УДК 004.8

Тришин И.Д.
студент
1 курс, факультет «Информационных технологий»
Московский политехнический институт
Россия, г. Москва
Сидин С.О.
студент
1 курс, факультет «Информационных технологий»
Московский политехнический институт
Россия, г. Москва
Trishin I.D.
student
1st year, faculty of «Information Technology»
Russia, Moscow
Sidin S.O.
student
1st year, faculty of «Information Technology»
Russia, Moscow

АНАЛИЗ И ОБНАРУЖЕНИЕ ФИШИНГА НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ

ANALYSIS AND DETECTION OF PHISHING BASED ON NEURAL NETWORKS

Аннотация. *В данной статье рассматривается фишинг – одна из наиболее распространённых и быстро меняющихся киберугроз, нацеленной на хищение учётных данных, платёжной и персональной информации. Классические методы защиты (чёрные/белые списки, фильтрация, эвристики и обучение пользователей) снижают риск, но недостаточно эффективны при высокой адаптивности атак и краткоживущих*

фишинговых ресурсах. В обзоре систематизированы современные подходы к обнаружению фишинга с использованием машинного и глубокого обучения, с фокусом на нейросетевых архитектурах (CNN, LSTM, BiLSTM, DNN, MLP и гибриды). Рассмотрены типы данных (URL, HTML-контент, e-mail, DNS-трафик, поведенческие логи), типовые этапы подготовки данных и ключевые метрики (accuracy, precision, recall, F1, AUC ROC). Показано, что нейросетевые модели, особенно CNN, LSTM/BiLSTM и гибридные решения, часто обеспечивают лучшую точность и обобщающую способность по сравнению с классическими ML-алгоритмами, что делает их перспективной основой антифишинговых систем.

Ключевые слова: фишинг; фишинговая атака; нейронные сети; машинное обучение; глубокое обучение; CNN; LSTM; BiLSTM; DNN; классификация URL; защита от фишинга.

Abstract. This article examines phishing – one of the most common and rapidly evolving cyber threats aimed at stealing credentials, payment, and personal information. Traditional protection methods (black/white lists, filtering, heuristics, and user training) reduce the risk but are insufficiently effective against highly adaptive attacks and short-lived phishing resources. The review systematizes modern approaches to phishing detection using machine and deep learning, with a focus on neural network architectures (CNN, LSTM, BiLSTM, DNN, MLP, and hybrid models). The types of data (URL, HTML content, e-mail, DNS traffic, behavioral logs), typical data preparation steps, and key metrics (accuracy, precision, recall, F1, AUC ROC) are considered. It is shown that neural network models, especially CNN, LSTM/BiLSTM, and hybrid solutions, often provide better accuracy and generalization capability compared to classical ML algorithms, making them a promising foundation for anti-phishing systems.

Keywords: phishing; phishing attack; neural networks; machine learning; deep learning; CNN; LSTM; BiLSTM; DNN; URL classification; phishing protection.

Введение: фишинг — кибератака, сочетающая социальную инженерию и технические средства, направленная на получение конфиденциальной информации через поддельные письма, сайты, сообщения и иные каналы. Для организаций фишинг часто выступает начальной точкой компрометации инфраструктуры.

Высокая результативность фишинга объясняется ростом интернет-аудитории, активным использованием онлайн-сервисов и тем, что даже пользователи с высоким уровнем образования нередко переоценивают способность распознавать угрозы. Традиционные меры защиты полезны, но не обеспечивают достаточной скорости и полноты детектирования в условиях постоянного изменения шаблонов атак[3]. Поэтому перспективным направлением являются методы ML/DL, прежде всего нейронные сети, способные автоматически извлекать признаки из разнородных данных и адаптироваться к новым паттернам фишинговой активности [1-3].

Для подготовки обзора были проанализированы русскоязычные и англоязычные публикации, посвящённые применению нейронных сетей и других методов ML/DL к задачам обнаружения фишинга в URL-адресах, содержимом веб-страниц, электронных письмах и DNS-трафике [1–3, 5, 8–10].

