МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Вестник Сыктывкарского университета.
Серия 1: Математика. Механика. Информатика.
Выпуск 1 (38). 2021

УДК 004.04 **DOI: 10.34130/1992-2752_2021_1_56**

О ВОСТРЕБОВАННОСТИ ПОДГОТОВКИ В ОБЛАСТИ ПАРСИНГА ДАННЫХ ДЛЯ WEB-РАЗРАБОТЧИКОВ

А. В. Ермоленко, Н. О. Котелина, Е. Н. Старцева, М. Н. Юркина

В статье с точки зрения подготовки современного web-разработчика рассматривается процесс приобретения навыка по парсингу данных. Приводятся примерные модельные задачи, которые рекомендуется рассматривать на лабораторных занятиях. Описываются решенные в рамках научных исследований практические задачи по парсингу данных. Подробно описывается решение задачи по получению SEO характеристик сайтов.

Ключевые слова: парсер, веб-скрейпинг, web-разработка, обучение.

Введение

В настоящее время остается актуальной задача получения данных из различных, часто неструктурированных, источников (pdf, docx, html и т.д.) [1]. Для автоматизации решения этой задачи востребован синтаксический анализ, называемый парсингом[1].

Примеры задач, при которых может потребоваться синтаксический анализ:

1) обновление информации для поддержки актуальности веб-ресурса, например отслеживание изменения курса валют или прогноза погоды при помощи веб-сервисов [2];

2) сбор и быстрое копирование информации с веб-ресурсов для размещения на собственном сайте, это актуально для новостных проектов, интернет-витрин [1];

[1]От анг. parse — делать грамматический разбор.

3) сбор больших данных для дальнейшего анализа, проверки статистических гипотез, машинного обучения при помощи веб-сервисов или веб-скрейпинга, так, парсинг широко используется для сбора данных в биоинформатике, например, у открытой базы данных генетических и белковых взаимодействий модельных организмов и людей BioGRID существует REST API [3].

Представленная статья носит методический характер, призвана показать учебные и прикладные задачи, решаемые при изучении студентами понятий web-интеграции и взаимодействия информационных систем как в рамках занятий, так и при написании курсовых и выпускных квалификационных работ.

1. Способы парсинга

На взгляд авторов, учитывая востребованность в условиях цифровой экономики [4] решения прикладных задач с использованием парсинга, за время обучения студенты — будущие программисты — должны ознакомиться со следующими способами парсинга:

1) использование готовых ресурсов;

2) создание программ (как с использованием, так и без использования API ресурсов).

Существует большое количество сайтов, посвященных обзору программ для парсинга, например [5]. Для знакомства с основами парсинга необходимо включать в задания дисциплин первого курса, таких как «Информационные технологии», задачи вида «Структурируйте в табличном формате числовые характеристики 20 наиболее популярных сайтов раздела «Образование по мнению Рамблер/топ-100»», которые следует выполнять вручную или с помощью готовой программы. Данное упражнение помогает понимать задачи парсинга и вызывает потребность в автоматизации этого процесса.

Следует объяснять студентам, что многие информационные системы изначально ориентированы на интеграцию с другими системами, поэтому предлагают готовый интерфейс пользователя, наличие которого существенно упрощает процесс кодинга, в частности, API предлагают такие ресурсы, как Википедия, Twitter, ВКонтакте.

Однако API может не существовать или не быть полезным по нескольким причинам [1]:

- необходимо собрать относительно небольшие ограниченные наборы данных на большом количестве веб-сайтов без единого API;

- требуемые данные довольно малы или необычны, и создатель не считал, что для этого нужен API;
- у источника нет инфраструктуры или технических возможностей для создания API.

При этом даже если API существует, объем запросов и ограничения скорости, типы данных или формат данных, которые он предоставляет, могут быть недостаточными для ваших целей.

Указанные причины приводят к тому, что при подготовке будущего программиста необходимо уделять много времени не только использованию API, но и созданию парсеров, не использующих API.

2. Учебные задания по развитию навыка парсинга

В настоящее время огромные объемы информации размещены в глобальной сети Internet, парсинг в Internet получил специальное название — веб-скрейпинг — технология получения веб-данных путем извлечения их со страниц веб-ресурсов [1; 6].Этот факт нашел отражение при создании дисциплин вида «Web-интеграция информационных систем», «Web-программирование». Рассмотрим примеры заданий этих дисциплин, связанные с парсингом.

