

Outline

1 Introduction

2 Data acquisition

3 Indexing

4 Future work



Introduction

Target

UK job market for general industry

Goal

Create a search system for job vacancies which has a user-friendly interface and search tools that are quick and easy to use in hope to bridge the gap between job seekers and employers



Data Acquisition

Tool

Scrapy is an open-source collaborative framework which has several useful tools to manage every step of web crawling

Websites scraped

- Reeds.co.uk : 933 jobs
- Indeed.co.uk: 904 jobs
- Technojobs.co.uk: 11,744 jobs
- Jobs 24: 4,001 jobs
- Jobs today: 3,974 jobs

Content scraped

Vacancy title, company name, job description, job location, URL, latitude and longitude, job post date, salary

Scrape the data

```
import scrapy
class JobsSpider(scrapy.Spider):
    name = "reed"
    download_delay = 1
    user_agent = 'University of Amsterdam - Information Retrieval Project (persenal_email@gmail.com)'
    httppcache_enabled = False
    start_urls = [
        'https://www.reed.co.uk/jobs',
    ]

    def parse(self, response):
        # follow links to author pages
        for href in response.css("article.job-result a::attr(href)").getall():
            if href.find("source") > 0:
                yield response.follow(href, self.parse_vacancie)
            else:
                print('Skip this page')

        # follow pagination links
        next_page = response.css("[id=nextPage]::attr(href)").get()
        if next_page is not None:
            yield response.follow(next_page, callback=self.parse)

    def parse_vacancie(self, response):
        list_disc = response.css("div.description *::text").getall()
        title_obj = response.css('div.col-xs-12 h1::text').get()
        company = response.css("div.posted span::text").get()
        location = response.css("div.location.col-xs-12.col-sm-6.col-md-6.col-lg-6 ::text").getall()

        yield {
            'title': title_obj,
            'company': company,
            'description': list_disc,
            'location': location,
            'URL': response.request.url,
        }
```


Indexing, Storage and Search

Tool

Elasticsearch is a real-time distributed, RESTful search and analytics engine that allows us to store and index our data for full text, structured search.

```
{  
  "title": "marketing executive",  
  "company": "mcf",  
  "description": "apply employer website marketing executive .....",  
  "location": "Berkshire",  
  "url": "https://www.reed.co.uk/jobs/marketing-executive/37258552"  
}
```

Text preprocessing

- Stopword removal
- Stemming
- Tokenization

Indexing

- Indices for all the features
- Subdivide indices in 5 shards

Indexing, Storage and Search

Search function

- Different weights for features
- Searching on one or more features

```
res = es.search(index="job", doc_type="job", body={"query": {"multi_match" : {  
  "query" : search,  
  "fields": ["title^3", "description", "location", "URL"]  
}}})
```

```
searchFunction("title", "data scientist")
```

```
139 documents found  
6A2EFwKBP7Y1UR11_nWA) data scientist  
5w2EFwKBP7Y1UR11-nQr) lead data scientist uk  
og2FFwKBP7Y1UR11AXaX) data engineer data scientist  
hQ2FFwKBP7Y1UR11BXeB) senior data scientist  
SQ2EFwKBP7Y1UR115nCk) data scientist ai engineer  
Vg2FFwKBP7Y1UR11AHZV) data scientist greek speaking  
mw2EFwKBP7Y1UR1142_A) health insurance data scientist london  
Ow2EFwKBP7Y1UR1183MA) bioprocessing scientist  
eg2EFwKBP7Y1UR1163HD) nlp scientist  
7w2EFwKBP7Y1UR116XBp) senior research scientist
```

Indexing, Storage and Search

Search function

- Term-based methods: BM25
- Spelling check for user typos:
 - Python package: pypellchecker
 - *Levenshtein Distance algorithm* to find permutations within an edit distance of 2

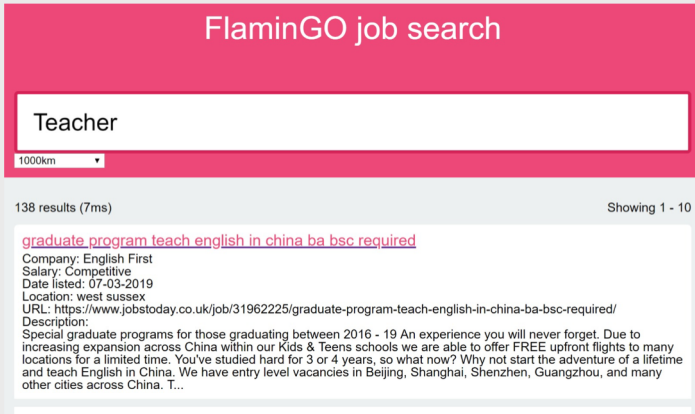
Ex: sitten -> sitting

sitten → sittin (substitution of "i" for "e")

sittin → sitting (insertion of "g" at the end)

Interface

- **Calaca project** : uses Elasticsearch as a back-end search engine and Angular for front-end web application



The screenshot displays the FlaminGO job search interface. At the top, a pink header contains the text "FlaminGO job search". Below this is a search input field containing the word "Teacher". Underneath the search field is a dropdown menu currently set to "1000km". The interface shows "138 results (7ms)" and "Showing 1 - 10". A single job listing is visible, with the title "graduate program teach english in china ba bsc required" in red. The listing details include: Company: English First, Salary: Competitive, Date listed: 07-03-2019, Location: west sussex, and URL: https://www.jobstoday.co.uk/job/31962225/graduate-program-teach-english-in-china-ba-bsc-required/. The description mentions special graduate programs for those graduating between 2016 - 19.



THANKS!
Any questions?