10. Лощилина И. BSC (Сбалансированная система показателей) и Business Studio / И. Лощилина // Business Studio. – Режим доступа: https://www.businessstudio.ru/articles/article/bsc_sbalansirovannaya_sistema_pokazateley_i_busine/, свободный. – Заглавие с экрана. – Яз. рус.
11. Скульский Д. В. Управление бизнес-процессами в муниципальных образованиях на основе искусственного интеллекта / Д. В. Скульский, В. Ф. Шуршев, М. И. Шикульский, Т. И. Гайрабекова // Вестн. Астрахан. гос. техн. ун-та. Сер. управление, вычисл. техн. информ. – 2022. – № 3. – С. 71–79.
12. Добролюбова Е. И. Методическое пособие по разработке (коррекции) и организации реализации государственных программ : учебное пособие / Е .И. Добролюбова, В. Н. Южаков. – Москва : Наука, 2017. – 114 с.
13. Норенков И. Автоматизированные информационные системы / И. Норенков. – Москва : Наука, 2017. – 345 с.
14. Зарипова В. М. Унаследованные информационные системы. проблемы и решения / В. М. Зарипова, И. Ю. Петрова // Инженерно-строительный вестник Прикаспия. – 2022. – № 2 (40). – С. 130–136.
15. Соболева В. В. Методика автоматизированного подбора образовательных технологий для оптимизации учебного процесса в вузе / В. В. Соболева, М. И. Шикульский // Инженерно-строительный вестник Прикаспия. – 2021. – № 1 (35). – С. 81–85.

ссылка для цитирования:
Скульский Д. В., Шуршев В. Ф., Шикульский М. И. Управленческие процессы и развитие искусственного интеллекта в муниципальных образованиях // Инженерно-строительный вестник Прикаспия : научно-технический журнал / Астраханский государственный архитектурно-строительный университет. Астрахань : ГАОУ АО ВО «АГАСУ», 2023. № 2 (44). С. 116–122.

УДК 004.492.3
DOI 10.52684/2312-3702-2023-44-2-122-127

ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ ДЛЯ ДЕТЕКТИРОВАНИЯ ВРЕДОНОСНЫХ URL

О. И. Евдошенко, Ю. А. Лежнина

Евдошенко Олег Игоревич, кандидат технических наук, доцент кафедры высшей математики и программирования, МИРЭА – Российский технологический университет, г. Москва, Российская Федерация;
Лежнина Юлия Аркадьевна, кандидат технических наук, доцент, заместитель директора по научной деятельности, МИРЭА – Российский технологический университет, г. Москва, Российская Федерация; e-mail: lejninau@mail.ru

В настоящее время мошенники широко применяют технические средства для осуществления кражи конфиденциальных данных в интернете, например, использование URL в корыстных целях (спам, фишинг, скрытая загрузка опасного содержимого). При обнаружении вредоносных URL-адресов традиционные классификаторы сталкиваются с проблемами, так как количество различных URL огромно, шаблоны вредоносных сайтов меняются со временем, а корреляции между функциями усложняются. В статье обоснована опасность URL, которые ведут на вредоносные сайты, а также приведены типы вредоносных сайтов. Представлены существующие подходы по определению типа URL, их достоинства и недостатки. Рассмотрен процесс создания и тренировки модели, разделенный на этапы: получение и подготовка данных, обработка естественного языка, выбор и тренировка модели, прогнозирование. Проведен анализ эффективности полученной модели на реальных URL и сравнение с веб-сервисом VirusTotal.