В качестве основных источников использованы:

1. обзор и сравнительный анализ ML/DL-подходов к защите от фишинга (CNN, LSTM, DNN, классические ML-алгоритмы).
2. работа по разработке системы защиты от фишинговых атак с использованием многослойного персептрона (MLP) и 30 признаков URL.
3. исследование по обнаружению фишинговых электронных писем с использованием рекуррентных нейронных сетей (RNN, LSTM, BiLSTM) на русскоязычном датасете.
4. обзор и практическое исследование методов машинного обучения для идентификации фишинговых ресурсов (логистическая

регрессия, деревья решений, Random Forest, CatBoost, SGD) на основе большого URL-датасета

5. ряд зарубежных работ, посвящённых CNN, eXpose-модели, CNN-LSTM-архитектурам, GRU-сетям, DNN и гибридным моделям для обнаружения фишинговых URL, вредоносных доменов и сайтов.

Отбор статей осуществлялся с учётом наличия экспериментальной части, описания структуры датасетов, архитектуры моделей и используемых метрик качества (accuracy, precision, recall, F1, AUC ROC).

Результаты оригинального авторского исследования:

В рамках данной работы авторами проведено **о**бзорно-аналитическое исследование современного состояния методов обнаружения фишинга с применением нейронных сетей. Основным результатом выступает систематизация подходов, типов данных, этапов подготовки и сопоставление типовых показателей качества, демонстрируемых нейросетевыми архитектурами.

По итогам анализа источников авторы выделили ключевые классы данных, на которых строятся антифишинговые решения: URL-строки, HTML-контент, электронные письма, DNS-трафик, а также поведенческие логи пользователей. Показано, что выбор и качество датасета критически влияют на итоговые метрики моделей и применимость решения в реальных условиях. Отмечено, что глубокие модели, как правило, требуют существенно больших наборов данных, чем классические методы машинного обучения, особенно при работе с «сырыми» строками URL и текстом.

Главный содержательный вывод исследования заключается в том, что нейросетевые модели в ряде задач фишинг-детектирования демонстрируют более высокую точность и лучшую обобщающую способность по сравнению с традиционными ML-алгоритмами, особенно при автоматическом извлечении признаков из текстовых и последовательностных данных.

Также сформулированы практические ограничения внедрения: дефицит регулярно обновляемых датасетов, адаптивность злоумышленников и требования к масштабируемости и задержкам при интеграции в инфраструктуру.

Цель работы: обобщить опыт применения нейросетей для обнаружения фишинга; систематизировать типы данных, подходы к датасетам/предобработке, архитектуры моделей и результаты по ключевым метрикам.

Фишинг как вид кибермошенничества опирается на два ключевых компонента: психологическое воздействие (социальная инженерия) и техническую инфраструктуру (фишинговые сайты, вредоносные вложения, подделка отправителя и др.). С точки зрения анализа средствами ML/DL, важна классификация по типу атакуемых и анализируемых элементов [1]:

1. Атаки на основе URL-адресов:

- Подмена доменного имени (набранного сходно с легитимным брендом)
- Использование IP-адресов вместо доменов
- Применение коротких URL-сервисов
- Внедрение лишних протоколов и двойных слешей
- Избыточная длина, большое число спецсимволов и подкаталогов [1].

2. Атаки через электронную почту:

- Подделка адреса и имени отправителя
- Внедрение ссылок на фишинговые сайты
- Вложенные документы (архивы, офисные файлы) с вредоносным кодом
- Использование актуальных тем (банковские уведомления, государственные сервисы и т.п.) [2].

3. Фишинговые веб-сайты:

- Клон легитимного сайта (визуальное и структурное сходство)
- Наличие форм ввода учётных данных и платёжной информации
- Использование поддельных или дешёвых SSL-сертификатов
- Изменённая структура гиперссылок и скриптов [1].

4. Атаки через DNS-трафик и доменные имена:

- Вредоносные или алгоритмически сгенерированные домены

- Аномальные паттерны в DNS-запросах и ответах
- Использование фишинговых доменов для компрометации IoT и других устройств.

5. Поведенческие и профильные данные:

- Поведение пользователей при переходе по ссылкам
- Типичные сценарии посещения сайтов
- Профили пользователей и потенциальных жертв фишинга.

С точки зрения построения нейросетевых моделей, наиболее распространёнными объектами анализа являются: URL-строки, текст и метаданные электронных писем, HTML-контент веб-страниц, а также DNS-трафик.

