ИПМ им.М.В.Келдыша РАН

Абрау-2019 • **Труды конференции**

Е.Л. Китаев, Р.Ю. Скорнякова

StructScraper – инструмент для динамического включения в контент веб страницы семантических данных внешних веб ресурсов

Рекомендуемая форма библиографической ссылки

Китаев Е.Л., Скорнякова Р.Ю. StructScraper – инструмент для динамического включения в контент веб страницы семантических данных внешних веб ресурсов // Научный сервис в сети Интернет: труды XXI Всероссийской научной конференции (23-28 сентября 2019 г., г. Новороссийск). — М.: ИПМ им. М.В.Келдыша, 2019. — С. 424-431. — URL: **http://keldysh.ru/abrau/2019/theses/34.pdf** doi:**10.20948/abrau-2019-34**

Размещена также презентация к докладу

StructScraper – инструмент для динамического включения в контент веб-страницы семантических данных внешних веб-ресурсов

Е.Л. Китаев, Р.Ю. Скорнякова

Институт прикладной математики им. М.В. Келдыша РАН

Аннотация. Извлечение данных из Сети (веб-скрейпинг) является популярной и в то же время весьма сложной задачей из-за слабой структурированности документов, размещенных в Сети. Наличие на сайте семантической разметки (микроразметки) упрощает извлечение данных, однако доступные инструменты, применяемые с этой целью, требуют программирования для включения извлеченных данных в контент веб-страницы и к тому же обладают рядом недостатков, делающих их неудобными, если стоит задача включения данных из нескольких источников. Описываемый в данной работе инструмент StructScraper позволяет при загрузке веб-страницы добавлять в ее контент данные разных источников, извлеченные из популярных типов микроразметок: «микроданных» и JSON-LD, а также метаданные, содержащиеся в тегах <meta> html-документов и свойствах документов Word и PDF. Его использование не требует программирования – необходимо знание только HTML и CSS. Инструмент может быть полезен при создании страниц с контактными данными организаций, с ценами на один и тот же товар в разных интернет-магазинах, для добавления метаинформации к гиперссылкам и т.п.

Ключевые слова: семантическая разметка, микроданные, JSON-LD, веб-скрейпинг

StructScraper – a tool for dynamically incorporating semantic data of external web resources into web page content

E.L. Kitaev, R.Y. Skornyakova

Keldysh Institute of Applied Mathematics

Abstract. Web data extraction (web scraping) is a popular and, at the same time, very difficult task due to the poor structure of documents posted on the Web. The presence of semantic markup simplifies web data extraction, however, the available tools used for this purpose require programming to include the extracted data in the

content of the web page and have some drawbacks that make them inconvenient if the task is to include data from several sources. The StructScraper tool described in this work allows one to add data from various sources extracted from popular types of semantic markup: “microdata” and JSON-LD, as well as metadata contained in <meta> tags of html documents and properties of Word documents and PDF files when the web page is loading. Its use does not require programming – only knowledge of HTML and CSS is required. The tool can be useful when creating pages with contact details of organizations, with prices for the same product in different online stores, for adding meta-information to hyperlinks, etc.

Keywords: semantic markup, microdata, JSON-LD, web scraping

В Сети размещено большое количество данных и достаточно актуальна задача их получения в динамике непосредственно из первоисточников, извлекая нужный контент «на лету» – это позволяет избежать избыточности и не требует организации промежуточного хранения. Такое динамическое извлечение данных особенно актуально, если эти данные непостоянны по своей природе: контакты отдельных лиц и организаций, цены на товары, рейтинги заведений массового обслуживания, нормативные документы и пр. Нередки ситуации, когда в списке организаций, размещенном в Сети, адреса устарели в связи, например, с переездом, хотя на сайте самой организации информация актуальна, либо, когда на товар в магазине назначена скидка, а в агрегаторе цен это еще не отражено. Для разного рода нормативных документов, а также научных публикаций важна дата их размещения в Сети и дата последнего обновления [1, 2]. Задача автоматического извлечения информации из веб-ресурсов «на лету» или динамический веб-скрейпинг (real-time web scraping) может использоваться для отслеживания цен конкурентов, для получения биржевых сводок, для анализа состояния рынка труда, для получения сведений о погоде и пр.

