DOI 10.37882/2223-2966.2023.05.29

РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ АНАЛИЗА САЙТОВ НА ПРЕДМЕТ УТЕЧКИ ПЕРСОНАЛЬНЫХ ДАННЫХ

WEBSITE ANALYSIS SOFTWARE DEVELOPMENT FOR THE LEAKAGE OF PERSONAL DATA

D. Purtov
V. Purtov
K. Shmitko
A. Rusakov
A. Melnikov
V. Filatov

Summary: This article presents a study on developing a software tool, Web-PD-Scanner, which aims to analyze web pages in HTML format to detect potential personal data leakage. The article provides an overview of modern software tools for parsing web resources, as well as a review of HTML-page parsing technologies and their limitations. The relevance of the proposed study is substantiated, and the object, subject of research, scope, and limitations of the software are defined. The main tasks to be performed by the software are formulated, and various mathematical methods, algorithms, and software tools that can be used to develop the Web-PD-Scanner software are identified. The article concludes that a hybrid approach that combines rule-based algorithms and machine learning is the most effective solution for detecting leaks of personal data on websites. The next stage of the research involves defining a model for storing aggregated personal data and selecting specific methods and algorithms for developing the Web-PD-Scanner software. This study provides valuable insights for researchers and practitioners interested in developing software tools for analyzing web pages for personal data leakage.

Keywords: web scraping, data mining, HTML parsing, personal data protection, software development.

Пуртов Даниил Владимирович
МИРЭА — Российский технологический университет
danpurtov@gmail.com
Пуртов Владимир Сергеевич
арт-директор, ООО «Элотра»
purtovdv3176@gmail.ru
Шмитько Кирилл Андреевич
МИРЭА — Российский технологический университет
shmitkokirill@gmail.com
Русаков Алексей Михайлович
старший преподаватель,
МИРЭА — Российский технологический университет
rusal@bk.ru
Мельников Алексей Олегович
доцент, МИРЭА — Российский технологический университет
melnikov.aleksey@gmail.com
Филатов Вячеслав Валерьевич
доцент, МИРЭА — Российский технологический университет
filv@mail.ru

Аннотация: В данной статье представлено исследование по разработке программного инструмента Web-PD-Scanner, предназначенного для анализа веб-страниц в формате HTML с целью обнаружения потенциальной утечки персональных данных. В статье представлен обзор современных программных средств для парсинга веб-ресурсов, а также обзор технологий парсинга HTML-страниц и их ограничений. Обосновывается актуальность предлагаемого исследования, определяются объект, предмет исследования, область применения и ограничения программного обеспечения. Сформулированы основные задачи, решаемые программным обеспечением, и определены различные математические методы, алгоритмы и программные средства, которые могут быть использованы для разработки программного обеспечения Web-PD-Scanner. В статье делается вывод о том, что гибридный подход, сочетающий алгоритмы на основе правил и машинное обучение, является наиболее эффективным решением для обнаружения утечек персональных данных на веб-сайтах. Следующий этап исследования предполагает определение модели хранения агрегированных персональных данных и выбор конкретных методов и алгоритмов разработки программного обеспечения Web-PD-Scanner. Это исследование предоставляет ценную информацию для исследователей и практиков, заинтересованных в разработке программных инструментов для анализа веб-страниц на предмет утечки личных данных.

Ключевые слова: скраппинг веб-страниц, интеллектуальный анализ данных, синтаксический анализ HTML, защита персональных данных, разработка программного обеспечения.

Глобальная сеть Интернет (далее — интернет) — важнейший источник информации для всех сфер жизни: СМИ, коммерции, образования, развлечений, медицины и других. Кроме визуального доступа к информации в интернете, для различных задач требуется автоматизированная загрузка и обработка — веб-агрегация данных.

Примерами задач веб-агрегации данных являются следующие:

а) новостные агрегаторы (Яндекс Дзен) — электронные СМИ, публикующие новостные статьи из различных источников;

б) электронные торговые площадки (Ali Express, eBay) — онлайн-рынки, выступающие посредниками в торговых сделках;

в) исследовательские и аналитические проекты, собирающие однотипные данные с целью их сопоставления и исследования.