Качество работы нейронных сетей критически зависит от качества и репрезентативности используемых датасетов [1–3, 5]. В работах по фишингу применяются как специализированные наборы URL/сайтов, так и смешанные датасеты, включающие HTML-контент, e-mail и сетевой трафик. Часто используемые источники фишинговых и вредоносных URL и доменов:

- PhishTank, OpenPhish, VirusTotal, Alexa, различные репозитории вредоносных доменов [2]
- UCI Machine Learning Repository, Kaggle и другие открытые сборники, содержащие размеченные URL с набором фиксированных признаков [2]
- Собственные сборы: русскоязычные фишинговые письма, собранные с почтовых серверов по IMAP, с последующей ручной разметкой [3].

Для классических ML-моделей часто достаточно датасетов размером от сотен до нескольких тысяч примеров, в то время как глубокие нейронные сети требуют значительно больших объёмов данных (десятки тысяч и более). Примеры:

- Порядка 19 млн URL-адресов для CNN-модели eXpose
- 60–150 тыс. URL-адресов для CNN, CNN-LSTM и GRU-сетей
- 2–3 млн URL-адресов и до 3,2 млн признаков для сравнения классических классификаторов
- Около 4 тыс. URL (2000 легитимных и 2000 фишинговых) для LSTM-модели с признаками URL
- Датасеты электронных писем малой мощности (около 300 писем) для сравнения RNN, LSTM и BiLSTM
- Свыше 800 тыс. URL-адресов в практической работе по оценке

методов ML (Logistic Regression, Random Forest, CatBoost и др.).

При этом важным аспектом является баланс классов (фишинговые/легитимные). Несбалансированность может существенно влиять на метрики (особенно accuracy). Для её устранения применяются методы взвешивания классов, oversampling/undersampling и генерация синтетических примеров (SMOTE) [5].

С точки зрения признакового пространства можно выделить несколько подходов:

1. Ручные (инженерные) признаки URL и доменов [2]:

- Длина URL, домена и поддоменов
- Использование IP-адреса вместо доменного имени
- Число подкаталогов, точек, специальных символов, цифр, дефисов, двойных слешей, протоколов
- Наличие «подозрительных» подстрок (login, verify, secure и пр.)
- Информация о SSL-сертификате и возрасте домена
- Использование URL-сокращателей.

2. Признаки содержимого (HTML, e-mail) [1]:

- Текстовые признаки (частота слов, n-граммы, TF-IDF, word2vec/эмбеддинги)
- Наличие форм ввода, скриптов, внешних ресурсов
- Структурные особенности документа.

2. Векторизация символов и слов [2]:

- Представление URL или текста как последовательности символов и построение эмбеддингов на уровне символов (character-level CNN/RNN)
- Использование предобученных языковых моделей (например, SpaCy ru_core_news_md) для лемматизации и получения векторных представлений слов [3]
- TF-IDF-векторизация лемм URL для последующей подачи на вход классическим или нейросетевым моделям [5].

3. Признаки DNS-трафика и поведения:

- Статистика длины доменных имён, частота символов
- Эмбеддинги на уровне символов/слов для доменных строк
- Временные паттерны DNS-запросов, графовые признаки.

Предобработка включает очистку данных (удаление дубликатов, исправление кодировок), обработку пропусков и выбросов, нормализацию числовых признаков, кодирование категориальных атрибутов, а также

выделение и лемматизацию токенов для текстовых данных [1–3, 5].

Recurrent Neural Networks (RNN), LSTM, BiLSTM Рекуррентные нейронные сети применяются в задачах, где важно учитывать последовательный характер данных: текст писем, URL-строки, последовательности DNS-запросов [1].

– LSTM (Long Short-Term Memory) эффективно обрабатывают длинные последовательности, запоминая дальний контекст и подавляя проблему исчезающего градиента. LSTM показывали точность свыше 99 % при обнаружении фишинговых URL и порядка 98 % при классификации писем в отдельных исследованиях.

– BiLSTM (двунаправленные LSTM) анализируют последовательность в двух направлениях (вперёд и назад), что позволяет лучше учитывать контекст. В исследовании русскоязычных писем BiLSTM-сеть продемонстрировала наилучший результат, корректно обнаружив 91,43 % фишинговых сообщений при малом объёме обучающего датасета (300 писем), в то время как обычная RNN показала лишь около 50,71 % [3].