Ключевые слова: *вредоносный URL, вирус, кибербезопасность, машинное обучение, фишинг, логистическая регрессия.*

USING MACHINE LEARNING TO DETECT MALICIOUS URL

O. I. Yevdoshenko, Yu. A. Lezhnina

Yevdoshenko Oleg Igorevich, Candidate of Technical Sciences, Associate Professor of the Department of Higher Mathematics and programming, MIREA - Russian Technological University, Moscow, Russian Federation;
Lezhnina Yuliya Arkadyevna, Candidate of Technical Sciences, Associate Professor, Deputy Director for Research, MIREA - Russian Technological University, Moscow, Russian Federation; e-mail: lejninau@mail.ru

Currently, fraudsters widely use technical means to steal confidential data on the Internet, for example, using URLs for personal gain (spam, phishing, hidden downloading of dangerous content). Traditional classifiers face challenges when detecting malicious URLs, as the number of different URLs is huge, malicious site patterns change over time, and correlations between features become more complex. The article substantiates the danger of URLs that lead to malicious sites, and also lists the types of malicious sites. The existing approaches to determining the type of URL, their advantages and disadvantages are presented. The process of creating and training a model is considered, divided into

stages: obtaining and preparing data, processing natural language, choosing and training a model, forecasting. The analysis of the efficiency of the obtained model on real URLs and comparison with the VirusTotal web service is carried out.

Keywords: *malicious URL, virus, cybersecurity, machine learning, fishing, logit model.*

Введение

Больше 50% населения Земли каждый день выходят в интернет, обмениваются сообщениями, читают новости, совершают покупки [1]. Обязательным атрибутом использования интернет-ресурсов является URL-адрес, который показывает путь к сайту или какой-либо конкретной странице на сайте. Большинство современных сайтов имеют вид «ИмяСайта.Домен», что обеспечивается службой DNS, иначе необходимо было бы указывать в адресной строке браузера полный IP-адрес сервера, на котором находится сайт. Появление данной службы было обусловлено желанием отказаться от трудно запоминаемых ip-адресов в пользу привычных букв, цифр и символа «-». Использование устаревших IP-адресов приводит к необходимости дополнительных исследований на уязвимости. Часть проблем при переходе сервисов на новый сайт аналогична проблемам унаследованных информационных систем [2]

Увеличение доступных для регистрации доменов, растущая популярность использования интернет-банкинга и онлайн-платежей, перенос деятельности магазинов в интернет, доминирование мобильного траффика [3] – все это привлекло внимание не только разработчиков и инвесторов, но и злоумышленников. Так, по статистике [4] только за первый квартал 2019 года более 113 млн. уникальных URL адресов были признаны вредоносными, при этом было обнаружено около 250 млн. вредоносных и потенциально нежелательных объектов. Опасность представляют также и прямые ссылки на файлы, так было найдено 905000 вредоносных установочных пакетов, 30000 инсталляционных пакетов для мобильных банковских троянов, 28000 инсталляционных пакетов для мобильных троянцев-вымогателей. Вредоносные ссылки для мобильных в основном распространяются в виде сокращенных URL, приводящих на вредоносный сайт или загрузку опасного ПО.

Существуют разные способы определения вредоносных URL, например, черный список, эвристика и машинное обучение [5]. Подходы, основанные на экспертной оценке для решения данной задачи применимы в меньшей степени [6]. Целью данной статьи является исследование эффективности применения машинного обучения для детектирования вредоносных URL.

Определение URL

Uniform Resource Locator (URL) – система унифицированных адресов электронных ресурсов, или единообразный определитель местонахождения ресурса. Стандарт URL закреплён в документе RFC 3986 [7]. Структура URL показана на рисунке 1.

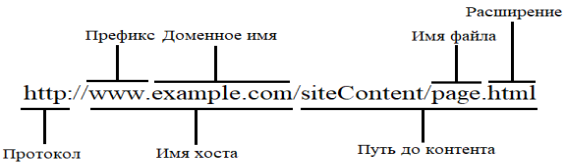

Рис. 1. Структура URL

Вредоносные URL

Можно выделить различные типы вредоносных URL-адресов в зависимости от нежелательного контента, на который они ведут. Каждый тип имеет свои собственные отличительные особенности, по которым может быть определен.

Спам

Веб-спам – это веб-сайты, которые пытаются обмануть алгоритмы поисковых систем, чтобы занимать при ранжировании более высокое место. Несмотря на то, что в выдаче такой сайт может занимать более высокое место, пользователь не найдет там никакой полезной информации по запросу [8].