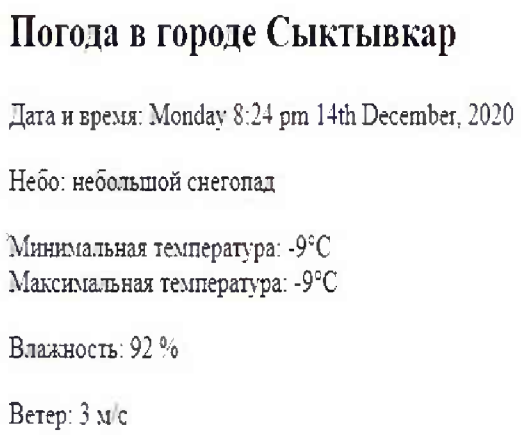

Рис. 1. Результаты парсинга JSON для REST API сервиса OpenWeather

Для отработки навыков работы с форматами XML, JSON и языком программирования PHP ставилась задача по получению информации из данных, полученных при помощи веб-сервисов. Так, при помощи REST API, бесплатно предоставляемого веб-сервисом OpenWeather [2], была получена погода на текущую дату в формате JSON. На рис. 1

приведены результаты парсинга JSON при помощи cURL PHP для Сыктывкара[2].

Следующая модельная задача основана на использовании SOAP-протокола, который возвращает данные в формате XML. На сайте Центробанка [7] есть веб-сервис на основе протокола SOAP, который возвращает xml-документ с курсами валюты. На рис. 2 приведен фрагмент результатов парсинга на PHP.

Текущий курс на 2020-12-26

Название	Курс	Номинал
Австралийский доллар	56.0502	1
Азербайджанский манат	43.0797	1
Фунт стерлингов Соединенного королевства	99.9265	1
Армянский драм	14.1043	100
Белорусский рубль	28.5352	1
Болгарский лев	45.9398	1
Бразильский реал	14.1259	1
Венгерский форинт	24.8197	100
Гонконгский доллар	95.04	10
Датская крона	12.0832	1
Доллар США	73.6921	1
Евро	89.8749	1
Индийская рупия	10.0182	10
Казахстанский тенге	17.5629	100
Канадский доллар	57.2633	1
Киргизский сом	88.0746	100
Китайский юань	11.2973	1

Рис. 2. Результаты парсинга XML для SOAP сервиса Центробанка

Код получения данных в XML формате через протокол SOAP приведен в листинге 1.

```
1 $wsdl = 'http://www.cbr.ru/DailyInfoWebServ/DailyInfo.
```

[2]Необходимые сведения для получения этих данных приведены по адресу https://active-vision.ru/blog/api-openweathermap/.

```
    asmx?WSDL';
try {
$cbr = new SoapClient($wsdl,array('soap_version'=>
SOAP_1_2, 'exceptions'=>true, 'trace' => 1));
$date = $cbr->GetLatestDateTime();
$result = $cbr->GetCursOnDateXML(array('On_date'=>$date
    ->GetLatestDateTimeResult));
}
catch (Exception $e) {
echo 'Error: '.$e->getMessage();
}
```

Листинг 1. Получение информации о курсах валют

Студенты обязательно должны иметь представление о таком популярном формате данных, как CSV. Данные в этом формате часто используются в задачах машинного обучения, они легко преобразуются в структуру DataFrame из библиотеки Pandas для последующего анализа при помощи языка Python [8]. На занятиях студенты решают задачи парсинга различных наборов данных в формате csv с последующей визуализацией или экспортом в базы данных для дальнейшего использования. Так, в качестве актуального набора данных была взята статистика по заболеваемости COVID-19 [9] с последующей визуализацией при помощи библиотеки AnyChart [10]. Результат визуализации приведен на рис. 3.