Таким образом, рекуррентные сети, особенно BiLSTM, рекомендуются для задач анализа текста писем и последовательностных представлений URL/доменных имён, особенно при относительно небольших объёмах качественно размеченных данных [1].

Convolutional Neural Networks (CNN)

Сверточные нейронные сети широко используются для анализа URL-строк и текстов на уровне символов и слов.

Их ключевые способности:

– применение свёрток к последовательности символов или токенов для извлечения локальных паттернов (подстрок, характерных комбинаций символов);

– возможность автоматического выделения информативных признаков без ручной инженерии;

– высокая производительность при работе с большими датасетами.

Гибридные модели (CNN-LSTM, CNN-GRU, DNN+эмбеддинги) Комбинации различных архитектур направлены на повышение точности и устойчивости моделей:

– CNN-LSTM: свёрточная часть выделяет локальные паттерны в URL или тексте, LSTM-часть моделирует последовательные зависимости. Такие модели показывали accuracy до 98,9 % и AUC свыше 0,99 для фишинговых [3].

– GRU-сети (Gated Recurrent Units) в сочетании с Random Forest: GRU-модель, обученная на признаках URL, демонстрировала лучшую точность (до 98,5 %) по сравнению с «лесом» на тех же данных.

– DNN (Deep Neural Networks) с семантическими признаками: интеграция word2vec-эмбеддингов и статистических признаков позволяла улучшить качество обнаружения фишинга по HTML и URL.

Многослойный персептрон (MLP)

Многослойный персептрон рассматривается как базовая архитектура для работы с табличными признаками. В работе разработана система защиты от фишинговых атак на основе MLP, анализирующего URL по 30 ключевым признакам (длина URL, наличие SSL-сертификата, использование IP-адресов и др.) [2].

– Архитектура включала входной слой (30 признаков), два скрытых слоя (32 и 10 нейронов) и выходной нейрон с сигмоидной активацией [2]

– Реализация на Python с использованием TensorFlow и Scikit-Learn показала точность 98,2 %, что значительно превысило результаты традиционных методов фильтрации (не более 85 %) [2]

Таким образом, MLP хорошо подходит для задач, где уже сформирован фиксированный набор информативных признаков (feature-based подход) и важно получить высокую точность при умеренных вычислительных затратах.

Нейросети для анализа DNS-трафика и профилей пользователей

Отдельный класс задач связан с обнаружением фишинга и вредоносной активности на уровне DNS-трафика:

– CNN и LSTM используются для анализа строк доменных имён (эмбеддинги символов, свёртки, последовательные модели);

– гибридные системы (DNS-фильтрация + CNN-LSTM) демонстрируют AUC порядка 0,95 при работе в реальном времени;

– обработка потоков до 10 млрд DNS-запросов в день требует высокопроизводительных и хорошо масштабируемых архитектур, где DL-модели показывают заметные преимущества .

Также изучались модели DNN для анализа поведенческих логов пользователей и формирования профилей потенциальных жертв фишинга, что позволяет выявлять аномальное поведение и повышать точность детектирования сложных атак.

Во многих работах проводилось сравнение нейросетевых моделей с классическими алгоритмами машинного обучения (SVM, Naive Bayes, Decision Tree, Random Forest, Logistic Regression, kNN, Gradient Boosting и др.).

Основные наблюдения:

– Для фишинговых URL методы глубокого обучения (CNN, LSTM, CNN-LSTM) часто превосходят классические алгоритмы по accuracy, F-мере и AUC, особенно когда модель работает напрямую с «сырыми» URL-строками без ручного конструирования признаков.

– В задачах с табличными признаками (фиксированный набор характеристик URL) хорошо себя проявляют Random Forest, Gradient Boosting и MLP, достигая точности свыше 97–99 % при корректном выборе признаков и балансировке классов [2].

– В практическом исследовании на ~800 тыс. URL-адресов логистическая регрессия, CatBoost, SGD-классификатор и дерево решений показали accuracy на уровне 0,85–0,93, что сопоставимо с некоторыми DL-подходами

при использовании только словарных признаков [5]. – Нейросетевые модели получают максимальные преимущества при наличии больших объёмов данных и возможности работать с текстом/URL в «сыром» виде, в то время как на малых датасетах грамотная инженерия признаков и ансамблевые ML-методы могут быть не менее эффективны.