В процессе веб-агрегации могут возникнуть проблемы технической реализации:

— загрузка данных требует много времени;
— информация может быть разнородной, то есть, иметь различные форматы в разных источниках, что требует предварительной унификации данных перед их обработкой;
— интерфейсы для загрузки данных из разных источников могут отличаться, что требует реализации отдельного программного модуля для каждого источника.

Таким образом, в связи с постоянным и быстрым ростом объема информации в интернете, актуальной становится проблема веб-агрегации данных.

В настоящей дипломной работе по теме «Разработка программного обеспечения для анализа сайтов на предмет утечки персональных данных» предлагается способ автоматической агрегации данных из открытых веб-ресурсов, позволяющий решить перечисленные выше проблемы программной реализации.

Основные понятия

Бот — программа-робот, выполняющая какие-либо рутинные процедуры по сбору данных или автоматизированному ведению диалога с пользователями.

Веб-агрегация данных — процесс загрузки и обработки разнородных данных из веб-ресурсов.

Веб-ресурс — ресурс, размещенный в интернете: портал, сайт, веб-служба, база данных, позволяющий каким-либо образом загружать из него данные, в том числе, непосредственно веб-страницу в формате HTML.

Капча — защитная функция на сайте, позволяющая идентифицировать пользователя как человека, а не бота, для доступа к определенным данным или функциям.

Парсер — программа (программный модуль, скрипт), выполняющий парсинг данных.

Парсинг — обработка данных, загруженных из веб-ресурса: синтаксический анализ текстового документа с целью преобразования в структурированный формализованный вид (например, парсинг HTML в JSON).

Персональные данные — личные сведения о каком-либо физическом или юридическом лице, которые могут предоставляться другим лицам.

HTML-парсинг — парсинг веб-страниц в формате HTML.

Парсить — выполнять парсинг.

Скрипт — короткая программа, сценарий, обычно для загрузки и выполнения на клиентском устройстве.

Парсинг сайтов. Общие понятия о парсинге сайтов

Парсинг сайтов позволяет компаниям автоматизировать процессы веб-агрегации данных в интернете, используя ботов или автоматические скрипты, называемые «обходчиками» веб-страниц, автоматическими сборщиками данных или веб-сборщиками (web crawlers). В связи с этим, парсинг сайтов иногда называют «обходом (сканированием) интернета» или «скрейпингом данных».

Процесс парсинга веб-сайтов включает в себя отправку запросов на получение веб-страницы и извлечение из нее машиночитаемой информации, то есть синтаксический анализ разметки HTML и преобразование ее в структурированный формализованный вид.

Цели парсинга

Цели парсинга сайтов могут быть различными:

а) технический анализ сайта в процессе SEO-оптимизации;

б) анализ семантической информации сайта ботами поисковых систем;

в) анализ семантической информации сайта ботами агрегаторов;

г) анализ сайта в рекламных и бизнес-целях;

д) технический анализ сайта для проведения каких-либо исследований.

Рассмотрим некоторые из них.

Технический парсинг сайта, которым в основном пользуются SEO-специалисты, используется для анализа работы сайта по различным критериям:

— поиск ошибок в разметке HTML;
— поиск неверных ссылок и некорректных редиректов;
— выявление проблем с мета-тегами и заголовками 1;
— анализа корректности содержимого файла robots.txt;
— проверка микроразметки на сайте;
— обнаружение нежелательных страниц, которые открыты для индексации;
— анализ размеров изображений, скриптов и скорости загрузки страниц;
— прочие технические задачи.

На основе полученных данных специалист составляет технические задания для устранения выявленных проблем с сайтом.

Анализ сайта в рекламных и бизнес-целях может включать следующие задачи:

— сбор информации об ассортименте конкурентов;
— парсинг названий товаров, артикулов, цен и прочего для наполнения своего собственного интернет-магазина. Это может быть как разовая задача, так и на основе регулярного мониторинга;
— анализ структуры сайтов-конкурентов с целью улучшения и развития собственной структуры.

Алгоритм парсинга

Парсинг выполняется с помощью специальных скриптов или программных модулей.