Для извлечения информации из Сети (веб-скрейпинга) предлагаются различные инструменты, среди которых есть позволяющие организовать скрейпинг «на лету» и предоставить данные через REST API (например, ScrapyRT [8], Import.IO [10]), что дает возможность визуализировать извлеченные данные на веб-странице, используя JavaScript API XMLHttpRequest или fetch. Однако поскольку в общем случае задача извлечения информации из Сети весьма сложна – данные слабо структурированы и предназначены в первую очередь для прочтения человеком – инструменты веб-скрейпинга общего назначения либо предполагают кодирование пользователем алгоритма извлечения данных (как, например, Scrapy), либо предлагают визуальные средства для конструирования такого алгоритма, чтобы пользователь мог на экране указать, какие именно данные необходимо извлечь (как, например, Import.IO). Второй способ не предполагает умения программировать, но также является трудозатратным. Сегодня набирает популярность семантическая разметка, используя которую поисковые системы более адекватно отображают содержимое сайтов – среди 10 млн наиболее

популярных сайтов тот или иной вид разметки использует уже половина, а разметку JSON-LD более четверти сайтов [3]. При наличии семантической разметки извлечение данных из веб-ресурсов без программирования или конструирования алгоритма возможно, и такую возможность предоставляют, например, веб-служба Apache Any23 [9] (Anything To Triples) или API Валидатора микроразметки [11] от Яндекса. Однако эти инструменты не предназначались для обращения к ним из веб-станиц с целью извлечения «на лету» данных из нескольких источников и обладают рядом недостатков при использовании с этой целью: для получения данных надо делать отдельный запрос для каждого из источников, из-за чего увеличивается общее время получения данных; в запросе нельзя указать тип данных – результат включает все, имеющиеся в разметке – и т.п. Стоит отметить также, что какой бы из имеющихся инструментов не использовался, для включения в контент веб-страницы извлеченных данных ее создатель должен писать программный код для клиентской части веб-приложения, что требует определенных навыков.

Для создания веб-страницы с актуальными данными, собранными из различных внешних источников, предназначен инструмент StructScraper. Он позволяет включать в контент веб-станицы семантические данные веб-ресурсов, содержащиеся в микроразметках «микроданными» [5] и JSON-LD, а также метаданные (для html – извлеченные из тегов <meta>, для документов формата pdf, doc и docx – из встроенных и пользовательских свойств). Его использование не предполагает кодирования ни на серверной, ни на клиентской стороне. При работе с StructScraper автору страницы достаточно разметить HTML-страницу и подключить стартовые скрипты, вставив в страницу фрагмент уже готового кода, а вся остальная работа будет выполнена автоматически в процессе загрузки страницы. REST API, входящий в состав StructScraper, допускает CORS (Cross Origin Resource Sharing, Совместное использование ресурсов между разными источниками) [12], т. е. разрешает производить кросс-доменные запросы, что избавляет от необходимости устанавливать его на том же сайте, где размещена веб-страница пользователя.

Этот инструмент может быть полезен блогерам, авторам страниц с кулинарными рецептами, научным работникам для создания персональных страниц и списков публикаций, его можно использовать для сравнения цен на товары, рейтингов сайтов и т.п.

Код инструмента размещен на сайте веб-сервиса GitHub: https://github.com/RimmaSkorn/struct-scraper.

1. Реализация StructScraper

StructScraper включает серверную и клиентскую части. Серверная часть представляет собой REST API для извлечения данных – ее можно использовать как вместе с клиентской частью, так и самостоятельно. Клиентская включает jQuery-плагины, вызов которых при загрузке веб-страницы выполняет работу по добавлению данных в контент.

REST API StructScraper реализован на языке C#, имеющем встроенную поддержку асинхронного программирования, с использованием технологии Microsoft ASP.NET Web API. Веб-API имеет три контроллера: для извлечения отдельных метаданных, для извлечения микроданных определенного типа и для извлечения данных формата JSON-LD определенного типа. Каждый из контроллеров имеет по два POST метода: для одного URL и для нескольких URL. Параметры запроса передаются в формате JSON. Они содержат адреса веб-ресурсов, из которых необходимо извлечь данные, и сведения о том, какие именно данные должны быть извлечены. Для метаданных, извлекаемых из тегов и свойств документов, передаются названия, для микроданных и разметки JSON-LD передается тип из словаря Schema.org [13].