Обсуждение и нерешённые проблемы:

Несмотря на высокие показатели точности, применение нейронных сетей для анализа фишинга сталкивается с рядом проблем:

1. **Датасеты:** недостаток открытых, репрезентативных и регулярно обновляемых наборов (особенно для русскоязычных писем) [3]; необходимость постоянного обновления из-за эволюции атак[1].
2. **Интерпретируемость:** нейросети часто выглядят как “чёрные ящики”, что затрудняет внедрение в критические системы и объяснение срабатываний.
3. **Адаптивный противник:** злоумышленники изменяют URL/контент, чтобы обходить детекторы; требуются методы повышения устойчивости к обфускации и атакующим примерам.
4. **Интеграция в инфраструктуру:** важны задержки, масштабируемость и встраивание в почтовые шлюзы, прокси, SIEM/SOC; это остаётся значимым прикладным направлением [2].

Тем не менее совокупность результатов показывает, что нейронные сети являются одним из наиболее эффективных инструментов для обнаружения фишинговых атак, особенно в сочетании с хорошо спроектированной системой признаков, регулярным обновлением датасетов и комплексной архитектурой защиты (объединяющей URL-анализ, фильтрацию почты, анализ контента и DNS-трафика).

Заключение

Фишинг — ключевая угроза, и традиционные методы защиты не успевают за скоростью изменения атак. Нейросетевые модели (CNN,

LSTM/BiLSTM, GRU, DNN и гибриды) демонстрируют высокую точность при анализе URL, сайтов, писем и доменов и часто превосходят классические ML-алгоритмы, особенно при работе с “сырыми” строками и/или большими объёмами данных. Для сценариев с фиксированными признаками URL практичны MLP и другие модели по табличным данным, обеспечивающие высокое качество при умеренной стоимости вычислений.

Перспективы развития включают: расширение и регулярное обновление датасетов (в т.ч. многоязычных), развитие методов отбора интерпретируемых признаков, повышение устойчивости к адаптивным атакам и внедрение моделей в комплексные системы мониторинга и реагирования в реальном времени.

Список литературы

1. Корнюхина С.П., Лапонина О.Р. Исследование возможностей алгоритмов глубокого обучения для защиты от фишинговых атак // International Journal of Open Information Technologies. 2023. Т. 11. № 6. С. 163–174.

2. Лукманова К.А., Картак В.М. Разработка системы защиты от фишинговых атак с использованием программно-аппаратной реализации методов машинного обучения // Моделирование, оптимизация и информационные технологии. 2024;12(4). DOI: 10.26102/2310-6018/2024.47.4.033.

3. Болдырихин Н.В., Ядрец Э.А. Обнаружение фишинговых электронных писем с помощью рекуррентных нейронных сетей // Вопросы кибербезопасности. 2025. № 4. С. 134–141.

4. Catal C. et al. Applications of deep learning for phishing detection: a systematic literature review // Knowledge and Information Systems. 2022. Vol. 64. No. 6. P. 1457–1500.

5. Гальцев Б.С. Применение методов машинного обучения для идентификации фишинговых ресурсов // Научный журнал КубГАУ. 2024. №

200(06). С. 1–18

6. Yerima S.Y., Alzaylaee M.K. High accuracy phishing detection based on convolutional neural networks // 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS). IEEE, 2020. P. 1–6.

7. Wang W. et al. PDRCNN: Precise phishing detection with recurrent convolutional neural networks // Security and Communication Networks. 2019. Vol. 2019. P. 1–15.

8. Saxe J., Berlin K. eXpose: A Character-Level Convolutional Neural Network with Embeddings For Detecting Malicious URLs, File Paths and Registry Keys. 2017.

9. Vazhayil A., Vinayakumar R., Soman K. Comparative Study of the Detection of Malicious URLs Using Shallow and Deep Networks // Proc. 2018 9th Int. Conf. on Computing, Communication and Networking Technologies (ICCCNT). 2018. P. 1–6.

10. Bahnsen A.C. et al. Classifying phishing URLs using recurrent neural networks // Proc. 2017 APWG Symposium on Electronic Crime Research (eCrime). 2017. P. 1–8.