Обобщенный алгоритм парсинга сайта (см. Рисунок 1) включает следующие действия:

1) поиск необходимых данных в исходном файле;
2) извлечение данных с отделением от программного кода;
3) формирование структурированного отчета из выделенных данных.

Рис. 1. Общий вид процесса парсинга веб-страницы

Отчет обычно сохраняется в виде JSON или CSV-файла.

Парсинг выполняется чаще всего на основе XPath-запросов — технологии, которая позволяет обращаться к определенному участку кода страницы и извлекать из него информацию по заданному критерию.

Обзор современных парсеров данных

В ходе анализа рынка существующих программных средств, выполняющих парсинг и веб-агрегацию данных, были выделены следующие, обзор которых представлен ниже.

Веб-парсер Octoparse

Octoparse — бесплатный, но функциональный веб-парсер, который используется для обработки различных типов данных из веб-ресурсов. Он позволяет извлекать данные с сайтов со сложной выдачей блоков данных, которые используют собственные встроенные инструменты Regex. Данный парсер использует инструмент XPath, а также собственные прокси-серверы с автоматической сменой IP-адресов, позволяющие обходить блокираторы.

Octoparse предлагает использовать готовые шаблоны для парсинга популярных сайтов: Amazon, Yelp, Tripadvisor и т.д.

Достоинствами Octoparse являются:

— большой функционал парсинга;
— возможность бесплатного использования;
— простота настройки параметров парсинга (достаточно указать URL страницы и ключевые слова для поиска).

Сервис Scraper API

Scraper API предоставляет платный прокси-сервис, предназначенный для парсинга данных из веб-ресурсов.

Данный парсер также использует тысячи собственных и сторонних прокси-серверов с автоматической сменой IP-адресов, позволяющие обходить блокираторы.

Работа с парсером производится через API. Для загрузки HTML страницы вызывается соответствующая функция API с указанием URL страницы.

Scraper API позволяет обрабатывать некоторые виды капчи.

Данный парсер используется обычно для мониторинга цен конкурентов, билетов, парсинга социальных сетей и т.д.

Программа для парсинга ScrapingHub

Scrapinghub — платная облачная программа для парсинга, использующая инструмент Crawlera — смарт-прокси-ротатор с функциями обхода защиты от ботов.

Работа с парсером производится с помощью API.

Scrapinghub предоставляет набор инструментов, каждый из которых можно оплатить отдельно.

Платформа Mozenda

Mozenda — это корпоративная облачная платформа для парсинга. Она состоит из двух частей: приложения для создания проекта извлечения данных и веб-консоли для запуска агентов, организации результатов и экспорта данных.

Mozenda также предоставляет доступ к API для получения данных и имеет встроенные интеграции с системами хранения и обмена файлов, такими как FTP, Amazon S3, Dropbox и другими. Данные экспортируются в форматы CSV, XML, JSON или XLSX.

Mozenda позволяет эффективно обрабатывать большие объемы данных.

Использование данной платформы требует обладания навыками программирования выше базовых.

Инструмент ScrapingBee

ScrapingBee — это инструмент для парсинга данных из веб-ресурсов на основе API, поддерживающий библиотеки для различных языков программирования.

Данный парсер использует динамическое переключение прокси-серверов, временные задержки и запросы без заголовков, имитируя работу браузера, с целью обхода блокировок от ботов.

Динамическая смена прокси-серверов позволяет обеспечивать высокую скорость доступа.

Инструмент ParseHub

ParseHub — это программный инструмент в виде расширения для браузера Firefox с несложным графическим интерфейсом, позволяющий извлекать данные с сайтов и сохранять их в структурированном виде в формате JSON, CSV или Google Sheets.

Визуализацию данных, полученных с помощью данного парсера, можно выполнять в Tableau.

ParseHub может обрабатывать интерактивные карты, календари, поиск, форумы, вложенные комментарии, бесконечную прокрутку, аутентификацию, выпадающие списки, формы, Javascript, Ajax и др.

Бесплатная версия парсера имеет ограничение в пять проектов с 200 страницами за запуск. Платная подписка обеспечивает 20 проектов с 10000 страниц на сканирование и ротацию IP.