Рис. 3. Визуализация данных, полученных при помощи парсинга CSV

Веб-скрейпинг удобно выполнять при помощи программного интерфейса DOM с использованием языков программирования JavaScript и PHP. DOM позволяет представить HTML-документ в виде дерева узлов, получить доступ к его содержимому и изменять его. На листинге 2 приведен пример перебора гиперссылок на языке PHP.

```
1 $html = file_get_html($url);
2
3 foreach($html->find('a') as $element)
4         echo $element->href . '<br>';
```

Листинг 2. Перебор гиперссылок средствами DOM

Использование DOM позволяет решить задачи по выводу постов из социальной сети Twitter, чтения RSS-лент и др.

3. Исследовательские проекты

Кроме учебных заданий в рамках изучения конкретных дисциплин обучающимися решается много практических задач, в основе которых лежит парсинг данных. В разное время в рамках курсовых и выпускных квалификационных работ студенты выполняли следующие задачи:

1. Парсинг расписания вуза для представления в другом формате [11].

2. Парсинг информации об обнаруженных уязвимостях различного программного обеспечения [12].

3. Парсинг по открытым источникам для системы автоматизации поиска информации о людях в сети Интернет [13].

4. Разработка программного обеспечения для проверки ссылок на электронные издания в рабочих программах дисциплин [14].

5. Исследование зависимости между продолжительностью жизни и числом генных взаимодействий на примере модельных организмов [3].

6. Реконструкция графов взаимодействия генов на основании доступных баз данных [15].

7. Поиск предложений заданной структуры в аннотациях публикаций PubMed с целью выявления генов, влияющих на продолжительность жизни модельных организмов.

8. Анализ художественных текстов при помощи методов теории графов.

9. Анализ СМС по ключевым словам.

10. Анализ посещаемости сайтов.

Остановимся более подробно на последней задаче, так как она лежит в сфере текущих интересов авторов. В настоящее время реализуется проект, связанный с анализом зависимости посещаемости сайта от его SEO-параметров. Для этого авторами была написана программа на языке Python для консолидации SEO-параметров тематических

сайтов в структуру DataFrame из библиотеки Pandas [16]. В DataFrame данные организованы в виде таблицы, при этом строки соответствуют описаниям отдельных объектов (в нашем случае сайтов), а столбцы — признакам (SEO-метрикам).

Веб-скрейпинг страниц с результатами SEO-анализа исследуемых сайтов был проведен при помощи популярной библиотеки Beautiful Soup. Beautiful Soup — это библиотека Python для извлечения данных из файлов HTML и XML [17], которая позволяет получить по исходной разметке дерево синтаксического разбора в объекте класса BeautifulSoup.

Разделы, в которых сгруппированы сходные метрики сайта, представляют собой html теги , поэтому в программе используем метод find_all('div') для поиска нужных разделов. На листинге 3 приведен фрагмент кода получения дерева синтаксического разбора.

```
1 import requests
2 from bs4 import BeautifulSoup
3...
4 res = []
5 response = requests.get(url)
6 soup = BeautifulSoup(response.text, 'lxml')
7 res.append(soup.find_all('div'))
```

Листинг 3. Получение дерева синтаксического разбора

Полученные элементы необходимо изучить и извлечь необходимые для анализа характеристики. Для этого мы просматриваем атрибуты полученных html-элементов и ищем среди них интересующий нас идентификатор id. На листинге 4 представлен фрагмент кода для блока «Поисковые системы».

```
1 a = x.attrs
2 if 'id' in a and a['id'].count("poiskovye_sistemy")>0:
3...
```

Листинг 4. Поиск блока «Поисковые системы»

После успешного обнаружения нужного блока параметров необходимо организовать поиск по всем вложенным в него элементам (с помощью того же метода find_all()) и после очистки и трансформации

найденных данных записывать их в какую-нибудь структуру, например в словарь Python. Внутреннее содержимое тегов мы получаем при помощи свойства text. По полученному словарю формируем набор данных pandas.DataFrame, пригодный для дальнейшего анализа.

Полученные данные можно использовать для дальнейших исследований. Например, на рис. 4 приведена тепловая карта [18] для матрицы корреляций SEO-метрик, полученная при помощи библиотеки Seaborn. Метрики в нашем случае включают: Яндекс ИКС, количество посещений страниц пользователями за разные временные периоды, рейтинг Alexa (позиция в мировом рейтинге), количество уникальных внешних ссылок, уровень качества домена, скорость загрузки страниц, количество слов на сайте, количество эффективных показов и другие.