Фишинг

Фишинг – вид интернет-мошенничества, цель которого – получить идентификационные данные пользователей [9].

Фишинг сайты мимикрируют под настоящие путем подделки контента и создания похожего доменного имени, чтобы пользователь ввел свои данные, предназначенные для оригинального сайта.

Drive by download атаки

Атаки Drive by Download относятся к вредоносным программам, которые скачиваются и устанавливаются на устройства без согласия пользователя [10].

Способы определения вредоносных URL

Существуют различные подходы и методы для анализа URL, на основе которых выносится решение о целесообразности дальнейшей работы с адресом.

Черный список

Черный список (ЧС) – это набор URL-адресов, которые были идентифицированы как вредоносные каким-то из способов (например, по многочисленным жалобам модераторов форумов). Обновляются такие списки автоматически или с помощью человека.

Главный плюс ЧС – это крайне низкая вероятность ошибки первого рода (False Positive, FP) [11], то есть безопасный URL был ошибочно принят как вредоносный.

Основным недостатком считается сложность контроля новых URL, что повышает вероятность ошибки второго рода (False Negative, FN),

когда вредоносная ссылка была ошибочно обозначена как не вредоносная.

Из-за сложности создания и поддержания ЧС в актуальном состоянии, они не создаются индивидуально администраторами сайтов, а используются в готовом виде на сторонних сервисах через API или иные интерфейсы [12].

Сервис ЧС, который будет использован в работе: Malware domain list [13].

Эвристический подход

При данном подходе хранятся функции, которые содержат поиск опасных URL по заданным паттернам, которые формируются признаками в адресе, что похоже на принцип работы черного списка, но позволяет детектировать угрозы в недавно сгенерированных URL [14].

Сам эвристический подход основывается на совокупности методов анализа, часть которых будет рассмотрена далее.

Лексический анализ

Лексический анализ основан на предположении, что вредоносные сайты имеют характерные паттерны. В основном это относится к фишинговым сайтам, где URL-адрес выглядит как исходный скопированный - настоящий URL-адрес имеет вид bank.io, а поддельный bank.qwe.acd.io, во втором случае адрес имеет большее количество поддоменов.

Главный плюс такого подхода в скорости, не требует больших затрат на хранение и внешних зависимостей.

Следует разделять лексический анализ на традиционный (описанный ранее) и расширенный [15].

Расширенный лексический анализ.

Анализ требует большее количество ресурсов. В исследованиях фишинговых сайтов было выявлено следующее [16]: для сокрытия конечного адреса применяются перенаправления, использование целевого имени домена в адресе хоста фишинг сайта, поддомены, грамматические ошибки. Используется анализ на основе данных хоста.

Контент-ориентированный анализ URL

Проверка HTML на количество рабочих гиперссылок; наличие iframe; соответствие навигации по страницах корневому домену. В JavaScript обращют внимание на функции: escape(), eval(), link(), unescape(), exec() [17].

PageRank

Алгоритм ранжирования PageRank (PR) применяется к списку документов, связанных гиперссылками, назначая каждому из них некоторое значение, измеряющее его "важность" среди остальных документов.

Среди доступных параметров популярность сайта, баллы в категориях, например, Spam Score, Domain Authority, Page Authority, Trust Metric [18].

Методы машинного обучения

Подходы в машинном обучении изначально включают эвристический подход для выявления признаков с влиянием на классификацию.

Применение машинного обучения делится на следующие этапы:

- получение и подготовка данных;
- применение обработки естественного языка;
- выбор и тренировка модели;
- прогнозирование.

Получение и подготовка данных

Использован датасет, содержащий 420 тыс. веб-адресов, из которых 75 тыс. являются вредоносными [19], часть показана на рисунке 2. Данные размечены: "good" – для безопасных сайтов, "bad" – для вредоносных.