Инструмент Easy Web Extract

Easy Web Extract (компании Web2Mine) — парсер данных из веб-ресурсов, разработанный на технологии .NET, позволяет применять встроенные скрипты преобразования данных (C#, VB, JS).

Easy Web Extract позволяет экспортировать выделенные данные файлы следующих типов: CSV (Excel), TXT (текст), XML, HTML, MS Access DB, SQL, файл сценария MySQL, форму отправки HTTP и источник данных ODBC.

Easy Web Extract позволяет выполнять парсинг одновременно нескольких страниц, контента Ajax / JS.

Особенностью данного парсера является возможность генерации формы HTTP из данных сайта для использования формы в последующих запросах.

Обзор технологий парсинга HTML-страниц. Понятие парсинга HTML-страниц

HTML — язык разметки, используемый для создания веб-страниц. Парсинг HTML-страницы включает в себя извлечение данных из HTML-документа, которые могут содержать текст, изображения, ссылки и другие элементы.

Существуют различные методы и технологии, используемые для анализа HTML-страницы.

Технология DOM

Одним из наиболее часто используемых методов анализа HTML является подход с использованием объектной модели документа — DOM. Это независимый от платформы API, который представляет HTML-документ в виде дерева объектов. В древовидной структуре DOM можно перемещаться и управлять ею с помощью таких языков программирования, как JavaScript или Python. Подход DOM очень эффективен для анализа HTML-страниц и управления ими, поскольку обеспечивает стандартное иерархическое представление документа. [1]

Метод SAX

Другим методом анализа HTML является метод SAX (Simple API for XML). Этот подход считывает HTML-документ как поток событий, что позволяет обрабатывать документ в режиме реального времени. Подход

SAX эффективен для больших HTML-документов, поскольку он не загружает весь документ в память сразу. Вместо этого он обрабатывает документ последовательно, генерируя события для каждого обнаруженного элемента. [3]

Инструмент XPath

XPath — это язык запросов для XML-документов, который используется для навигации и выбора определенных элементов на HTML-странице. Синтаксис XPath позволяет выполнять сложные запросы, которые выбирают элементы на основе их атрибутов или положения в документе.[7] Использование XPath эффективно при анализе больших документов HTML или сложных структур HTML.

Регулярные выражения

Другой популярный подход к анализу HTML — использование регулярных выражений. Регулярные выражения представляют собой последовательность символов, определяющую шаблон поиска, и позволяют выполнять поиск конкретных шаблонов или элементов на HTML-странице. Регулярные выражения можно использовать для извлечения такой информации, как номера телефонов, адреса электронной почты и т.п.

Библиотека BeautifulSoup

BeautifulSoup — это библиотека Python, используемая для парсинга документов HTML и XML. Данная библиотека эффективна особенно при обработке плохо отформатированных HTML-страниц.[4]

Машинное обучение

Для анализа HTML страниц также применяются методы нейросетей с машинным обучением, позволяющие идентифицировать определенные элементы или шаблоны на HTML-странице.[2] Алгоритмы машинного обучения особенно эффективны при обработке больших объемов данных и могут использоваться для таких задач, как анализ тональности, распознавание изображений и классификация текста.

Проблемы парсинга HTML-страниц

Одной из проблем парсинга HTML-страниц является изменчивость HTML-кода. Различные сайты имеют отличающуюся структуру HTML-страниц, страницы на одном сайте могут постоянно меняться в результате доработок, могут изменяться версии HTML и CSS. Это затрудняет создание универсального подхода к синтаксическому анализу.

Для решения данной проблемы используются комбинация различных технологий парсинга одной и той же HTML-страницы, например: регулярные выражения — для выделения определенных шаблонов, XPath — для навигации по структуре HTML и алгоритмы машинного обучения — для идентификации изображений.