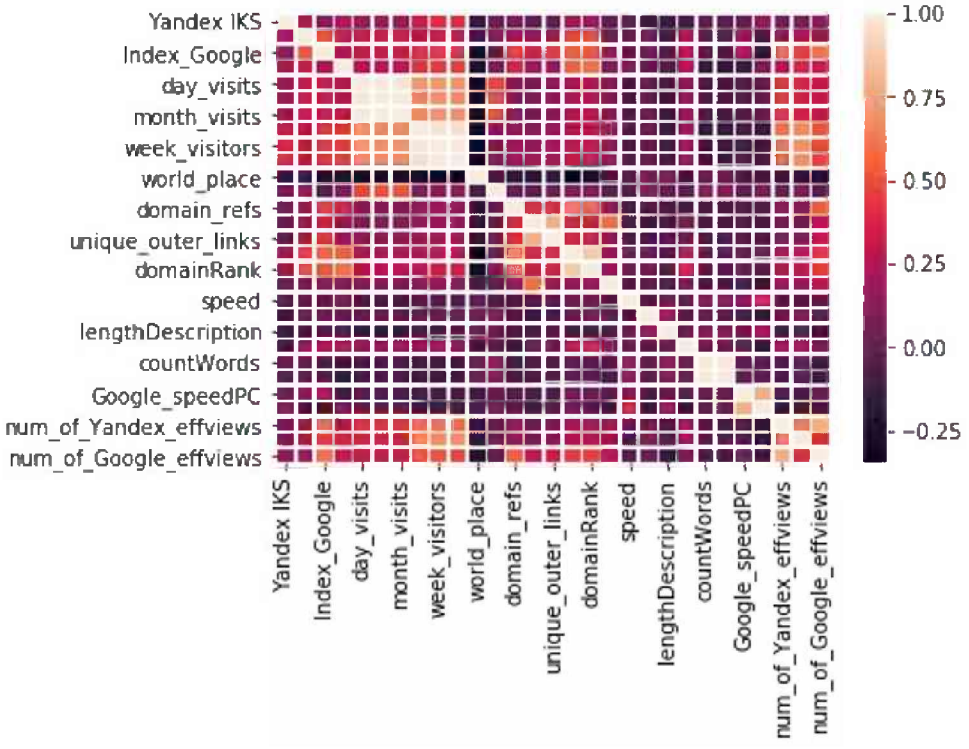

Рис. 4. Тепловая карта для матрицы корреляций

Заключение

Важным фактором формирования профессиональной грамотности начинающих программистов и повышения уровня вовлеченности обучающихся в учебный процесс является усиление прикладной составляю-

щей и уделение большего внимания межпредметным связям при изучении дисциплин. Включение в учебный процесс прикладных задач, связанных с парсингом, этому значительно способствует, а также является существенным резервом для успешного структурирования изучаемого материала.

Список литературы

1. **Mitchell R.** Web Scraping with Python. Sebastopol,: O'Reilly Media, Inc., 2018. 306 p.

2. Сервис OpenWeather. URL: https://openweathermap.org/api (дата обращения: 11.01.2021).

3. **Котелина Н. О., Арсеньев В. В., Соловьёв И. А.** Поиск корреляций между инкрементом и декрементом продолжительности жизни и число взаимодействий гена в клетке конкретного организма // *Математическое моделирование и информационные технологии: IV Всероссийская научная конференция с международным участием (12–14 ноября 2020 г., г. Сыктывкар) : сборник материалов [Электронный ресурс] 1 опт. компакт-диск (CD-ROM) / отв. ред. А.В. Ермоленко. Сыктывкар: Изд-во СГУ им. Питирима Сорокина, 2020. С. 42.*

4. **Babenko V., Golchevskiy Yu., Yermolenko A.** The strategy of development of educational programs connected with IT in the regional educational institutions // *DEFIN '20: Proceedings of the III International Scientific and Practical Conference, 2020. N 38. Pp. 1–4.*

5. 30+ парсеров для сбора данных с любого сайта. URL: https://habr.com/ru/company/click/blog/494020/ (дата обращения: 11.01.2021).

6. **Gábor L. H.** Website Scraping with Python. Library of Congress Control Number: 2018957273. 235 p.