420443	dataplues.com/quincy/pony/gate.php,bad
420444	60.250.76.52/,bad
420445	f4321y.com/,bad
420446	mykings.pw/,bad

Рис. 2. Фрагмент датасета

Под подготовкой данных подразумевается удаление NaN значений и дубликатов. Использованы методы notnull и drop_duplicates библиотеки Pandas [20]. Для визуализации результатов использована библиотека matplotlib [21].

Результат показан на рисунке 3.

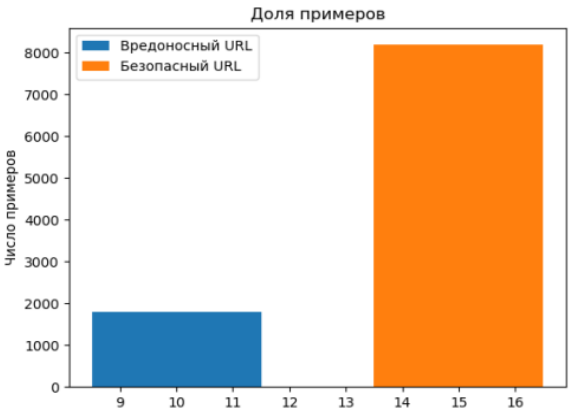

Рис. 3. Результат обработки данных

В результате обработки видно, что безопасных адресов значительно больше, это позволит уменьшить число ложных срабатываний из-за слов, общих для обеих категорий URL.

Применение обработки естественного языка

Алгоритмы обучения работают с числовым набором данных, поэтому необходимо преобразовать слова в числовые векторы. Самым популярным решением является применение модели Bag-of-words [22].

Перед преобразованием необходим токенизатор, который бы получал набор слов из URL для дальнейшего извлечения признаков путем парсинга данных по специальным символам (двоеточие, слеш, дефис и т. д.).

Результат работы токенизатора для https://puzzle-english.com/directory/tongue-twisters будет следующим: [b'https: ', 'puzzle', 'english.com', 'com', 'directory', 'twisters', 'english'].

Для каждого слова из набора модели указывается некоторый "вес".

Таким образом, модель текста представляет собой множество пар "слово – вес". Алгоритмы определения веса TF-IDF будут приведены далее. При этом веса могут присваиваться словам или основам слов.

Игнорирование семантических связей между словами – главный недостаток модели Bag-of-words. Также эта модель характеризуется экспоненциальным ростом сложности вычислений из-за увеличения размерности данных.

Вследствие недостатков Bag-of-words будет использована статистическая мера TF-IDF, используемая для оценки важности слова в контексте документа, являющегося частью коллекции документов или корпуса. Вес некоторого слова пропорционален частоте употребления этого слова в документе и обратно пропорционален частоте употребления слова во всех документах коллекции [23].

TF (term frequency – частота слова) – отношение числа вхождений некоторого слова к общему числу слов документа.

Результат определения TF для ссылки "https://puzzle-english.com/directory/tongue-twisters" с помощью токенизатора показан на рисунке 4.

```
Counter({'directory': 0.14285714285714285, "b'https:": 0.14285714285714285,
'com': 0.14285714285714285, "twisters'": 0.14285714285714285, 'english.com':
0.14285714285714285, 'puzzle': 0.14285714285714285, 'english': 0.14285714285
714285})
```

Рис. 4. Определение TF для URL

IDF (inverse document frequency – обратная частота документа) – инверсия частоты, с которой некоторое слово встречается в документах коллекции. TF-IDF является произведением значений TF и IDF.

Выбор и тренировка модели

Два наиболее популярных алгоритма для классификации – логистическая регрессия (LR) и метод опорных векторов (SVM) [24].

Основные отличия LR от SVM состоят в том, что LR выдает вероятностные значения, в то время как SVM – 1 или 0. Таким образом, LR не делает абсолютного прогноза и не предполагает, что данных достаточно для принятия окончательного решения [25]. Это главный критерий в пользу выбора LR, так как с помощью машинного обучения подразумевается получение вероятностной оценки за отсутствием высокой достоверности данных. Данная оценка будет использована совместно с другими методами, приведенными ранее.