Постановка задачи

С ростом объемов информационных ресурсов и количества онлайн-площадок одновременно растет и число уязвимостей информации, в первую очередь персональных данных. Сегодня интернет становится всё менее анонимным, и почти все веб-ресурсы требуют регистрации и указания персональных данных. Чаще всего это имя, фамилия, номер телефона и адрес электронной почты. При покупках в интернете нам приходится указывать все реквизиты банковских карт, а для доставки — домашний адрес. Веб-ресурсы, выполняющие финансовые операции, и транспортные компании требуют, в соответствии с законодательством, указания паспортных данных. Веб-ресурсы, предоставляющие государственные услуги, требуют многих других персональных данных и реквизитов личных документов.

С тотальным распространением ресурсов, обладающих правами на обработку персональных данных, возрастает риск их утечки. Не все веб-ресурсы имеют надежные средства защиты информации. Но и защищаемые базы данных могут быть подвержены хакерским атакам. Помимо факторов безопасности цифрового доступа к данным, существуют и другие факторы риска, например, социальный: сотрудник компании, имеющий доступ к данным, может их выкрасть с целью продажи или другой целью.

Кроме небольшого, но неприятного ущерба для конкретной личности, данные стали предметом утечки, нарушение безопасности информации может иметь очень большие последствия, если утекли данные многих пользователей (иногда это миллионы), в том числе, для компании, допустившей утечку.

В зависимости от степени ущерба финансовые и репутационные убытки компании могут быть колоссальными. Например, в 2020 году компании Facebook (США) присудили пять миллиардов долларов компенсации за утечку данных десятков миллионов клиентов. Этот пример демонстрирует так же и то, что даже веб-ресурсы крупнейших компаний могут быть уязвимы.

В связи с этим, важность анализа веб-ресурсов (в первую очередь, сайтов) на предмет потенциальной утечки персональных данных постоянно возрастает.

Анализ сайтов на предмет утечек является первым этапом в системе обеспечения информационной без-

опасности веб-ресурсов. Анализ сайтов, также, позволяет предотвратить возможные утечки данных, заведомо выявить потенциальные уязвимости системы безопасности.

В результате эффективного анализа сайта на предмет утечек персональных данных владелец сайта может своевременно принять меры по устранению уязвимостей или улучшению системы безопасности информации. Это прежде всего позволит уберечь владельца и пользователей сайта от значительных финансовых убытков и репутационного ущерба.

Регулярные аналитические проверки веб-ресурсов на предмет потенциальной утечки персональных данных позволит выявить целостную картину уязвимостей информации такого рода, на основе которой станет возможным определение закономерностей, системных ошибок средств защиты информации. Данная деятельность может стать основой корректирования существующих и введения новых правил, стандартов и других законодательных документов, комплексных регулирующих мер, направленных на защиту персональных данных.

Таким образом, можно сделать вывод, что анализ сайтов на предмет утечки персональных данных является критически важным вопросом, имеющим серьезные последствия для отдельных лиц, организаций и общества в целом. Данный анализ позволяет предпринять упреждающие шаги для защиты данных, минимизировать потенциальный ущерб и способствовать созданию более безопасной онлайн-среды.

Целью настоящей статьи является описание процесса разработки и исследования ПО «Веб-ПД-Сканер», предназначенного для анализа веб-страниц в формате HTML на предмет утечки персональных данных.

Целью исследований ПО является определение исходных технических данных для разработки программных средств, предназначенных для защиты и предотвращения несанкционированного доступа к персональным данным.

ПО «Веб-ПД-Сканер» должно предоставлять инструменты для снижения рисков утечки и защиты личных данных. Разработка ПО включает соблюдение правил и стандартов конфиденциальности, таких как GDPR и CCPA, и обеспечение безопасного интерфейса пользователя.

ПО может включать разработку новых алгоритмов, библиотек и фреймворков для синтаксического анализа HTML и анализа данных, которые можно интегрировать в программные приложения.

ПО может, также, включать разработку новых методов визуализации данных, которые помогут разработчикам сайтов лучше понять потенциальные риски утечки данных и способы их устранения.

Исследование посвящено проектированию, разработке и внедрению программных инструментов и методов, предназначенных для защиты персональных данных и предотвращения несанкционированного доступа или кражи личной информации.

Таким образом, объектом исследования настоящей работы является процесс разработки ПО «Веб-ПД-Сканер», предназначенного для анализа веб-страниц в формате HTML на предмет утечки персональных данных.