7. Сайт Центрального банка Российской Федерации (Банка России). URL: https://cbr.ru/ (дата обращения: 11.01.2021).

8. **Груздев А., Хейдт М.** Изучаем pandas. Высокопроизводительная обработка и анализ данных в Python. М.: ДМК Пресс, 2019. 684 с.

9. COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University URL: https://github.com/CSSEGISandData/COVID-19 (дата обращения: 11.01.2021).

10. AnyChart. URL: https://www.anychart.com/ru/ (дата обращения: 11.01.2021).

11. **Гольчевский Ю. В., Виноградов И. М.** Опыт разработки Интернет-сервиса расписания учебных занятий // *Информатизация образования и науки. 2016. № 1. С. 16–25.*

12. **Гольчевский Ю. В., Северин П. А.** Анализ динамики обнаружения уязвимостей популярных систем управления контентом // *Вопросы защиты информации. 2013. № 4 (102). С. 58–66.*

13. **Гольчевский Ю. В., Кузнецов Д. И.** Автоматизация механизмов поиска информации на основе открытых источников в сети Интернет // *Информация и безопасность. 2017. Т. 20. № 3 (4). С. 414–417.*

14. **Ушаков Д. А.** Разработка программного обеспечения для проверки ссылок на электронные издания в рабочих программах дисциплин // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика. 2020. Вып. 2 (35). С. 49–58.*

15. **Котелина Н. О., Матвийчук Б. Р., Соловьев И. А.** Реконструкция графов взаимодействия генов и их приоритизация на основании доступных баз данных // *Математическое моделирование и информационные технологии: IV Всероссийская научная конференция с международным участием (12–14 ноября 2020 г., г. Сыктывкар) : сборник материалов [Электронный ресурс] 1 опт. компакт-диск (CD-ROM) / отв. ред. А.В. Ермоленко. Сыктывкар: Изд-во СГУ им. Питирима Сорокина, 2020. С. 44.*

16. Документация Pandas. pandas documentation. Date: Feb 09, 2021 Version: 1.2.2. URL: https://pandas.pydata.org/pandas-docs/stable/index.html (дата обращения: 11.01.2021).

17. Документация Beautiful Soup. Beautiful Soup 4.9.0 documentation. URL: https://www.crummy.com/software/BeautifulSoup/bs4/doc/ (дата обращения: 11.01.2021).

18. Документация Seaborn 0.11.1. seaborn.pydata.org. URL: https://seaborn.pydata.org/generated/seaborn.heatmap.html (дата обращения: 11.01.2021)

Summary

Yermolenko A. V., Kotelina N. O., Startseva E. N., Yurkina M. N. On the demand for data parsing training for web developers

In the article, from the point of view of training a modern web developer, the process of acquiring the skill of parsing data is considered. Approximate model problems that should be considered in laboratory classes are given. The practical tasks of data parsing solved in the framework of scientic research are described. The solution to the problem of obtaining SEO characteristics of sites is described in detail.

Keywords: parser, web scraping, web development, training.

References

1. **Mitchell R.** *Web Scraping with Python*, Sebastopol: O'Reilly Media, Inc., 2018, 306 p.

2. Openweathermap Service. Available at: https://openweathermap.org/api (accessed: 11.01.2021).

3. **Kotelina N. O., Arsenyev V. V., Solovyev I. A.** Poisk korrelyatsiy mezhdu inkrementom i dekrementom prodolzhitel'nosti zhizni i chislo vzaimodeystviy gena v kletke konkretnogo organizma (Search for correlations between the increment and decrement of life span and the number of gene interactions in the cell of a particular organism), *Mathematical modeling and information technologies: IV all-Russian scientific conference with international participation* (12–14 November 2020, Syktyvkar): collection of materials, Executive editor A. V.

Yermolenko, Syktyvkar: Publishing house of SSU Pitirima Sorokina, 2020, pp. 42.

4. **Babenko V., Golchevskiy Yu., Yermolenko A.** The strategy of development of educational programs connected with IT in the regional educational institutions, *DEFIN '20: Proceedings of the III International Scientific and Practical Conference*, 2020, No. 38, pp. 1–4.