Модель подсчитывает взвешенные суммы входных признаков (плюс член смещения), но взамен выдачи результата напрямую, она выдает логистику результата, представляющую из себя сигмовидную функцию (логистическую функцию). Сигмовидная функция – это S-образная кривая, которая может принимать любое действительное число и отображать его в значение в диапазоне от 0 до 1

Для тренировки модели были использованы модули LogisticRegression и train_test_split библиотеки scikit-learn [26].

С помощью train_test_split делится выборка на тренировочную и тестовую часть. Обучение происходит на тренировочной выборке, на тестовой - проверка полученных результатов. Ключевые входящие параметры: labels, features, df.index, которые содержат всю информацию о URL из датасета в формате «номер строки:категория URL», test_size равен 0,2, что соответствует отведению 20 % данных на тесты.

LogisticRegression получает на вход X_train, y_train, что соответствует матрице признаков URL и самих URL из датасета.

Фрагмент программного кода для тренировки модели показан на рисунке 5. "train accuracy" – это точность модели на примерах, на которых она была построена, "test accuracy" – это точность модели на примерах, которые ранее не были представлены.

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
vectorizer = TfidfVectorizer( tokenizer=Tokenise ,use_idf=True, smooth_idf=True, sublinear_tf=False)
features = vectorizer.fit_transform(df.url).toarray()
labels = df.label
model = LogisticRegression(random_state=0)
X_train, X_test, y_train, y_test, indices_train, indices_test = train_test_split(features, labels, df.index, test_size=0.20, random_state=0)
model.fit(X_train, y_train)
y_pred_proba = model.predict_proba(X_test)
y_pred = model.predict(X_test)
clf = LogisticRegression(random_state=0)
clf.fit(X_train,y_train)
train_score = clf.score(X_train, y_train)
test_score = clf.score(X_test, y_test)
print ('train accuracy =', train_score)
print ('test accuracy =', test_score)
```

Рис. 5. Листинг кода тренировки модели

"train accuracy" равно 0.9025, "test accuracy" равно 0.8815.

Необходимо провести анализ полученной модели с помощью оценки ошибок первого и второго рода [11].

Ошибкой первого рода является ложное срабатывание (FP), второго рода (FN) – неверно принятое решение.

Из scikit-learn для построения матрицы ошибок использован модуль confusion_matrix, для

визуализации библиотека seaborn и модуль heatmap для визуализации в формате тепловой карты [27]. Результат на рисунке 6.

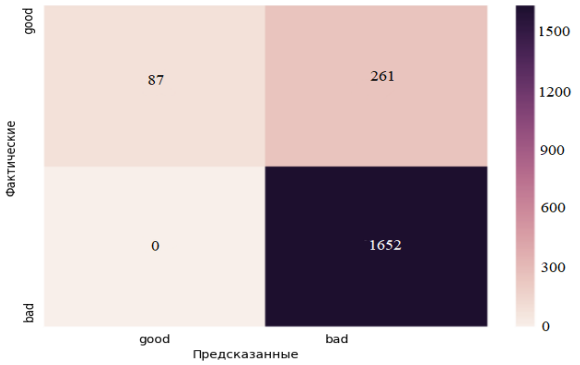

Рис. 6. Матрица ошибок

1652 опасных URL-адреса были правильно определены как вредоносные. Однако 261 безопасный URL-адрес был признан вредоносным. Эти «ложные срабатывания» ошибка FP. Ошибки такого рода являются одним из примеров того, почему конечная система выявления вредоносных ссылок не должна состоять только из методов машинного обучения.

Прогнозирование

При реальных испытаниях системы необходимо использовать данные, которые не входили в датасет, использованный при обучении. Адреса брались с сайта Malware domain list [13]. Тестовые адреса: googel.com, google.com (безопасный), photoscape.ch/Setup.exe (троян), amazon-sicherheit.kunden-ueberpruefung.xyz/ (фишинг). Код прогнозирования показан на рисунке 7.

