

привлечь, сколько удержать внимание пользователя. Пользователи будут игнорировать дизайн, который игнорирует их.

Список использованных источников:

1. Chris Bank, Jerry Cao, Waleed Zuberi Web UI design best practices, 2014
2. Adit Gupta // Applying Mathematics To Web Design Журнал Smashing Magazine [Электронный ресурс]. Режим доступа: <http://4design.xyz/nombre-d-or-suite-de-fibonacci-et-autres-grilles-de-mise-en-page-pour-le-design-web> - Дата доступа: 25.01.2017.
3. Bruno Bichet // Nombre d'or, suite de Fibonacci et autres grilles de mise en page pour le design web Портал 4design [Электронный ресурс]. Режим доступа: <https://www.smashingmagazine.com/2010/02/applying-mathematics-to-web-design/> - Дата доступа: 15.01.2017.

МОДУЛЬ ПОЛУЧЕНИЯ ДАННЫХ ИЗ ВНЕШНИХ ОТКРЫТЫХ ИСТОЧНИКОВ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Пресняцкий В.Ю., Рожков Д.Н., Дорошкевич П.Е., Свито А.И.

Стержанов М.В. – канд.техн.наук, доцент

Web-crawlers (also known as robots or scrapers) enable the process by following the hyperlinks in web pages to automatically download a fractional snapshot of the web site. This paper describes developed web crawler named MMScraper aimed to process informational web resources for further getting statistical properties and performing data analytics.

В настоящее время в связи с бурным развитием сети Интернет наблюдается обилие электронной неструктурированной информации, представленной текстами на естественных языках. Всё более востребованной становится задача автоматической обработки таких текстов с целью извлечения структурированных данных, которые затем используются при решении различного рода проблем: извлечения фактических данных, поиска информации и т.п. Нами решается задача обработки контента информационно-новостных ресурсов с целью анализа лексико-терминологической информации.

Для сбора требуемых данных требуются специализированные инструменты - поисковые роботы, также называемые «веб-пауками» (web-spider), краулерами (webcrawler) или скребками (webscraper). Поисковый робот — это программный комплекс, который осуществляет навигацию по веб-ресурсам и сбор информации для базы данных приложения-агента [1, 2].

Нами планируется значительная работа по обследованию ряда информационных сайтов, чтобы собрать выборку данных требуемого размера. Анализ имеющихся в свободном доступе решений показал, что открытые реализации зарубежных веб-краулеров слабо приспособлены к решаемой нами задаче, так как требуют весьма трудоемкой настройки, а после нее показывают низкую производительность и существенно нагружают информационный источник. В связи с этим было принято решение разработать собственное решение.

Краулер, названный MMScraper, был разработан нами в соответствии со следующими требованиями:

- * Получать список доменных имен сайтов, предназначенных для сканирования, в качестве исходных данных. Предполагается, что имеется некоторое множество заранее определенных для исследования сайтов.
- * Делать обход каждого сайта, начиная с главной (индексной) страницы, перемещаясь по внутренним гиперссылкам в заданном порядке обхода («сначала вширь»).
- * Полученные результаты сохранять в базу данных. Интерес представляют следующие атрибуты: адрес страницы, автор публикации, дата публикации, содержимое публикации.
- * Позволять получать данные с сайта через программный интерфейс API.
- * Иметь расширяемую архитектуру, чтобы впоследствии развивать функциональность.
- * Добавление нового сайта должно быть простым и не требовать привлечения квалифицированного программиста.

Работа разработанного краулера описывается следующим образом: сайт сканируется, начиная с главной страницы, после этого робот обрабатывает ссылки, размещённые на ней, переходя по ним, на другие страницы сайта. Каждая страница проходит анализ на наличие нужной информации, которая в случае обнаружения копируется в соответствующее хранилище. Процесс продолжается тех пор, пока не будет обработано требуемое количество страниц или пока не будет достигнута некая цель. Модуль получения данных разработан на языке программирования Ruby и состоит из трех основных частей: блока сканирования и обработки данных, блока управления краулером (команды вводятся через консоль) и база данных. Информация, которую собирает робот, состоит из сылочной структуры обрабатываемого ресурса и веб-страниц. В качестве основной базы данных была выбрана бесплатная удобная в использовании СУБД MySQL. Для упрощения взаимодействия с БД нами используется библиотека Sequel, которая позволяет представлять данные в виде объектов.

В работе приводится описание главных требований, общей архитектуры и конфигурации краулера MMScraper, который предназначен для решения довольно узкой, но важной задачи, а именно – сбора различного рода информации о новостных и информационно-аналитических публикациях.