Обработка нескольких URL на сервере происходит асинхронно (рис.1), время ответа равно максимальному времени ответа от одного ресурса. Поэтому время ответа на клиентский запрос не больше, чем если бы запросы с клиентской стороны для каждого URL производились отдельно, а за счет сокращения времени на установку отдельных соединений к REST API оно становится меньше.

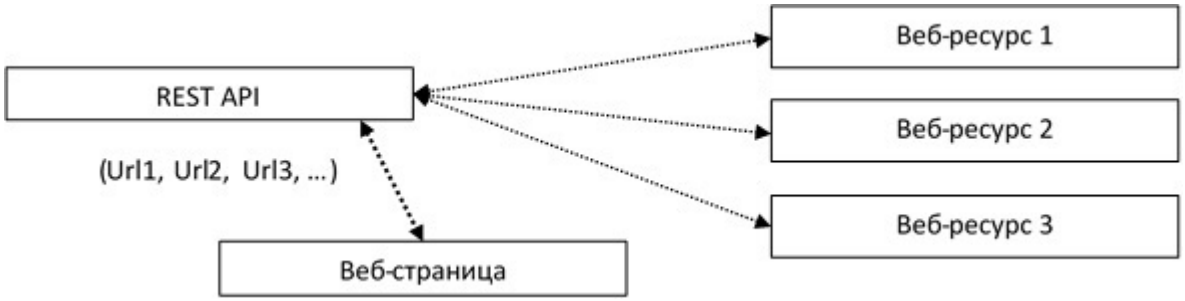

Рис. 1. Запрос к нескольким веб-ресурсам через прокси API

REST API StructScraper допускает CORS, запросы к нему могут производиться из клиентских частей веб-приложений любых доменов. CORS (Cross Origin Resource Sharing) – это технология, реализованная в современных браузерах, частично снимающая ограничения правила одного источника, введенного из соображений безопасности с целью дать возможность коду из веб-страницы одного сайта свободно взаимодействовать с ресурсами этого же сайта и максимально ограничить такое взаимодействие с ресурсами других сайтов. Без такого ограничения скрипт из страницы, загруженной с какого-нибудь сайта, мог бы обратиться, например, к почтовому серверу пользователя через сессию, открытую в другом окне браузера, получить его почту или отправить от его имени письмо, что непременно бы использовали злоумышленники. В соответствии с правилом одного источника браузеры запрещают производить ajax-запросы к стороннему серверу. До появления технологии CORS такие запросы были запрещены полностью. Технология CORS, реализованная как надстройка над HTTP протоколом, позволяет их

осуществлять если сторонний сервер дает на это явное разрешение. При этом сервер также контролирует детали кросс-доменных запросов: разрешенные методы, возможности передачи авторизующих заголовков и т.п.

Работу по загрузке клиентом данных на веб-страницу осуществляет JavaScript код, оформленный в виде плагинов jQuery (github.com/RimmaSkorn/struct-scraper/tree/master/plugins). Для того чтобы плагин загрузил данные на веб-страницу, в нее должна быть добавлена специальная разметка, по которой код плагина определяет, из каких веб-ресурсов должны быть загружены данные и какие именно данные необходимо загрузить. Эта разметка использует атрибуты class.

Пример разметки для извлечения метаданных имеется в файле test-meta.html (github.com/RimmaSkorn/struct-scraper/blob/master/HTML%20Examples/test-meta.html) на GitHub, а соответствующая веб-страница с подгружаемыми метаданными на alive.keldysh.ru/Test_Pages/test-meta.html. На GitHub имеются также примеры разметок для извлечения JSON-LD и микроданных.

