МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»
Кафедра дискретной математики и информационных технологий

ИСПОЛЬЗОВАНИЕ ТЕХНОЛОГИЙ ВЕБ-СКРАПИНГА ДЛЯ АНАЛИЗА ДАННЫХ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 421 группы
направления 09.03.01 — Информатика и вычислительная техника
факультета КНиИТ
Акмаева Виктора Сергеевича

Научный руководитель
доцент, к. э. н. _____ Г. Ю. Чернышова

Заведующий кафедрой
к. ф.-м. н. _____ Л. Б. Тяпаев

Саратов 2019

ВВЕДЕНИЕ

Объем данных, которые присутствуют в глобальной сети Интернет, растет на протяжении многих лет. Большая часть этих данных представлена в гипертекстовом формате. Такой формат подразумевает неструктурированный набор данных, что затрудняет использование стандартных алгоритмов, интеллектуального анализа данных. Извлечение полезной информации может являться главной целью для различных практических задач, таких как анализ интернет-магазинов конкурентов, объединение новостей из различных интернет-ресурсов на одной странице, автоматическая классификация веб-ресурсов.

Большая часть алгоритмов обработки, использующие методы машинного обучения и статистической обработки, ориентированы на структурированные данные. Преобразование веб-контента в структурированный вид становится нетривиальной задачей. Для этого в настоящее время используются технологии веб-скрапинга как перспективное направление для автоматического сбора данных с веб-сайтов.

Цель бакалаврской работы – анализ веб-ресурсов средствами веб-скрапинга.

Поставленная цель определила следующие задачи:

— анализ методов и инструментов для разработки приложения по сбору данных с веб-ресурсов;
— разработка приложения с использованием методов веб-скрапинга;
— осуществление процесса сбора данных веб-ресурсов для их обработки;
— анализ содержимого веб-страниц на сайтах ВУЗов.
— сравнение оценок содержимого новостных лент сайтов ВУЗов.

В качестве методологических основ для бакалаврской работы по теме «Использование технологий веб-скрапинга для анализа данных» использовались работы Митчел Р., Baesens, B., Vanden Broucke, S., официальная документация по библиотекам языков Python, Java, C#.

Практическая значимость бакалаврской работы заключается в разработке приложения для анализа текстовой информации, представленной на гипертекстовых страницах отдельных ресурсов, средствами веб-скрапинга.

В первом разделе выпускной квалификационной работы представлены различные подходы к реализации технологии веб-скрапинга. Во втором разделе описываются основные этапы разработки приложения для извлече-

ния и анализа данных веб-ресурсов ВУЗов. В третьем разделе рассмотрены результаты применения технологий веб-скрапинга для новостных лент национально-исследовательских и государственных университетов России.

Бакалаврская работа состоит из введения, 3 разделов, заключения, списка использованных источников и 1 приложения. Общий объем работы – 40 страниц, из них 35 страниц – основное содержание, включая 16 рисунков и 2 таблицы, цифровой носитель в качестве приложения, список использованных источников информации – 22 наименования.

Первый раздел «Методы веб-скрапинга» посвящен анализу методов и инструментов веб-скрапинга.

Веб-скрапинг – автоматизированный сбор данных из Интернет. Данное направление подразумевает сбор данных при помощи любых средств, кроме человека, которое использует браузер или программный интерфейс приложения. Программы, осуществляющие сбор данных, чаще всего автоматически делают запрос на веб-сервер для анализа данных, после производят синтаксический анализ данных, чтобы получить необходимую информацию.

Существует большое количество практических сфер, где требуется доступ к данным неограниченного объема. В задачах прогнозирования рынка, машинного перевода, медицинской диагностики анализ данных с новостных сайтов, различных форумов, других веб-ресурсов позволят отследить последние тенденции в конкретном направлении.

Постоянная оптимизация интернет-ресурсов под мобильные устройства, растущие скорости интернета, технологические решения на уровне аппаратного и программного обеспечения, а также дизайнерские поиски сделали всемирную сеть Интернет сложным аппаратом для сбора и анализа данных. При скрапинге веб-ресурсов могут возникнуть трудности, касающиеся правовых вопросов. Это связано с тем, что собранное содержимое из разных ресурсов присваивается человеком, который производит скрапинг. Чтобы не возникали проблемы при веб-скрапинге, необходимо учитывать, что извлекаемое содержимое не должно быть защищено авторским правом, сбор данных не должен мешать работе сайта, процесс не должен нарушать условия использования сайта, разработанное приложение не должно извлекать личную информацию пользователя, контент, который подвергается сбору, должен отвечать стандартам правомерного использования.

Регулярные выражения позволяют быстро отсеивать большое количество ненужных элементов из исходного текстового контента, не изменив основной контент, такими элементами могут быть остатки HTML кода. Выполнение HTTP-запросов или разбор HTML кода, позволяет получать динамические и статические страницы путем отправки HTTP запросов к удаленным серверам. Разбор DOM структуры на основе скраперов экрана является одним из часто используемых методов веб-скрапинга. Динамический контент — один из проблемных моментов веб-скрапинга. Для его получения можно

использовать любой полноценный веб-браузер, который воспроизведет динамический контент и скрипт на стороне клиента. Если стоит задача скрапинга сотен или тысяч сайтов, при этом они имеют различную верстку и написаны на разных языках и фреймворках. В такой ситуации рациональнее всего будет использовать методы искусственного интеллекта или онтологий.

