Система защиты от фишинг-атак на основе методов машинного обучения

Минко А.В.
Аспирант
Кафедра вычислительной техники и защиты информации
Уфимский университет науки и технологий
Уфа, Россия
e-mail: alex_shtem@mail.ru

Вульфин А.М.
Профессор, д.т.н.
Кафедра вычислительной техники и защиты информации
Уфимский университет науки и технологий
Уфа, Россия
e-mail: vulfin.am@ugatu.su

Кириллова А.Д.
Доцент, к.т.н.
Кафедра вычислительной техники и защиты информации
Уфимский университет науки и технологий
Уфа, Россия
e-mail: kirillova.ad@ugatu.su

Аннотация[1]

Актуальность темы обусловлена непрерывно возрастающим количеством атак на информационные системы с использованием методов социальной инженерии, включая атаки с подменой ссылок на официальные сайты организаций (фишинг) для получения конфиденциальной информации. Применение организационных мер и обучение пользователей должно сопровождаться применением встроенных и наложенных средств защиты информации от направленных и веерных фишинг-атак. Эффективная программно-аппаратная реализация методов машинного обучения в задачах обнаружения фишинговых ссылок позволит повысить оперативность анализа и снизить затраты на вычислительные мощности.

Ключевые слова: фишинг; машинное обучение; центр мониторинга информационной безопасности; система обнаружения фишиновых ссылок.

1. Введение

По данным экспертов сервиса управления навыками кибербезопасности Security Awareness ГК «Солар» в 2024 году 34% входящих электронных писем в российских организациях содержат спам, фишинговые ссылки и вредоносное программное обеспечение. Несмотря на разнообразие методов, используемых злоумышленниками для достижения своих целей, фишинг является один из наиболее распространенных и успешных методов получения личных данных пользователей.

За последние несколько лет были разработаны различные решения для обнаружения фишинга, однако данная проблема все еще актуальна, так как разработка эффективного, не требующего большой вычислительной мощности подхода является сложной задачей. Более того, большинство таких решений имеют высокий уровень ложных срабатываний.

Хотя использование алгоритмов машинного обучения (Machine Learning, ML) для решения задачи классификации и, в частности, обеспечения информационной безопасности (ИБ) и обнаружения вредоносного программного обеспечения, вызывает большой интересу исследователей, все еще не существует универсального решения для эффективной защиты от фишинг-атак.

Следовательно, актуальным является создание эффективной системы на основе методов машинного обучения в задачах обнаружения фишинговых ссылок для повышения оперативности анализа и снижения затрат на вычислительные мощности.

Цель работы заключается в повышении эффективности обнаружения фишинговых ссылок за счет разработки и программно-аппаратной реализации классификатора на основе методов машинного обучения.

Труды X Международной научной конференции "Информационные технологии интеллектуальной поддержки принятия решений", 12-14 ноября, Уфа-Баку-Чандигарх, 2024

2. Анализ проблем выявления фишинговых ссылок на основе методов машинного обучения

При выполнении анализа существующих работ, было обнаружено, что алгоритмы машинного обучения являются одними из самых популярных способов обнаружения фишинговых веб-сайтов (Таблица 1).

Было выявлено, что все существующие традиционные решения способны обнаружить только 20% фишинговых атак, совершаемых на сервер системы или онлайн-пользователя [1].

Чтобы построить модель машинного обучения для системы обнаружения фишинговых атак, имеющиеся данные должны иметь признаки, связанные с классами фишинга и легитимных веб-сайтов.