```
X_predict = ['googel.com/', 'google.com/', 'photoscape.ch/Setup.exe/', 'amazon-sicherheit.kunden-ueberpruefung.xyz/']
X_predict = vectorizer.transform(X_predict)
y_Predict = clf.predict(X_predict)
print(y_Predict)
```

Рис. 7. Листинг кода для проверки модели

Результаты: bad, good, bad, bad. За исключением первого адреса, остальные были определены верно. Ошибка в первом случае объясняется схожестью googel с google, модель определила, что данный адрес может быть фишинговым. На самом деле googel.com перенаправляет на google.com и является безопасным.

В таблице 1 представлено сравнение модели с веб-сервисом VirusTotal, который использует антивирусные решения [28].

Таблица 1

Сравнение модели с VirusTotal

URL	Вердикт модели	VirusTotal
googel.com	bad	No engines detected this URL
google.com	good	No engines detected this URL
photoscape.ch/Setup.exe	bad	8 engines detected this URL (Malware)
amazon-sicherheit.kunden-ueberpruefung.xyz	bad	4 engines detected this URL (Malware)

VirusTotal во всех случаях выдал правильные результаты.

Заключение

Целью данной статьи является исследование эффективности применения машинного обучения для детектирования вредоносных UR. В ходе статьи была описана актуальность тематики исследования, рассмотрены типы вредоносных сайтов и популярные подходы к их детектированию, процесс создания модели, начиная от получения датасета до прогнозирования.

Исходный датасет имел более 400 тысяч адресов, для обработки естественного языка использовался алгоритм TF-IDF, для классификации – логистическая регресиия.

Train accuracy составила свыше 90 %, test accuracy свыше 88 %. Ложное срабатывание произошло в 261 случае.

Как видно из результатов, модель нуждается в доработке, например, увеличении начального датасета, корректировке весов вручную, а конечным продуктом должна быть система, которая использует комплексный подход, состоящий не только из машинного обучения, но и из эвристического подхода и черного списка, также возможно использование API сторонних антивирусных сервисов, например, VirusTotal.

Список литературы

1. Internet statistic // BroadbandSearch. – Режим доступа: https://www.broadbandsearch.net/blog/internet-statistic (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

2. Зарипова В. М. Унаследованные информационные системы. Проблемы и решения / В. М. Зарипова, И. Ю. Петрова // Инженерно-строительный вестник Прикаспия. – 2022. – № 2 (40). – С. 150–158. – DOI 10.52684/2312-3702-2022-39-1-150-158.

3. Вся статистика интернета на 2020 год — цифры и тренды в мире и в России // WebCanape. – Режим доступа: https://www.web-canape.ru/business/internet-2020-globalnaya-statistika-i-trendy/ (дата обращения: 23.01.2021), свободный. – Заглавие с экрана. – Яз. рус.

4. IT threat evolution Q1 2019. Statistics // SECURELIST by Kaspersky. – Режим доступа: https://securelist.com/it-threat-evolution-q1-2019-statistics/90916/ (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

5. Dhanalakshmi Ranganayakulu, Chellappan C Detecting Malicious URLs in E-mail – An Implementation // AASRI Procedia. – 2013. – №10. – С. 125–131.

6. Гостюнина В. А. Способ экспертной оценки веб-контента на основе модели систем репутаций / В. А. Гостюнина, Н. В. Давидюк, Ю. А. Гостюнин // Инженерно-строительный вестник Прикаспия. – 2018. – № 3 (25). – С. 41–44.

7. Rfc3986 // IETF Tools. – Режим доступа: https://tools.ietf.org/html/rfc3986 (дата обращения: 223.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

8. What Is Search Engine Spam? // Search Engine Land. – Режим доступа: https://searchengineland.com/what-is-search-engine-spam-the-video-edition-15202 (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

9. Что такое «фишинг» // ИТ-энциклопедия Касперского. – Режим доступа: https://encyclopedia.kaspersky.ru/knowledge/what-is-phishing/ (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

