo foruma*, Kazan', Akademiya nauk Respubliki Tatarstan. 2020, vol. 2, pp. 63–70.
6. Lavrova D. S., Zegzhda D. P., Zaitseva E. A. Modelirovanie setevoj infrastruktury slozhnyh ob′ektov dlya re-sheniya zadachi protivodejstviya kiberatakam [Modeling the network infrastructure of complex objects to solve the problem of countering cyberattacks]. *Voprosy kiberbezopasnosti*, 2019. no. 2, pp. 13–20.
7. Zegzhda D. P., Lavrova D. S., Pavlenko E. Yu. Upravlenie dinamicheskoj infrastrukturoj slozhnyh sistem v uslo-viyah celenapravlennyh kiberatak [Managing the dynamic infrastructure of complex systems in the face of targeted cyberattacks]. Izvestiya RAN. *Teoriya i sistemy upravleniya*, 2020, no. 3, pp. 50–63.
8. Kotenko I. V., Parashchuk I. B. Model' sistemy upravleniya informaciej i sobytiyami bezopasno-sti [Security Information and Event Management System model]. *Vestnik Astrahanskogo gosudarstvennogo tekhnicheskogo universiteta*, 2020. no. 2. pp. 84–94.
9. Kornienko A. A., Kornienko S. V., Karpov D. Yu. Metodika detalizirovannogo ocenivaniya znachimosti sistem orga-nizacii dvizheniya poezdov kak ob′ektov kriticheskoj informacionnoj infrastruktury [Methodology for detailed assessment of the significance of train traffic management systems as critical information infrastructure objects] *Byulleten' rezul'tatov nauchnyh issledovanij*. 2020. no. 2. pp. 5–19.
10. Danilin G. V., Sokolov S. S., Nyrkov A. P., Knysh T. P. Mul'tiservisnye seti: metody povysheniya zashchishchennosti dannyh v usloviyah setevyh atak [Multiservice Networks: Methods for Enhancing Data Security in the Face of Network Attacks]. *XXI vek: itogi proshlogo i problemy nastoyashchego plyus*. 2020. T. 9. no. 2 (50). pp. 158–163.
11. Saenko I. B., Kotenko I. V., Al-Barri M. Kh. Primenenie iskusstvennyh nejronnyh setej dlya vyyavleniya ano-mal'nogo povedeniya pol'zovatelej centrov obrabotki dannyh [Application of artificial neural networks to detect abnormal user behavior in data centers]. *Voprosy kiberbezopasnosti*, 2022, no. 2 (48). pp. 87–97.
12. Zelenin D. D. Razrabotka modeli upravleniya riskami dlya korporativnoj seti, ata-kuemoj vredonosnym programmnym obespecheniem tipa setevoj cherv' [Development of a risk management model for a corporate network that is being attacked by malware such as network worms]. *Vestnik Voronezh-skogo instituta vysokih tekhnologij,* 2020, no. 2 (33), pp. 137–142.
13. Semykina N. A. Matematicheskaya model' zashchity v informacionnoj sisteme ierar-hicheskoj struktury [Mathematical model of protection in a hierarchical information system]. *Yuzhno-Sibirskij nauchnyj vestnik*, 2021, no. 4 (38). pp. 75–80.
14. Gryzunov V. V. Adaptivnoe upravlenie dostupnost'yu resursov geoinformacion-noj sistemy kriticheskogo primeneniya v usloviyah destruktivnyh vozdejstvij [Adaptive Management of the Availability of Critical Applications of Geoinformation Systems in the Face of Destructive Influences]. Ph. D. thesis. Saint Petersburg, 2022, 395 p.
15. Lipatnikov V. A. Raspoznavanie vtorzhenij narushitelya pri upravlenii kiber-bezopasnost'yu infrastruktury integrirovannoj organizacii na osnove nejro-nechetkih setej i kognitivnogo modelirovaniya [Intrusion detection in the cyber security management of an integrated organization's infrastructure based on neuro-fuzzy networks and cognitive modeling]. *VIII Mezhdunarodnaja nauchno-tekhnicheskaja i nauchno-metodicheskaja konferencija*, Saint Petersburg, SPbGUT, 2018.
16. Aho A., Lam M., Seti R., Ul'man D. Kompilyatory. Principy, tekhnologii, instrumenty [Principles, technologies, tools]. *Vil'yams*, 2001. 1184 p.
17. Vitkova L. A. Model' vredonosnoj informacii i ee rasprostranitelya v soci-al'nyh setyah [Model of malicious information and its spreader on social media]. *Zashchita informacii. Insajd*, 2020, no. 3(93), pp. 66–72.
18. Krasov A. V. Obespechenie bezopasnosti peredachi multicast-trafika v IP-setyah [Ensuring the security of multicast traffic transmission in IP networks]. *Zashchita informacii. Insajd*, 2017, no. 3(75), pp. 34–42.
19. Saharov D. V. Ispol'zovanie matematicheskih metodov prognozirovaniya dlya ocen-ki nagruzki na vychislitel'nuyu moshchnost' IoT-seti [Using mathematical forecasting methods to assess the computational load on an IoT network]. *Vestnik Sankt-Peterburgskogo universiteta Gosudarstvennoj protivopozharnoj sluzhby MChS Rossii*, 2020, no. 2, pp. 86–94.
20. Abramova T. V. *Obnaruzhenie anomalij i nejtralizaciya ugroz v raspredelennyh avtomatizirovannyh sistemah upravleniya na osnove monitoringa setevyh informacion-nyh potokov* [Detection of