Таблица 1. – Обзор исследований с использованием методов машинного обучения

Авторы и год публикации	Используемые алгоритмы обучения	Используемый набор данных	Основные комментарии и выявленные пробелы
Saha I. и др. 2020 [2]	Многослойный персептрон	Kaggle	Отсутствует параметр соотношения набора данных, нет коэффициента разделения набора данных на обучающую и тестовую выборку, используется небольшое количество признаков вебсайта, нет анализа с другими алгоритмами машинного обучения
Gupta B.B. и др., 2021 [3]	Метод k-ближайших соседей, случайный лес, логическая регрессия, метод опорных векторов	ISCXU RL-2016	Несбалансированный набор данных, не проводится сравнительный анализ между алгоритмами машинного обучения, нет динамического анализа
Odeh A. и др., 2020 [4]	Многослойный персептрон	PhishTank, поиск Google, MillerSmiles	Отсутствуют данные о соотношении набора данных, нет динамического анализа, нет сравнительного анализа с другими методами машинного обучения
Subasi A., Kremic E. , 2019 [5]	Метод k-ближайших соседей, случайный лес, дерево решений, метод опорных векторов, искусственная нейронная сеть, Adaboost Multiboost	UCI Machine Learning repository	Размер набор данных не указан, нет коэффициента разделения набора данных на обучающую и тестовую выборку, отсутствуют методы выбора признаков, высокое вычислительное время системы
Hannousse A., Yahiouche S., 2021 [6]	Дерево решений, случайный лес, логическая регрессия, наивный байесовский классификатор, метод опорных векторов	Phishtank, Openphish, Alexa, Yardex	Нет коэффициента разделения набора данных на обучающую и тестовую выборку, непригодность некоторых используемых функций для динамического анализа
Kumar P. P., Jaya T., Rajendran V., 2021 [7]	Сверточная нейронная сеть	Kaggle	Набор данных несбалансированный, нет анализа с другими алгоритмами машинного обучения, нет коэффициента разделения набора данных на обучающую и тестовую выборку, нет динамического анализа

Для сбора соответствующих признаков требуется тщательное исследование. Так как фишинг является постоянной угрозой, которая постоянно эволюционирует, то решение проблемы заключается в регулярном изучении новых признаков атак и включении их в обучающий набор данных. Однако это увеличивает размер набора данных. Существует потребность в алгоритме классификации, который использует не самое большое количество признаков, чтобы предвидеть новые попытки фишинга.

В рассмотренных работах результаты многих алгоритмов обучения показывают, что случайный лес имеет наилучшую общую производительность с точностью результатов от 94,6% до 99,57%. Другие модели и алгоритмы, такие как: метод опорных векторов, многослойный персептрон, логистическая регрессия, машина экстремального обучения (ELM), повышения градиента, искусственная нейронная сеть, сверточная нейронная сеть показали наивысшую общую точность от 97,6% до 99,9%, но уже с большими затратами вычислительных ресурсов и машинного времени.

Поскольку важной частью защиты от фишинга является точное и своевременное обнаружение фишинговых URL-ссылок, предлагается выбрать в качестве приоритета случайный лес или один из вышеупомянутых алгоритмов машинного обучения, для сравнения – многослойный персептрон, наряду с использованием сбалансированного набора данных и соответствующих методов выбора функций.

3. Разработка структурной модели системы обнаружения фишинговых ссылок в составе центра мониторинга ИБ

Предлагаемая структурная схема системы обнаружения фишинговых ссылок представлена на рисунке 1.

Система включает в себя подсистемы: А – сбора данных и доменных имен; Б – выделения признаков; В – создание и управление моделями машинного обучения; Г – управление данными из внешних источников.

