

ОБНАРУЖЕНИЕ XSS-АТАК С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

В.А. Землянов¹

¹Самарский университет, г. Самара
e-mail: mister.voker163@yandex.ru

Аннотация

В работе рассматривается проблема обнаружения межсайтового скриптинга (XSS) – одного из наиболее распространенных видов атак на веб-приложения. Предлагается подход к детектированию XSS-атак с использованием алгоритма градиентного бустинга на решающих деревьях, реализованного в библиотеке CatBoost. Приведена демонстрация работы процесса анализа. Экспериментальный анализ подтвердил эффективность алгоритма детектирования с метрикой F1 более 0,89 на наборе данных.

Ключевые слова: XSS, межсайтовый скриптинг, машинное обучение, классификация, безопасность веб-приложений.

Введение

С ростом числа веб-приложений проблема обеспечения их информационной безопасности становится все более актуальной. Атаки типа межсайтовый скриптинг (XSS) стабильно входят в топ-10 самых опасных уязвимостей веб-приложений по версии OWASP. Суть XSS-атаки заключается во внедрении вредоносного JavaScript-кода в веб-страницу, выполняемого затем в браузере жертвы, что может привести к краже cookies, сессий и личных данных пользователей, а также к непредвиденному поведению работы самого веб-приложения [1]. Традиционные сигнатурные методы обнаружения (WAF) не всегда эффективны против новых, ранее не встречавшихся атак [2]. Методы, основанные на предварительном выполнении кода в изолированной среде, являются ресурсозатратными. Целью данной работы является исследование возможности применения методов машинного обучения для обнаружения XSS-атак на основе текста, который ввел пользователь на странице, перед его сохранением в базу данных и демонстрацией другим.

1. Выбор алгоритма для обучения модели.

В качестве основного классификатора был выбран алгоритм CatBoost [3] – реализация градиентного бустинга на решающих деревьях. Преимуществами данного алгоритма применительно к задаче обнаружения XSS являются:

1. Работа с категориальными признаками: CatBoost использует технику симметричных деревьев и эффективно кодирует категориальные признаки (например, тип HTTP-метода, наличие конкретного тега), что исключает необходимость ручного кодирования (One-Hot Encoding) и снижает риск переобучения.

2. Устойчивость к выбросам: в отличие от линейных моделей, бустинг устойчив к редким, но важным признакам, характерным для замаскированных XSS-пайлоадов.

3. Интерпретируемость: несмотря на сложность ансамбля, CatBoost предоставляет встроенные инструменты для оценки важности признаков, что позволяет понять, какие именно характеристики запроса (например, наличие обработчика onerror или длина) вносят наибольший вклад в детектирование атаки.

2. Подготовка и анализ признаков.

Для построения модели необходимо преобразовать входной текст в набор числовых признаков и категориальных признаков. Ключевыми для обнаружения XSS являются:

1) url_keywords_evil: данный признак указывает на содержание в тексте слов alert, prompt, confirm, eval, script, iframe.

2) js_dom_alert: данный признак указывает на содержание в тексте слова alert.

3) html_attr_class: данный признак указывает на содержание в тексте html-атрибута class.

Также алгоритм CatBoost предоставляет возможность передать исходный текст как признак, который также входит в список ключевых.

Вычисление каждого признака в тексте производится с помощью регулярных выражений без учета регистра. Всего их на данный момент используется более 80 [4]. Оценка качества модели проводилась по общепринятым метрикам. Результаты приведены в таблице 1 и на рисунке 1.

Таблица 1 – Метрики качества модели

Метрика	Значение (%)
Accuracy	88.80
Precision	83.01
Recall	97.67
F1-score	89.75
FP-rate	19.99

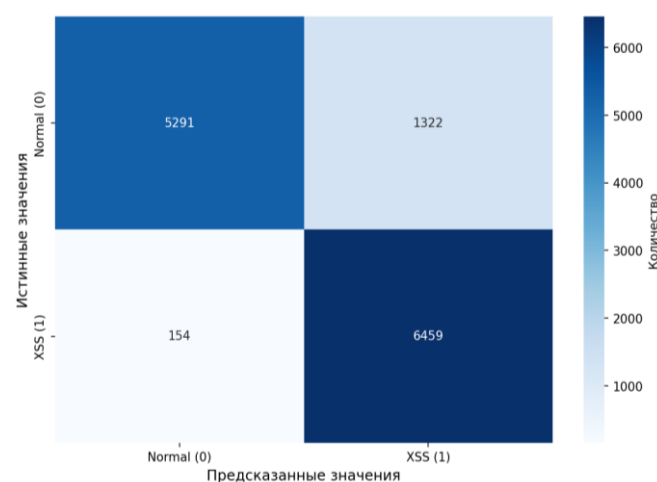


Рисунок 1 – Матрица ошибок

3. Стенд для демонстрации работы процесса обнаружения XSS.

Данная модель создается в качестве дополнения к текущим методам обеспечения безопасности веб-приложений, выступая первым этапом проверки текста. Логика пайплайна:

- 1) Клиентская часть отправляет запрос серверной части.
- 2) Получив сообщение, основной сервер отправляет текст сообщения микросервису с моделью по http.
- 3) Если модель утверждает, что текст является XSS, то сервер отправляет клиенту ошибку с http-кодом 400. В ином случае сохраняет текст в базу данных и отправляет клиенту ответ с http-кодом 204.

Заключение

В ходе работы был продемонстрирован метод обнаружения XSS-атак с помощью методов машинного обучения и тестовый стенд для демонстрации процесса обнаружения в реальных проектах. Использование предложенного подхода позволило решить задачу детектирования XSS-атак с точностью более 0,89 по критерию F1.

Список использованных источников

1. TrustyMon: Practical Detection of DOM-based Cross-Site Scripting Attacks Using Trusted Types / S. Park, J. Kim, S. Keum [et al.] // Proceedings of the ACM Asia Conference on Computer and Communications Security (ASIA CCS '25). – Hanoi, Vietnam : ACM. – 2025. – P. 1–15.
2. Alhamyani, R. Machine Learning-Driven Detection of Cross-Site Scripting Attacks / R. Alhamyani, M. Alshammari // Information. – 2024. – V. 15, № 7. – P. 420.
3. CatBoost: unbiased boosting with categorical features / L. Prokhorenkova, G. Gusev, A. Vorobev [et al.] // Advances in Neural Information Processing Systems. – 2018. – V. 31. – P. 6638–6648.
4. XSShield: A novel dataset and lightweight hybrid deep learning model for XSS attack detection / G.-H. Luu, M.-K. Duong, T.-P. Pham-Ngo [et al.] // Results in Engineering. – 2024. – V. 23. – P. 103363.

DETECTION OF XSS ATTACKS USING MACHINE LEARNING METHODS

V.A. Zemlyanov¹

²*Samara University, Samara
e-mail: mister.voker163@yandex.ru*

Abstract

The paper considers the problem of detecting cross—site scripting (XSS), one of the most common types of attacks on web applications. An approach to detecting XSS attacks using the gradient boosting algorithm on decision trees implemented in the CatBoost library is proposed. A demonstration of the operation of the analysis process is provided. Experimental analysis confirmed the effectiveness of the detection algorithm with an F1 metric of more than 0.89 on the studied data set.

Keywords: XSS, cross-site scripting, machine learning, classification, web application security.