anomalies and neutralization of threats in distributed automated control systems based on monitoring of network information flows]. Ph. D. thesis. Orenburg, 2024. 235 p.
21. Sheluhin O. I. Obnaruzhenie anomal'nyh vybrosov telekommunikacionnogo tra-fika metodami diskretnogo vejvlet-analiza [Detection of abnormal telecommunications traffic emissions using discrete wavelet analysis]. *Elektromagnitnye volny i elektronnye sistemy*, 2012. V. 17, no. 2, pp. 15–26.
22. Shelukhin O. I. *Intellektual'nye tekhnologii informatsionnoi bezopasnosti* [Intelligent Information Security Technologies]. Moscow, Goryachaya liniya – Telekom, 2023. 384 p.
23. Zegzhda D. P. *Kiberbezopasnost' tsifrovoi industrii. Teoriya i praktika funktsional'noi ustoichivosti k kiberatakam* [Cybersecurity of the Digital Industry. Theory and Practice of Functional Resilience to Cyberattacks]. Moscow, Goryachaya liniya – Telekom, 2022. 560 p.
24. Alaloye H. H. Utilizing NLP to Optimize Municipal Services Delivery Using a Novel Municipal Arabic Dataset. *International Journal of Advanced Computer Science and Applications*, 2025, vol. 16, no. 2.
25. Shterenberg S. I. *Programma vzaimodeistviya s bibliotekoi AutoML dlya testirovaniya neironnykh setei na uyazvimosti* [A Program for Interacting with the AutoML Library for Testing Neural Networks for Vulnerabilities]. Certificate of State Registration of the Computer Program No. 2025617273 Russian Federation. Applied March 10, 2025; Published March 25, 2025.
26. Sobolevskij V. A. Ispol'zovanie tekhnologii AutoML dlya avtomaticheskogo analiza vremennykh ryadov [Using AutoML Technologies for Automatic Time Series Analysis]. *Aviakosmicheskoe priborostroenie*, 2024, no. 10, pp. 10–18.
27. Zagagova A. M. Ispol'zovanie push-uvedomlenij v fishing-atakah [Using push notifications in phishing attacks]. Informacionnye tekhnologii v sovremennom mire, Ekaterinburg, Gumanitarnyj universitet, 16 May, 2025, pp. 78-79.

Stanislav I. Shterenberg

Candidate of Technical Sciences, Associate Professor of the Department of Secure Communication Systems, St. Petersburg State University of Telecommunications named after Prof. M.A. Bonch-Bruevich (SPbSUT, 193232, St. Petersburg, Bolshevikov ave., 22/1), tel. +7 931 337 98 61, e-mail: stas.shterenberg.89@mail.ru, ORCID ID: 0000-0002-4216-6370.

Dmitry N. Grechukhin

Student of the Institute of Magistracy, St. Petersburg State University of Telecommunications named after Prof. M.A. Bonch-Bruevich (SPbSUT, 193232, St. Petersburg, Bolshevikov ave., 22/1), tel. +7 (812) 305-12-55, 1255, e-mail: darkokos1@mail.ru.

Andrey S. Krivets

Student of the Institute of Magistracy, St. Petersburg State University of Telecommunications named after Prof. M.A. Bonch-Bruevich (SPbSUT, 193232, St. Petersburg, Bolshevikov ave., 22/1), tel. +7 (812) 305-12-55, 1255, e-mail: krivets_2002@mail.ru.