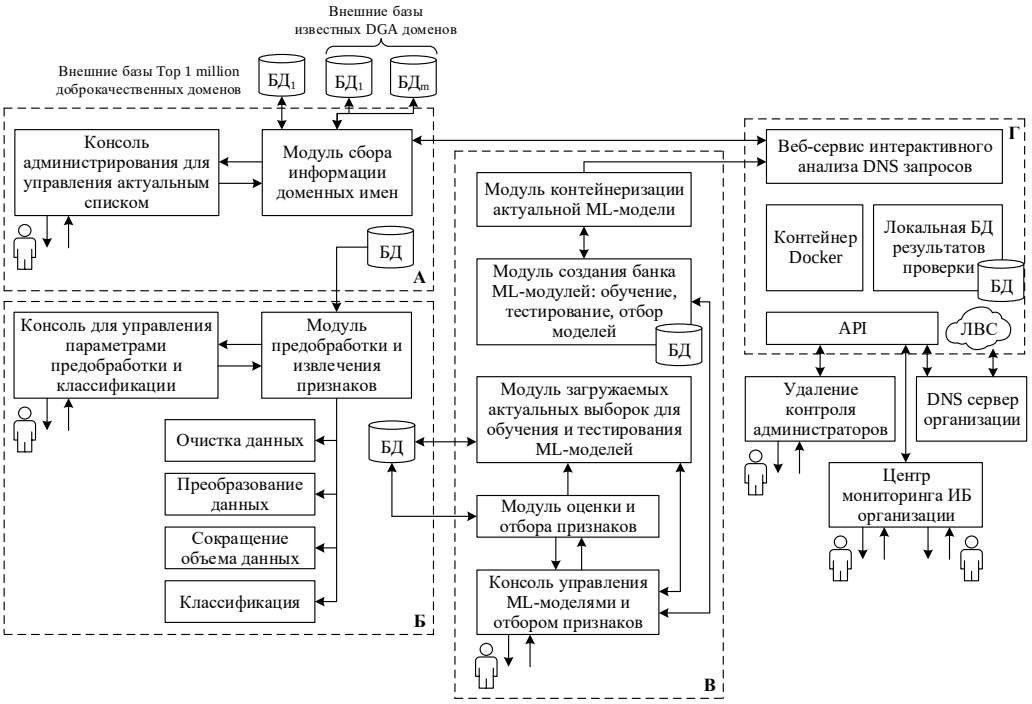

Рис. 1. Структурная схема системы обнаружения фишинговых ссылок

В подсистеме сбора данных о доменных именах содержится модуль сбора информации доменных имен, который необходим для дальнейших обучений разработанной системы.

В подсистеме выделения признаков находится модуль предобработки и извлечения признаков, который выполняет предобработку данных, преобразование и сокращение объема данных.

4. Разработка методического обеспечения применения системы в составе центра мониторинга ИБ

Большинство современный различных межотраслевых и крупных предприятий, а также государственных организациях стараются внедрить эффективные центры мониторинга ИБ. Такие центры в основном состоят из систем сбора и анализа событий, которые появляются как во внешних, так и во внутренних структурах [8].

Разработанная система обнаружения фишинговых ссылок направлена не только на быстродействие и точность определения, а также на возможность легко встроить систему в существующие центры мониторинга ИБ. На рисунке 2 показана схема интеграции предложенного решения в качестве подсистемы центра мониторинга ИБ.

Разработанная система анализирует ссылки и при обнаружении подозрительных элементов обозначает их как события ИБ и направляет в SIEM-систему.

5. Оценка эффективности предложенного решения на натурных данных

Для проверки эффективности разработанной системы проведен эксперимент на различных аппаратных платформах.

В созданном наборе данных общее количество легитимных ссылок составило 1094, а количество примеров, относящихся к фишинговым URL – 1362. Все признаки, извлекаемые из символьной ссылки, нормализованы и имеют бинарные значения для определения: от -1 до 1, где -1 означает фишинговая ссылка, 0 – подозрительная, и 1 – легитимная.

Набор данных был разделен на тестовую и обучающую выборку. Для этого использовали соотношение 20% и 80%.

При помощи перекрестной проверки с разбиением в 10 групп происходит оценка обобщающей способности классификатора. В качестве классификатора использованы: дерево решений, случайный лес и многослойный персептрон. Для построения дерева решений был применен алгоритм максимизации критерия неопределенности Джини.

Рис. 2. Схема интеграции предложенного решения в состав центра мониторинга ИБ

Для оценки производительности классификатора использованы такие параметры, как Accuracy, Recall, Precision, мера F1. Также для оценки качества модели в задаче классификации использована матрица ошибок.

Далее выполнен отбор наиболее значимых признаков набора данных с использованием классификаторов на основе дерева решений и случайного леса, результаты оценки значимости признаков представлены на рисунках 3 и 4 соответственно. Признаки отбирались с помощью алгоритма оценки снижения точности при удалении признака (out-of-bag ошибка).

