Курсова работа

по Извличане на информация, ФМИ, Софийски унивеситет на Мария Георгиева Терзиева, ф. н. 25026, спец. Изк. интелект

Тема: Извличане и обработка на информация за обувки

Съдържание:	
Мотивация, Задача на курсовата работа	2
Кратък обзор	2
Моето решение	
Програмна реализация	2
Резултати	
Заключение	
Литература	

Декларация за липса на плагиатство:

Тази курсова работа е моя работа, като всички изречения, илюстрации и програми от други хора са изрично цитирани. Тази курсова работа или нейна версия не са представяни в друг университет или друга учебна институция. Разбирам, че ако се установи плагиатство в работата ми, ще получа оценка "Слаб"

Дата: 20.02.2016г.

Подпис:

Мария Георгиева Терзиева

1. Мотивация, Задача на курсовата работа

Задачата на тази курсова работа е да обедини данните от няколко сайта за обувки, да ги индексира и да даде възможност да се търсят обувки по вид, цвят, размер, марка, модел. Мотивацията е потребителят да има по-голяма вероятност да намери обувките, които търси, и то - със заявка на едно единствено място.

2. Кратък обзор

Преди аз лично не съм извършвала работа в тази област, но има сайтове, които предлагат подобни услуги.

Идеята е да се изтеглят данните от няколко сайта за обувки (в конкретния случай cornersport, sportdepot и shopsector). Върху тези данни след това трябва да бъде реализирана търсачка, която да може да обработва заявки от вида:

```
"розови маратонки adidas 38 номер"
"adidas hoops"
"женски кецове в лилаво"
"черни"
и др. подобни
```

като се изисква резултатите да бъдат изведени от най-релевантен към найнерелевантен за конкретната заявка.

3. Моето решение

В моето решение съм използвала scrapy за извличането (crawl-ването) на данни от сайтовете. Тези данни са запазени в база от данни MongoDB. За всяка обувка се пази id, заглавие (съдържащо някои от следните: "тип на обувката", "марка", "модел", "някакви кодове"), линк към снимка, линк към оригиналния сайт, списък от цветове, в които е налична обувката, списък от размери, в които е налична обувката, цена и вид: "мъжки", "женски".

Реализиран е уеб сайт с Ruby on Rails, през който се извършва търсенето и извеждането на резултатите.

В проекта при кликане върху обувка евентуално се показват подобни или еднакви с нея обувки от други сайтове. Тази част от проекта е реализирана за курса по Подходи за обработка на естествен език и нейната имплементация няма да бъде разглеждана тук.

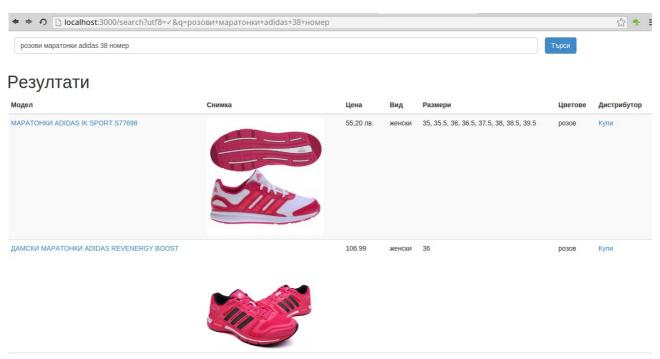
4. Програмна реализация

За crawl-ването е използван scrapy и по-точно CrawlerSpider с помощта на

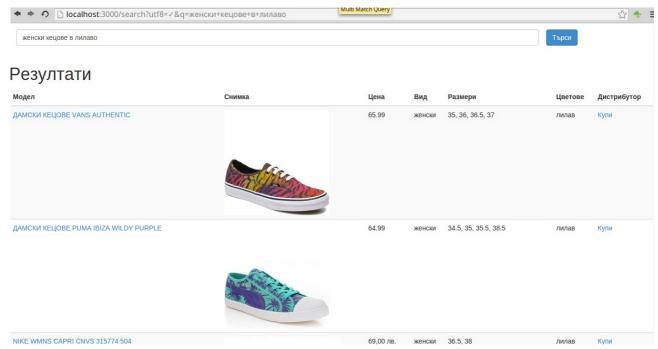
LinkExtractor и правила. Извлечените данни се записват в MongoDB чрез дефиниран pipeline във файла pipelines.py и подходящи настройки в settings.py. Файловете могат да бъдат намерени в кода на проекта, като са обособени в различни директории за различните сайтове.

За индексирането на данните и реализирането на търсачката е изполвана elasticsearch библиотеката. Използван е stemmer-ът за български език, който предоставя elasticsearch. Изполвано е също така свеждане на думите в индекса до lowercase. Търсенето е имплементирано, като е използвана "multi_match" заявка от elasticsearch от тип "best_fields" с "tie_breaker: 0.5" и полета "fields: ['title', 'gender', 'colors', 'sizes', 'price']". Този вид заявка работи по следния начин: Намира документи, които match-ват с което и да е поле от зададените полета в заявката, но използва оценката от най-доброто поле. Когато към тази заявка бъде добавен и tie_breaker, тя изчислява оценката по следния начин: оцената, от полето с най-добро съвпадение + tie_breaker * оценката за всички други съвпадащи полета. Кодът може да бъде намерен в calceus/app/models/shoe.rb.

5. Резултати



Фиг. 1.: Резултати от търсенето на "розови маратонки adidas 38 номер"



Фиг 2.: Резултати от търсенето на "женски кецове в лилаво"

6. Заключение

Какво е направено?

Извлечени са данни от три сайта за обувки – cornersport, sportdepot и shopsector. Може да се обработват заявки като тези от краткия обзор. Търсенето става по модел, марка, цвят, размер, тип на обувката.

Какво още може да се направи?

Има проблеми с множествените числа на прилагателните за цвят: сини, червени, сиви, бели, зелени. Идва от stemmer-а на elasticsearch. В бъдеще това трябва да се оправи, като може би се използва друг stemmer.

Трябва да се реализира по-добро търсене по цена вместо пълното съвпадение, както и да се добави функционалност за търсене по размер чрез задаване на диапазон.

Може да се ограничат връщаните резултати с отиването надолу по списъка да не включват не толкова релевантни обувки.

7. Литература

https://realpython.com/blog/python/web-scraping-with-scrapy-and-mongodb/ https://realpython.com/blog/python/web-scraping-and-crawling-with-scrapy-and-mongodb/

http://doc.scrapy.org/en/1.0/intro/tutorial.html

 $\underline{http://doc.scrapy.org/en/latest/topics/request-response.html\#topics-request-response-ref-request-callback-arguments}$

http://www.sitepoint.com/full-text-search-rails-elasticsearch/

https://coderwall.com/p/4rv1bg/handle-mongodb-arrays-in-rails-forms

http://blog.codescrum.com/2015/06/25/elasticsearch rails and mongoid/

http://exploringelasticsearch.com/searching data.html

https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-multi-match-query.html

http://joelabrahamsson.com/elasticsearch-101/

https://github.com/elastic/elasticsearch-rails/tree/master/elasticsearch-model