Для реализации поставленных задач был выбран метод выполнения HTTP-запросов, так как он позволяет быстро получать большой объем данных с веб-ресурсов, имеющие различную структуру.

Существует большое количество средств для написания приложения по сбору данных. Можно использовать различные языки программирования и специальные библиотеки к ним. Для написания приложения по сбору данных на Java существует библиотека Jaunt. Библиотека предоставляет быструю, сверхлегкую программу просмотра сайтов, которая не имеет графического интерфейса. Предоставляет функции веб-скрапинга, доступа к DOM и контроля над каждым HTTP-запросом / ответом. Одним из способов написания программы для сбора данных на C# является библиотека Iron WebScrapper. Это библиотека классов и фреймворков для C # и платформы программирования .NET, которая позволяет программно обходить веб-сайты и извлекать их содержимое. Для сбора данных по средствам JavaScript необходимо использовать программную платформу node.js и два модуля npm (Node Package Manager) с открытым исходным кодом для упрощения задачи.

Python - стремительно развивающийся язык. Данный язык прост в освоении, поставляется под открытой лицензией и является свободно распространяемым. Это позволяет языку быть очень популярным среди разработчиков, так как появляется огромное количество библиотек, расширений, готовых компонентов, которые можно использовать в своих проектах, что значительно экономит время на разработку.

С учетом всех преимуществ языка Python и наличия специальных библиотек для извлечения данных с веб-страниц, данный язык является наиболее удобным для разработки программы для сбора и анализа данных с веб-сайтов.

Второй раздел «Создание приложения для извлечения и анализа данных новостных ресурсов» посвящен созданию приложения для сбора и анализа данных из новостных веб-ресурсов.

Для создания приложения по извлечению данных с веб-сайтов необходимо подключение библиотек. На данный момент существует две основные библиотеки для сбора данных из сети Интернет, это Beautiful Soup и Scrapy. BeautifulSoup достаточна проста в освоении. Наличие документации с большим количеством примеров популярных задач позволяет быстро использовать данную библиотеку для извлечения нужных данных. Scrapy занимается не только извлечением контента, но и позволяет быстро решить многие проблемы при обходе веб-страниц. Библиотека имеет встроенные функции для решения наиболее распространенных проблем, таких как: перенаправления, повторные попытки выполнения определенных типов запросов, HTTP-кэширование, фильтрация дублированных запросов и так далее.

Выбор библиотеки Scrapy является целесообразным в том случае, если необходимо разработать эффективное приложение, которое может сканировать множество наборов данных за короткое время. Однако, если проект небольшой, логика не очень сложная и необходимо, чтобы работа выполнялась быстро, будет логично использовать BeautifulSoup, чтобы сохранить проект простым. Исходя из вышеизложенного, было принято решение использовать BeautifulSoup, как библиотеку, которая имеет все необходимые инструменты для успешного извлечения данных с веб-сайтов.

Прежде чем начинать писать код, необходимо изучить принцип построения новостных страниц. Для разработки правильного алгоритма обхода всего новостного ресурса было решено разобрать схему страницы новостной ленты. При каждом переходе на новую страницу программа должна обработать HTML код и оставить только текстовую часть, без тегов. Затем преобразовать получившиеся выражения в начальную форму, также как и фразы из списка ключевых слов, после этого будет осуществлен сбор статистики по ключевым словам, которые будут встречены при разборе отформатированной страницы. Каждое найденное слово или словосочетание из списка будет обработано и подсчитано.

Создание приложения на данном этапе подразумевает создание графического интерфейса для удобного представления элементов взаимодействия с приложением и необходимой текстовой информации. Для этого было решено разместить графики и таблицы со значениями содержания ключевых слов на веб-странице в интерфейсе программы. Qt Designer - это инструмент

для проектирования и построения графических пользовательских интерфейсов. Он позволяет создавать виджеты, диалоги или заполнять основные окна, используя экранные формы и простой интерфейс перетаскивания. Для взаимодействия, запуска и просмотра статистики рабочей программы обхода новостных ресурсов был спроектирован интерфейс программы. При запуске приложения наблюдается появление окна с информацией о количестве университетов, выбранных для обхода их новостных ресурсов.

Разработанный интерфейс приложения в среде разработки Qt Designer и созданные методики сбора данных с веб-сайтов позволили извлечь данные с веб-сайтов, перейти к анализу и выводу результатов.

Третий раздел «Анализ содержимого веб-ресурсов ВУЗов» посвящен анализу данных собранных из новостных веб-ресурсов.