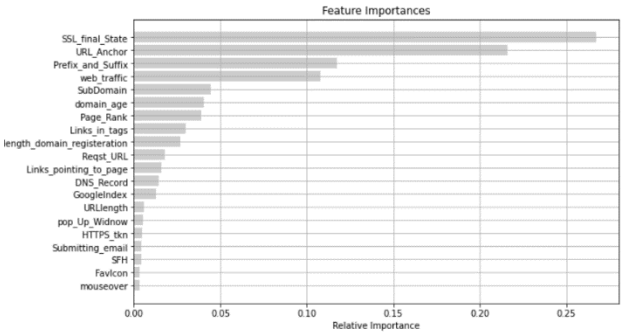

Рис. 3 Оценка значимости признаков с помощью классификатора на основе дерева решений

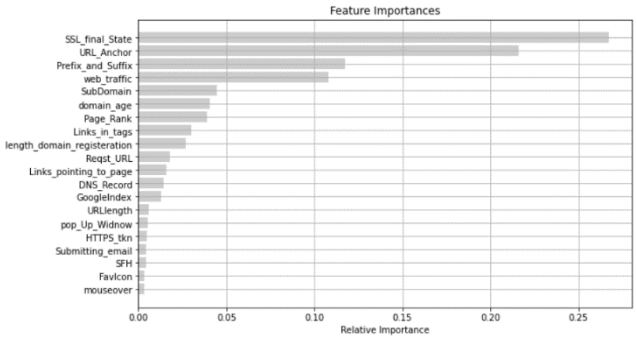

Рис. 4 Оценка значимости признаков с помощью классификатора на основе случайного леса

Для построение итогового классификатора на основе модели случайного леса выполнен подбор гиперпараметров (таблица 2), где сравнительным критерием является Accuracy и мера F1. Был проведен анализ 2000 сочетаний гиперпараметров классификатора.

Таблица 2. – Подбор гиперпараметров классификатора на основе случайного леса

Параметр	Варианты	Выбранное значение
Количество деревьев в лесу	10, 15, 25, 50	50
Ограничение на число объектов в листьях деревьев	1, 3, 5, 7, 10	1

Число признаков для расщепления	3, 5, 10, 15	3

Ниже приведена матрица ошибок для классификатора случайного леса на тестовой выборке после подбора гиперпараметров (таблица 3).

Таблица 3. – Матрица ошибок для классификатора случайного леса на тестовой выборке после подбора гиперпараметров

		Прогнозируемый результат	
		Легитимное	Фишинг
Реальный результат	Легитимное	275	6
	Фишинг	1	210

Полученные метрики качества классификации тестовой выборки показаны в таблице 4.

Таблица 4. – Метрики качества классификации тестовой выборки

Метрика	Значение
Accuracy	0,986
F1-мера	0,986

Результаты оценки эффективности предложенных классификаторов для защиты от фишинговых атак представлены в таблице 5.

Параметры классификатора многослойного персептрона: используется оптимизация веса из семейства квазиньютоновских методов lbfgs, количество эпох равно 2000, 2 скрытых слоя с 32 и 10 нейронами соответственно.

Параметры классификатора случайного леса: подобранные выше оптимальные параметры из таблицы 2, число деревьев 50.

Таблица 5. – Оценка эффективности классификации фишинговых ссылок

Модель классификатора	Accuracy, доля верных классификаций	F1-мера
Многослойный персептрон	0,98	0,98
Случайный лес	0,99	0,99

При помощи транслятора Emlearn [9] с языка Python обученные модели преобразованы в код на языке C: заголовочные файлы и файлы реализации с сохранением полученных во время обучения коэффициентов моделей.

Первый эксперимент проведен на машине с операционной системой Windows 10 версии 21H2, 8 гигабайтами оперативной памяти, процессором Intel Core i5-8300H с 4 ядрами и 8 логическими процессорами.

Полученные значения работы классификаторов для сравнения приведены ниже в таблице 6.

При применении компилятора компании Intel, работа классификаторов показывает наименьшее время. Это обусловлено тем, что он выполняет высокоуровневые и целевые оптимизации специально под процессоры Intel, один из которых и используется на данном компьютере.