5. 30+ parserov dlya sbora dannykh s lyubogo sayta (30+ parsers to collect data from any site), Available at: https://habr.com/ru/company/click/blog/494020/ (accessed: 11.01.2021).

6. **Gábor L. H.** *Website Scraping with Python*, Library of Congress Control Number: 2018957273, 235 p.

7. Website of the Central Bank of the Russian Federation (Bank of Russia), Available at: https://cbr.ru/ (accessed: 11.01.2021).

8. **Gruzdev A., Heidt M.** *Izuchayem pandas. Vysokoproizvoditel'naya obrabotka i analiz dannykh v Python* (Exploring pandas. High-performance data processing and analysis in Python), Moscow: DMK Press, 2019, 684 p.

9. COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University, Available at: https://github.com/CSSEGISandData/COVID-19 (accessed: 11.01.2021).

10. AnyChart. Available at: https://www.anychart.com/ru/ (accessed: 11.01.2021).

11. **Golchevskiy Yu. V., Vinogradov I. M.** Opyt razrabotki Internet-servisa raspisaniya uchebnykh zanyatiy (Schedule Internet-service developing experience), *Informatization of education and science*, 2016, No. 1, pp. 16–25.

12. **Golchevskiy Yu. V., Severin P. A.** Analiz dinamiki obnaruzheniya uyazvimostey populyarnykh sistem upravleniya kontentom (Dynamics of common content management systems vulnerability detection analysis), *Information security issues*, 2013, 4 (102), pp. 58–66.

13. **Golchevskiy Yu. V., Kuznetsov D. I.** Avtomatizatsiya mekhanizmov poiska informatsii na osnove otkrytykh istochnikov v seti Internet (Information search in open sources on the internet technique automation), *Information and security*, 2017, Vol. 20, No. 3 (4), pp. 414–417.

14. **Ushakov D. A.** Razrabotka programmnogo obespecheniya dlya proverki ssylok na elektronnyye izdaniya v rabochikh programmakh distsiplin (Development of software for checking links to electronic publications in the work programs of disciplines), *Bulletin of Syktyvkar University. Series 1: Mathematics. Mechanics. Informatics*, 2020, 2 (35), pp. 49–58.

15. **Kotelina N. O., Matviyuchuk B. R., Solovyev I. A.** Rekonstruktsiya grafov vzaimodeystviya genov i ikh prioritizatsiya na osnovanii dostupnykh baz dannykh (Reconstruction of gene interaction graphs and their prioritization based on available databases), *Mathematical modeling and information technologies: IV all-Russian scientific conference with international participation* (12–14 November 2020, Syktyvkar): collection of materials, Executive editor A. V. Yermolenko, Syktyvkar: Publishing house of SSU Pitirima Sorokina, 2020, pp. 44.

16. Dokumentatsiya Pandas (Pandas documentation), Date: Feb 09, 2021 Version: 1.2.2. Available at: https://pandas.pydata.org/pandas-docs/stable/index.html (accessed: 11.01.2021).

17. Dokumentatsiya Beautiful Soup (Beautiful Soup 4.9.0 documentation), Available at: https://www.crummy.com/software/BeautifulSoup/bs4/doc/ (accessed: 11.01.2021).

18. Dokumentatsiya Seaborn (Seaborn 0.11.1 documentation), seaborn.pydata.org. Available at: https://seaborn.pydata.org/generated/seaborn.heatmap.html (accessed: 11.01.2021).

Для цитирования: Ермоленко А. В., Котелина Н. О., Старцева Е. Н., Юркина М. Н. О востребованности подготовки в области парсинга данных для web-разработчиков // *Вестник Сыктывкарского университета. Сер. 1: Математика. Механика. Информатика. 2021. Вып. 1 (38). С. 56–69. DOI: 10.34130/1992-2752_2021_1_56.*

For citation: Yermolenko A. V., Kotelina N. O., Startseva E. N., Yurkina M. N. On the demand for data parsing training for web developers, *Bulletin of Syktyvkar University. Series 1: Mathematics. Mechanics. Informatics*, 2021, 1 (38), pp. 56–69. DOI: 10.34130/1992-2752_2021_1_56.

СГУ им. Питирима Сорокина *Поступила 28.02.2021*