2. Примеры использования

Контактные данные организаций. При помощи StructScraper можно составлять актуальные списки контактных данных предприятий, извлекаемых непосредственно с их сайтов. Загрузка данных непосредственно с сайта организации избавляет составителя списка от необходимости отслеживать их изменение. На рис. 2 представлены два скриншота, сделанные в начале и в конце загрузки веб-страницы http://alive.keldysh.ru/test_Pages/test-org.html. Вначале страница содержит только гиперссылки, в конце – уже все контактные данные.

Мониторинг цен. Сейчас многие интернет-магазины добавляют микроразметку к описаниям товаров, представленных в каталогах, включающую цену на товар. Рис. 3 содержит скриншоты веб-страницы (http://alive.keldysh.ru/Test_Pages/test-product.html) с таблицей цен на одну и ту же модель телефона в разных интернет-магазинах, сделанные в начале и в конце ее загрузки. Изначально в таблице содержатся только названия интернет-магазинов. StructScraper по имеющимся в коде страницы гиперссылкам извлекает данные о товарах, заключенные в микроразметке, и добавляет их на страницу, что гарантирует актуальность цен на момент загрузки.

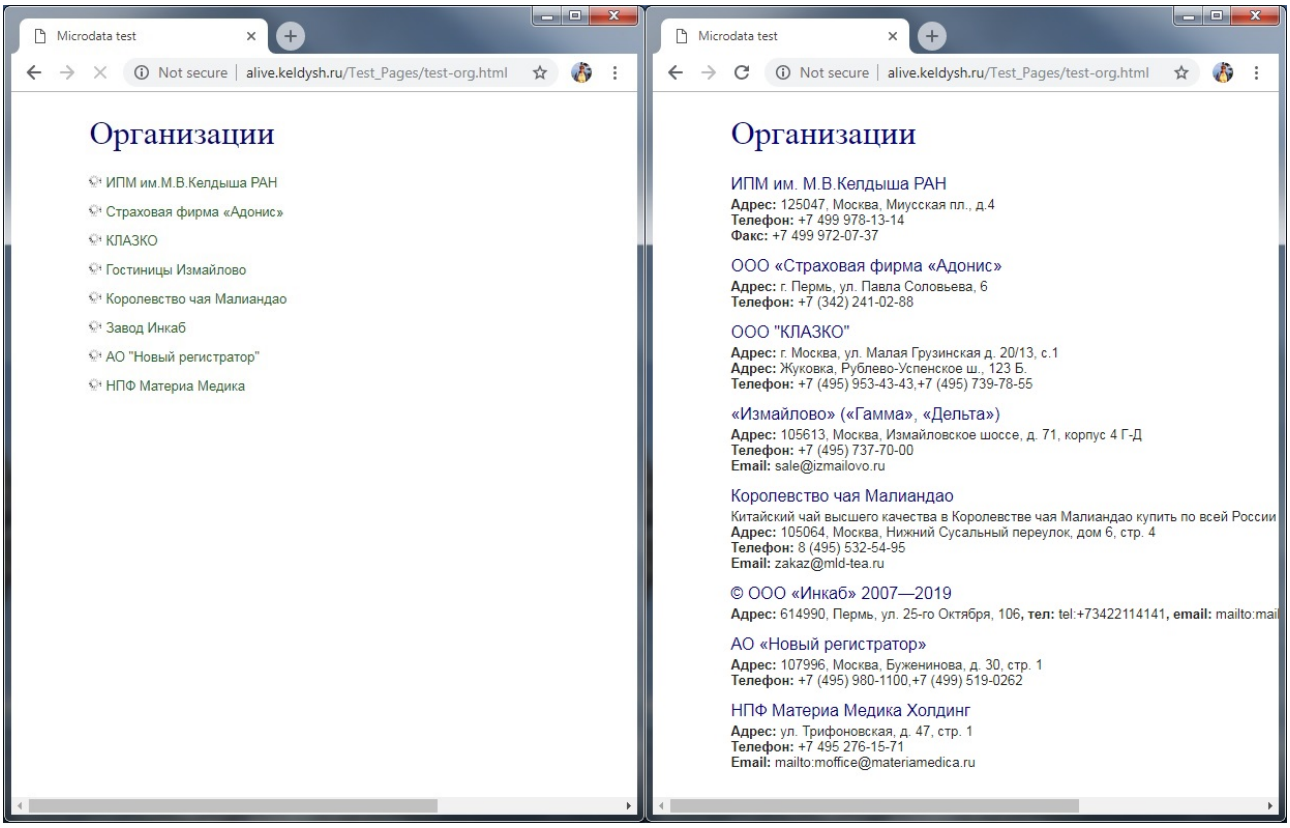

Рис. 2. Добавление "на лету" контактных данных организаций

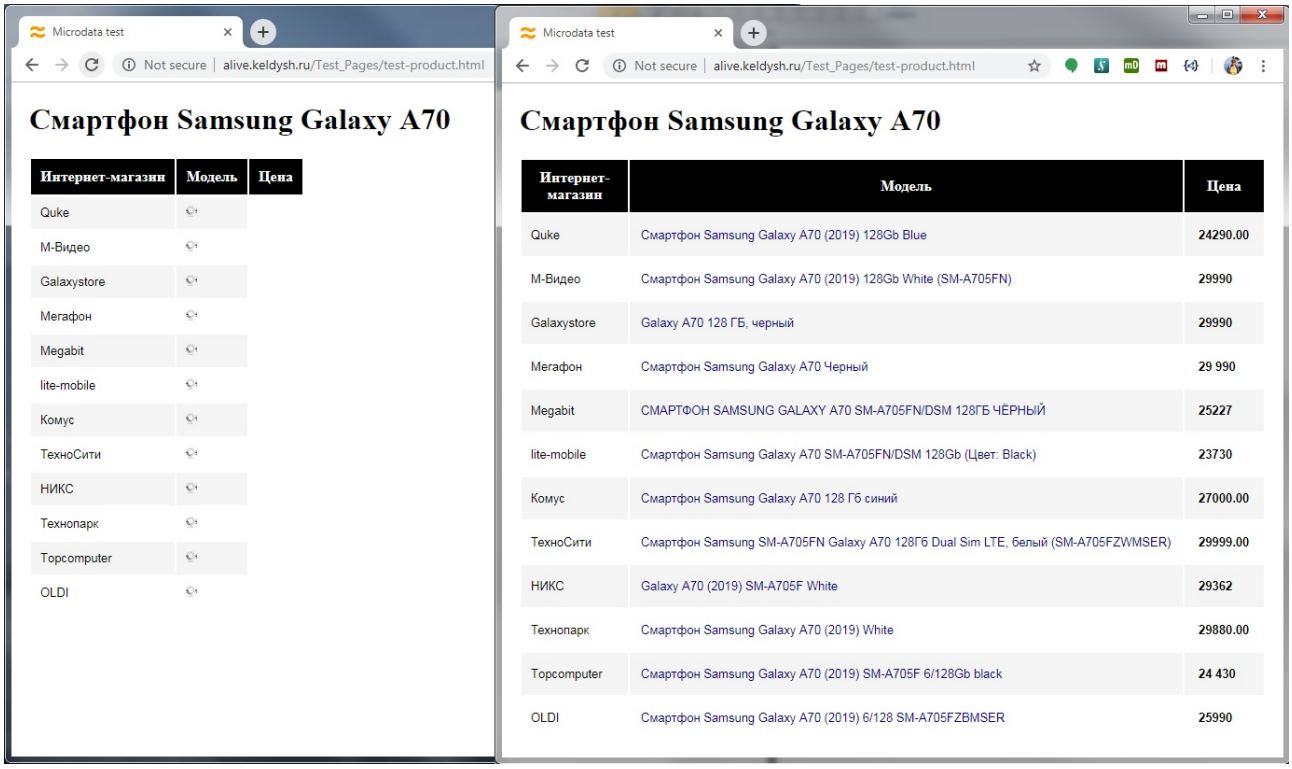

Рис.3. Цены на одну модель телефона в разных интернет магазинах, актуальные на момент загрузки страницы

* * *

Сегодня во Всемирной Паутине еще редко встречаются страницы, предъявляющие посетителю информацию из нескольких активно обновляемых сайтов, собранную именно в момент обращения к этой странице. Инструмент StructScraper позволяет создавать такие страницы без программирования при работе с сайтами, имеющими семантическую разметку. Модуль метаданных серверной части StructScraper несколько лет работает в режиме промышленной эксплуатации, в частности, на сайте ИПМ им. М.В. Келдыша для добавления в ссылки на живые публикации даты последней редакции документов. В дальнейшем планируется расширить возможности инструмента за счет добавления обработки страниц, использующих gzip-сжатие, загрузки изображений, задания списка типов из словаря Schema.org и др.