Предметом исследования настоящей работы является ПО «Веб-ПД-Сканер», предназначенное для анализа веб-страниц в формате HTML на предмет утечки персональных данных.

Разработанные алгоритмы, библиотеки и фреймворки для синтаксического анализа HTML и анализа данных, методы визуализации данных, интерфейс пользователя, входящие в состав ПО, также являются предметом исследования.

Область применения и ограничения

Область применения ПО «Веб-ПД-Сканер» зависит от следующих ограничительных факторов.

1. Тип персональных данных.

Ограничение по типу персональных данных определяет набор данных, которые выделяет и обрабатывает ПО: номер телефона, адрес электронной почты, номер кредитной карты и т.д.

2. Структура веб-сайта.

Возможности анализа HTML-страниц со сложной структурой, например, с динамических контентом или запросами AJAX, могут быть ограничены.

3. Хранение данных и безопасность.

Разрабатываемое ПО может не обеспечивать доступ к данным, которые хранятся в неподдерживаемом формате, зашифрованы или защищены иным образом.

4. Соответствие нормативным требованиям.

ПО должно быть разработано с учетом соответствующих правил и стандартов конфиденциальности, таких как GDPR и CCPA, и его применимость может быть ограничена, если оно не соответствует этим требованиям.

5. Ограничения вычислительных ресурсов.

ПО может быть ограничено в своей применимости, если для его эффективной работы требуются значительные вычислительные ресурсы или специализированное оборудование.

Перечисленные факторы ограничивают область применения ПО «Веб-ПД-Сканер», зависят от реализации ПО, структуры анализируемых сайтов и данных, но не влияют на его функционирование и основную цель — анализ веб-страниц в формате HTML на предмет утечки персональных данных.

Формализованная постановка задачи

ПО «Веб-ПД-Сканер» предназначено для предотвращения потенциальной утечки персональных данных на сайтах вследствие уязвимостей в системе безопасности сайта или непреднамеренного раскрытия конфиденциальной информации путем анализа веб-ресурса.

Основными проблемами для функционирования ПО при этом являются:

а) динамический контент сайтов;

б) наличие запросов AJAX;

в) другие методы и технологии представления веб-страниц, затрудняющие анализ HTML-страниц и данных.

При разработке ПО необходимо учитывать правила и стандарты конфиденциальности, такие как GDPR и CCPA, предъявляющие требования к сбору, хранению и обработке персональных данных. Для этого следует включить разработку и внедрение функций по защите информации, таких как шифрование, минимизация данных и контроль доступа.

Предлагаемые методы решения

В результате проведенных исследований можно выделить следующие математические методы, алгоритмы и программные средства, которые могут быть использованы для разработки ПО «Веб-ПД-Сканер», предназначенного для анализа веб-страниц в формате HTML на предмет утечки персональных данных:

а) регулярные выражения — метод анализа HTML-страниц для извлечения необходимой информации путем сопоставления шаблонов в тексте. Недостатком их является сложность для написания и отсутствие возможности масштабирования для больших наборов данных;

б) библиотеки анализа HTML (например, Beautiful Soup и HtmlAgilityPack), предназначенные для анализа HTML-страниц на различных языках программирования и предоставляющие функции для навигации по объектной модели HTML-документа (DOM) и фильтрации данных на основе определенных критериев;

в) обработка естественного языка (NLP) — это область искусственного интеллекта, которая фокусируется на взаимодействии между естественным языком и компьютерами. Алгоритмы NLP можно использовать для идентификации и извлечения соответствующей информации из текста, включая персональные данные.[5] Недостатком алгоритмов NLP является потребность значительного объема данных, и возможность неточного определения для отдельных типов текста;

г) алгоритмы машинного обучения (ML) можно использовать для выявления потенциальных рисков утечки данных путем выявления закономерностей в данных, которые указывают на потенциальные уязвимости в безопасности сайта. Алгоритмы ML можно обучать на наборе данных известных уязвимостей, чтобы выявлять похожие закономерности в новых данных, а также использовать для классификации данных на основе определенных критериев, таких как идентификация персональных данных. Недостатком алгоритмов ML является потребность в значительных вычислительных ресурсах и объемах данных;