10. What Is a Drive by Download // Kaspersky. – Режим доступа: https://www.kaspersky.com/resource-center/definitions/drive-by-download (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

11. ГОСТ Р 50779.10-2000. Статистические методы. Вероятность и основы статистики. Термины и определения. – Дата введения 2001–07–01 // Кодекс. – Режим доступа: https://docs.cntd.ru/document/1200017686, свободный. – Заглавие с экрана. – Яз. рус.

12. A Brief, Opinionated History of the API // InfoQ. – Режим доступа: https://www.infoq.com/presentations/history-api/ (дата обращения: 23.01.2021), свободный. – Заглавие с экрана. – Яз. рус.

13. Malware Domain List. – Режим доступа: https://www.malwaredomainlist.com/mdl.php (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

14. Jin Lee Lee. Heuristic-based Approach for Phishing Site Detection Using URL Features / Jin Lee Lee, Dong Hyun Kim, Lee Chang Hoon // CEET – 2015 : Proceedings of the Third International Conference on Advances in Computing, Electronics and Electrical Technology – Kuala Lumpur, Malaysia : CEET, 2015. – С. 131–135.

15. Sahoo Doyen. Malicious URL Detection using Machine Learning: A Survey / Sahoo Doyen, Liu Chenghao, C. H. Hoi Steven, 2017. – Режим доступа: https://arxiv.org/pdf/1701.07179.pdf, свободный. – Заглавие с экрана. – Яз. рус.

16. Garera S. A framework for detection and measurement of phishing attacks / S. Garera, N. Provos, M. Chew, A. Rubin // Proceedings of the 2007 ACM workshop on Recurring malcode. – New York, NY, United States : Association for Computing Machinery, 2007. – С. 1–8.

17. Manan W. Intelligent Computing & Optimization / W. Manan, A. Ghani, M. Nizam. – 2 изд., доп. – California, USA : Springer International Publishing, 2019. – 575 с.

18. Althobaiti K. Vaniea A Review of Human-and Computer-Facing URL Phishing Features / K. Althobaiti, G. Rummani, K. Vaniea // European Workshop on Usable Security. – Stockholm, Sweden : IEEE, 2019.

19. Using machine learning to detect malicious urls // GitHub. – Режим доступа: https://github.com/VAD3R-95/Malicious-Url-Detection (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

20. Pandas documentation // Pandas. – Режим доступа: https://pandas.pydata.org/pandas-docs/stable/index.html (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

21. Matplotlib. – Режим доступа: https://matplotlib.org/ (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

22. Hanna M. Wallach. Topic Modeling: Beyond Bag-of-Words / Hanna M. Wallach // ICML '06 : Proceedings of the 23rd international conference on Machine learning. – Cambridge, UK : Cavendish Laboratory, 2006. – С. 977–984.

23. Jones K. A statistical interpretation of term specificity and its application in retrieval / K. Jones // Journal of Documentation. – Cambridge, UK : MCB University Press, 2004. – С. 493–502.

24. A tour of the top 10 algorithms for machine learning // Towards Data Science. – Режим доступа: https://towardsdatascience.com/a-tour-of-the-top-10-algorithms-for-machine-learning-newbies-dde4edffae11 (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

25. Орельен Ж. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем / Ж. Орельен. – 2 изд., доп. - Киев, Украина : Диалектика-Вильямс, 2018. – 688 с.

26. Scikit-learn // Machine Learning in Python. – Режим доступа: https://scikit-learn.org/stable/index.html (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

27. Scikit-learn // Seaborn: statistical data visualization. – Режим доступа: https://seaborn.pydata.org/ (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

28. VirusTotal. – Режим доступа: https://www.virustotal.com/ (дата обращения: 23.01.2023), свободный. – Заглавие с экрана. – Яз. рус.

Ссылка для цитирования:

Евдошенко О. И., Лежнина Ю. А. Применение методов машинного обучения для детектирования вредоносных URL // Инженерно-строительный вестник Прикаспия : научно-технический журнал / Астраханский государственный архитектурно-строительный университет. Астрахань : ГАОУ АО ВО «АГАСУ», 2023. № 2 (44). С. 122–127.