После обхода всех страниц, относящихся к определенному ВУЗу, необходимо сохранить обработанные данные в файл для дальнейшей обработки. JSON сокращенно обозначается как JavaScript Object Notation. Он используется в качестве синтаксиса для хранения и обмена данными. Структура JSON основана на паре имя / значение, в которой представлены данные. Фигурные скобки для хранения объектов и каждого имени, за которым следуют двоеточие и пары, разделяются запятой. CSV сокращается до значения, разделенного запятыми. В файле CSV табличные данные были сохранены в виде простых текстовых данных, разделенных запятой. Формат CSV является самым компактным форматом из всех форматов файла. Формат CSV составляет примерно половину размера JSON, что увеличивает пропускную способность. Было принято решение хранить получившиеся данные в файле с расширением «.csv». Формат CSV является наиболее распространенным форматом импорта и экспорта для электронных таблиц и баз данных.

После обхода всех интересующих страниц и составления файла с результатами необходимо проанализировать данные. Для оценки полученных значений было решено применить описательную статистику - один из разделов статистической науки, в рамках которого изучаются методы описания и представления основных свойств данных. Это позволит обобщать первичные результаты, полученные при наблюдении или в эксперименте. В общем виде описательные статистики сводятся к группировке данных по их значениям, построению распределения их частот, выявлению центральных тенден-

ций распределения и к оценке разброса данных по отношению к найденной центральной тенденции. Для количества встречающихся ключевых слов на каждом новостном ресурсе и получения статистических данных были рассчитаны: среднее арифметическое, медиана, квартили. Данные описательной статистики отображаются в новом окне приложения в виде таблиц. В окне приложения строятся две таблицы для разных групп ВУЗов. Первая группа - национально-исследовательские университеты, вторая - государственные ВУЗы России.

Построение графиков является простейшей техникой описательной статистики для наглядного представления переменных. В настоящее время существует несколько библиотек для отображения графиков в Python. Чтобы выбрать подходящую библиотеку необходимо проанализировать их. Некоторые библиотеки для визуализации полученных данных работают вместе с пакетом Numpy. Он позволяет выполнять основные операции над n-массивами и матрицами: сложение вычитание, деление, умножение, транспонирование, вычисление определителя и т. д. Благодаря механизму векторизации, NumPy повышает производительность и, соответственно, ускоряет выполнение операций. Таким образом, наиболее популярными являются библиотеки Pandas, Matplotlib, Bokeh, Plotly.

Визуализация данных реализуется при помощи библиотеки Bokeh, в связи с тем, что данный инструмент позволяет эффективно и быстро создавать графики. Гистограммы содержат в себе значения, найденные по ключевым словам при обходе новостных лент ВУЗов. Сравнение происходит по двум группам, это национально-исследовательские университеты и государственные ВУЗы России. По результатам можно сделать выводы, что по всем ключевым словам, кроме упоминания в прессе, группа НИУ превосходит группу государственных ВУЗов.

ЗАКЛЮЧЕНИЕ

В связи со стремительно растущим объемом неструктурированного текстового контента возникает проблема использования стандартных алгоритмов интеллектуального анализа для обработки гипертекстовых страниц.

В выпускной квалификационной работе был выполнен анализ методов и инструментов для разработки приложения по сбору данных с веб-ресурсов. Предлагается использовать метод выполнения HTTP-запросов, так как он позволяет быстро получать большой объем данных с веб-ресурсов, имеющие различную структуру.

Разработка приложения с использованием методов веб-скрапинга была выполнена при помощи библиотеки BeautifulSoup для языка Python, которая позволяет осуществлять HTTP-запросы, проводить разбор и анализ страницы. Интерфейс программы был спроектирован в среде разработки графических интерфейсов Qt Designer. Приложение позволяет выполнить сбор данных с веб-ресурсов по описанному методу обхода веб-страниц.

Полученные данные имеют структурированный вид, пригодный для анализа и сравнения полученных сведений с новостных лент сайтов ВУЗов.

Разработанное приложение обеспечивает настройку параметров поиска в виде задания набора ключевых слов, указание временных интервалов для сбора данных, получение описательной статистики в формате CSV, построение различного вида гистограмм с возможностью описания рядов данных.

Практическое применение разработанного подхода позволяет дополнить стандартные сведения о деятельности ВУЗов, представленные на сайте, дополнительными данными, извлеченными с новостных лент. Это позволит расширить возможности анализа деятельности ВУЗов, дополнив набор показателей для оценки университетов.

ОСНОВНЫЕ ИСТОЧНИКИ ИНФОРМАЦИИ

1. Митчелл, Р. Скрапинг веб-сайтов с помощью Python. пер.: А. В. Груздев. - М.: ДМК Пресс, 2016. 280 с.
2. Baesens, B. and Vanden Broucke, S. Practical Web Scraping for Data Science, Berkley, Apress, 2018. 306 с.
3. Beautiful Soup Documentation [Электронный ресурс]. URL: https://www.crummy (дата обращения: 03.05.2019). Загл. с экрана. Яз. англ.
4. Jaunt Java Web Scraping & JSON Querying [Электронный ресурс]. URL: https://www.jaunt-api.com/ (дата обращения: 09.05.2019). Загл. с экрана. Яз. англ.
5. The C# WebScraping Library [Электронный ресурс]. URL: https://ironsoftware.c (дата обращения: 09.05.2019). Загл. с экрана. Яз. англ.