В практическом плане разработанный MMScraпер позволит составить экспериментальную базу для исследования задач интеллектуальной обработки текста.

Нам видится важным продолжить исследования результатов краулинга и реализации дополнительных возможностей MMScraпер, улучшающих результаты работы на очень крупных сайтах всемирной паутины. Осуществление таких возможностей предусмотрено расширяемой архитектурой краулера.

Список использованных источников:

1. A.H.F. Laender, B. A. Ribeiro-Neto, Juliana S.Teixeria. A brief survey of web data extraction tools // ACM SIGMOD Record 31(2), pp 84-93. 2002
2. Baeza-Yates R., Castillo C. Crawling the Infinite Web: Five Levels are Enough // Lecture Notes in Computer Science. Algorithms and Models for the Web-Graph, Third International Workshop. 2004. Vol. 3243. P. 156–167.

УНИКАЛЬНАЯ ИДЕНТИФИКАЦИЯ ЦИФРОВЫХ СИСТЕМ НА ОСНОВЕ ДОЗУ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Пучков А. В.

Иванюк А. А. – д-р. техн. наук, доцент

В современных встраиваемых системах практически обязательным компонентом является оперативное запоминающее устройство (ОЗУ), часто динамическое (ДОЗУ). Помимо своего основного назначения ДОЗУ может быть использовано в качестве криптографического примитива, обеспечивающего уникальную неклонировуемую идентификацию цифровой системы.

Криптографические примитивы, в основе которых лежит структурная сложность физических, в частном случае электронных, систем, являются предметом изучения физической криптографии, одним из ключевых понятий которой является физически неклонировуемая функция (ФНФ). В контексте цифровых систем ФНФ в общем случае можно рассматривать как устройство, задача которого состоит в получении значений ответов на подаваемые входные воздействия – запросы, при этом пары запрос-ответ являются уникальными, непредсказуемыми и неклонировуемыми на других экземплярах интегральных схем, выпущенных в рамках конкретного технологического процесса [1]. Функционирование таких устройств основывается на том факте, что в процессе производства цифровых систем принципиально невозможно управлять значениями отдельных их параметров, которые принимают уникальные для конкретного экземпляра цифровой системы значения. Основной задачей ФНФ является извлечение таких параметров, например, задержек распространения сигналов или различий результирующих частот идентичных тактовых генераторов. Уникальность пар запрос-ответ ФНФ позволяет использовать их как криптографический примитив в методах неклонировуемой идентификации и аутентификации цифровых систем.

Статические и динамические оперативные запоминающие устройства характеризуются тем, что при включении питания часть запоминающих ячеек находится в состоянии 0, в то время как другая часть – в состоянии 1, причем распределение таких ячеек в общем случае случайно и статистически уникально, что позволяет рассматривать ОЗУ как ФНФ, где запросом является включение питания, а ответом – содержимое массива запоминающих элементов после включения питания [2]. Практическое применение такой ФНФ вызывает затруднение в той связи, что ее использование может быть произведено лишь всякий раз при включении питания ОЗУ, что в подавляющем большинстве случаев означает включение цифровой системы. В случае ДОЗУ альтернативным подходом может являться отключение регенерации части запоминающих элементов, что может быть неоднократно выполнено в процессе функционирования цифровой системы [2]. Стоит отметить, что ДОЗУ получает все более широкое распространение, в том числе во встраиваемых системах из-за своей сравнительно низкой стоимости.

Для экспериментального исследования данного метода использовались интегральные схемы ДОЗУ Micron M45W8MW16, которыми были оснащены имеющиеся в наличии 10 плат быстрого прототипирования на основе FPGA Digilent Nexys 4. Данные интегральные схемы ДОЗУ имеют объем 128 Мбит и обладают интерфейсом, схожим с ОЗУ, т.е. обеспечение регенерации запоминающих элементов достигается использованием дополнительной логики, которой можно удобно управлять при помощи доступных пользователю регистров.

Был разработан аппаратно-программный комплекс, позволяющий осуществлять управление данными ОЗУ с рабочей станции, устанавливая различные режимы работы, а также производить чтение региона ДОЗУ для последующего анализа. Задачей экспериментов было оценить различные методы извлечения уникальности интегральных схем ДОЗУ.

В ходе первого из экспериментов верхняя половина адресного пространства ОЗУ была заполнена константными значениями '1', затем была отключена регенерация запоминающих элементов этого региона. Далее производилось многократное чтение содержимое указанной области с целью последующего анализа изменений, которые происходят в ней с течением времени. Выбор значения '1' обусловлен физическими процессами, проходящими в запоминающей ячейке, а именно '1' соответствует присутствию заряда, который в