Таблица 6. – Время работы классификаторов, собранных различными компиляторами

		Классификатор	
		Случайный лес	Многослойный персептрон
Время выполнения работы классификатора, мс	MinGW	23379	57940555
	Intel compiler	22355	16235246

Следующим шагом эксперимента является проверка работы классификаторов на предполагаемом сетевом устройстве. Использован эмулятор на основе программы с открытым исходным кодом QEMU. Используя динамическую трансляцию, в QEMU достигается очень хорошая производительность.

Конфигурация эмулированного устройства представлена ниже в таблице 7.

Таблица 7. – Конфигурация эмуляции устройства ARM на QEMU

Характеристика	Описание
Процессор	Arm Cortex-a53
Количество виртуальных процессоров	8
Занимаемое место на HDD	55 гигабайт
Оперативная память	8 гигабайт
Операционная система	Linux Ubuntu

Результаты проведенного исследования приведены в таблице 8.

Таблица 8. – Время выполнения работы классификаторов на эмулированном устройстве

Классификатор	Время выполнения работы классификатора, мс
Случайный лес	1082
Многослойный персептрон	11051668

Результаты показали, что созданные классификаторы работают в разы быстрее при эмуляции процессора Arm Cortex-53, чем собранные исходные файлы на платформе персонального компьютера. К тому же стоит заметить, что классификатор случайного леса показал наилучший результат по сравнению с классификатором многослойного персептрона на всех этапах данного эксперимента.

6. Заключение

Разработана структурно-функциональная организация и прототип системы защиты от фишинг-атак на основе

программно-аппаратной реализации методов машинного обучения.

Предложенные алгоритмы и модель анализа фишинговых ссылок реализованы в виде программного обеспечение на языке Python с оценкой эффективности на натурных данных. В итоге у полученных классификаторов вероятность правильно предсказать класс объекта составила 98% у многослойного персептрона и 99% у случайного леса, F1-мера составила у тех же алгоритмов 98% и 99% соответственно (Табл. 1).

Протестирована эффективность реализации с помощью эмулятора процессора ARM Cortex-53. Алгоритм случайного леса при компиляции с установленными флагами максимальной оптимизации превосходит классификатор на основе многослойного персептрона по времени обработки тестовой выборки в 4,5 раза.

Научная новизна предлагаемого решения заключается в разработке комплекса моделей анализа символьного доменного имени, на основе методов нейросетевого моделирования и построения ансамбля случайных деревьев, отличающихся оптимизацией для аппаратных платформ, что позволяет повысить оперативность анализа при встраивании в существующие системы мониторинга.

Список используемых источников

1. Sapkal V., More D. N., Agme M. R. A briefed review on phishing attacks and detection approaches // Related eJournals. 2022. P. 6
2. Saha I. et al. Phishing attacks detection using deep learning approach // 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE, 2020. P. 1180-1185.
3. Gupta B. B. et al. A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment // Computer Communications. 2021. Vol. 175. P. 47–57.
4. Odeh A., Keshta I., Abdelfattah E. Efficient prediction of phishing websites using multilayer perceptron (mlp) // Theor Appl Inf Technol. 2020. P. 3353-3363.
5. Subasi A., Kremic E. Comparison of adaboost with multiboosting for phishing website detection // Procedia Computer Science. 2020. Vol. 168. P. 272–278.
6. Hannousse A., Yahiouche S. Towards benchmark datasets for machine learning based website phishing detection: An experimental study // Engineering Applications of Artificial Intelligence. 2021. Vol. 104. P. 104347.
7. Kumar P. P., Jaya T., Rajendran V. SI-BBA–A novel phishing website detection based on Swarm intelligence with deep learning //Materials Today: Proceedings. 2023. Vol. 80. P. 3129-3139.
8. Минаев В.А., Бондарь К.М., Дунин В.С. Возможности имитационного моделирования деятельности центра оперативного управления и мониторинга информационной безопасности в условиях кибератак различного масштаба // Вестник Воронежского института МВД России. 2021. № 3. – С. 49–59.
9. Emlearn. URL: https://github.com/emlearn/emlearn (дата обращения: 29.09.2024).