Работа выполнена при поддержке Российского фонда фундаментальных исследований, проект 19-01-00069 А.

Литература

1. Горбунов-Посадов М.М. Живая публикация // Открытые системы — 2011. — № 4. — С. 51 52. URL: https://keldysh.ru/gorbunov/live.htm .
2. Горбунов-Посадов М.М., Скорнякова Р.Ю. Обновляемая дата последней редакции в ссылке на живую публикацию // Препринты ИПМ им. М.В. Келдыша. 2017. № 82. 14 с. doi:10.20948/prepr-2017-82 URL: https://library.keldysh.ru/preprint.asp?id=2017-82 .
3. Usage Statistics of Structured Data Formats for Websites. URL: https://w3techs.com/technologies/overview/structured_data/all .
4. Китаев Е.Л., Скорнякова Р.Ю. Скрейпинг «на лету» внешних веб-ресурсов, управляемый разметкой HTML страницы // Препринты ИПМ им. М.В.Келдыша. 2019. № 20. 31 с. doi:10.20948/prepr-2019-20 URL: http://library.keldysh.ru/preprint.asp?id=2019-20 .
5. HTML Microdata. URL: https://www.w3.org/TR/microdata .
6. JSON-LD - JSON for Linking Data. URL: https://json-ld.org .
7. Scrapy. URL: https://scrapy.org/.
8. Scrapyrt (Scrapy realtime). URL: https://github.com/scrapinghub/scrapyrt .
9. Apache Any23. URL: https://any23.apache.org/ .
10. Import.io. URL: https://www.import.io/.
11. API Валидатора микроразметки — Технологии Яндекса. URL: https://tech.yandex.ru/validator/ .
12. CORS protocol. URL: https://fetch.spec.whatwg.org/#http-cors-protocol .
13. Schema.org. URL: https://schema.org/ .

References

1. Gorbunov-Posadov M.M. Zhivaia publikatsiia // Otkrytye sistemy — 2011. — № 4. — S. 51 52. URL: https://keldysh.ru/gorbunov/live.htm .
2. Gorbunov-Posadov M.M., Skorniakova R.Iu. Obnovliaemaia data poslednei redaktsii v ssylke na zhivuiu publikatsiiu // Preprinty IPM im. M.V. Keldysha. 2017. № 82. 14 s. doi:10.20948/prepr-2017-82 URL: https://library.keldysh.ru/preprint.asp?id=2017-82 .
3. Usage Statistics of Structured Data Formats for Websites. URL: https://w3techs.com/technologies/overview/structured_data/all .
4. Kitaev E.L., Skorniakova R.Iu. Skreiping «na letu» vneshnikh veb resursov, upravliaemyi razmetkoi HTML stranitsy // Preprinty IPM im. M.V.Keldysha. 2019. № 20. 31 s. doi:10.20948/prepr-2019-20 URL: http://library.keldysh.ru/preprint.asp?id=2019-20 .
5. HTML Microdata. URL: https://www.w3.org/TR/microdata .
6. JSON-LD - JSON for Linking Data. URL: https://json-ld.org .
7. Scrapy. URL: https://scrapy.org/.
8. Scrapyrt (Scrapy realtime). URL: https://github.com/scrapinghub/scrapyrt .
9. Apache Any23. URL: https://any23.apache.org/ .
10. Import.io. URL: https://www.import.io/.
11. API Validatora mikrorazmetki — Tekhnologii Iandeksa. URL: https://tech.yandex.ru/validator/ .
12. CORS protocol. URL: https://fetch.spec.whatwg.org/#http-cors-protocol .
13. Schema.org. URL: https://schema.org/ .