д) алгоритмы интеллектуального анализа данных можно использовать для выявления шаблонов в больших наборах данных, включая HTML-страницы. Алгоритмы интеллектуального анализа можно использовать для выявления общих шаблонов персональных данных, выбросов в данных, которые могут указывать на потенциальные риски утечки.[6] Недостатком алгоритмов интеллектуального анализа является потребность в значительных вычислительных ресурсах и, кроме того, они могут быть неэффективны для некоторых типов данных;

е) алгоритмы кластеризации можно использовать для группировки данных веб-страниц на основе определенных критериев, а также для выявления выбросов в данных, которые могут указывать на потенциальные риски утечки данных. Недостатком алгоритмов интеллектуального анализа является потребность в значительных вычислительных ресурсах;

ж) алгоритмы анализа тональности можно использовать для определения тона и тональности текста, в частности, для выявления текста, который может указывать на потенциальные риски утечки данных, например негативные комментарии о безопасности сайта или личных данных. Однако алгоритмы анализа тональности могут быть неточными для всех типов текста, и для их эффективности может потребоваться значительный объем данных;

з) алгоритмы сетевого анализа можно использовать для анализа структуры сетей сайтов и выявления потенциальных рисков утечки данных путем выявления доменов сайтов, которые могут быть связаны с утечкой данных, или для выявления ссылок на сайты, которые могут привести к потенциальным рискам Недостатком алгоритмов сетевого анализа является потребность в значительных вычислительных ресурсах;

Таким образом, существует несколько математических методов, алгоритмов и программных средств, которые можно использовать для разработки ПО «Веб-ПД-Сканер». Очевидно, что наиболее эффективным решением для обнаружения утечек персональных данных на сайтах является сочетание алгоритмов на основе правил и машинного обучения.

Выбор метода будет зависеть от конкретных требований к ПО и характеристик анализируемых данных.

Заключение

В данном разделе описаны результаты исследований методов, алгоритмов и средств парсинга веб-ресурсов.

В результате проведенных исследований установлено, что наиболее эффективным решением для обнаружения утечек персональных данных на сайтах является гибридный подход, сочетающий алгоритмы на основе правил и машинного обучения.

На следующем этапе предполагается определить модель для хранения агрегированных персональных данных и выбрать конкретные методы и алгоритмы для разработки ПО «Веб-ПД-Сканер» анализа веб-страниц в формате HTML на предмет утечки персональных данных.

ЛИТЕРАТУРА

1. Гладкова Е.С. Обзор методов парсинга веб-страниц // Современные проблемы науки и образования. 2018. № 6. С. 120–123.
2. Сорокин А.Н., Родионов Д.А. Применение нейросетей и машинного обучения для анализа содержания веб-страниц // Современные информационные технологии и ИТ-образование: Сборник научных трудов. 2018. Т. 14. №. 2. С. 52–57.
3. Камаев В.Н., Попова И.В. Анализ методов обработки HTML-документов // Научно-технический вестник информационных технологий, механики и оптики. 2020. Т.20, № 6. С. 1177–1182.
4. Сагдеев, Р.В., & Газизов, Р.Ш. (2019). Инструменты и методы парсинга web-страниц. Электронный научный журнал, 9(3), 15–24.
5. Кузнецова Н.В. Извлечение персональных данных из текстовых источников // Проблемы современной науки и образования. 2017. №. 6-2. С. 272–276.
6. Кузнецов, А.В. Применение методов анализа данных для выявления персональных данных на веб-страницах [Электронный ресурс] / А.В. Кузнецов, И.М. Петров, Д.В. Кравченко // Молодежь и наука: актуальные вопросы науки и образования: сборник статей по материалам II международной (интернет) конференции молодых ученых и студентов. Хабаровск, 2020. С. 200–204. URL: https://moluch.ru/conf/tech/archive/343/17430/
7. Митчелл Р. «Web-скрапинг на Python. Сбор данных с помощью BeautifulSoup и Scrapy». М.: ДМК Пресс, 2018. С. 47–48.

Журнал «Современная наука: актуальные проблемы теории и практики»