

Zadanie 1: Elasticsearch

Vyhľadávanie informácií

Zadanie (in english)

- Retrieve non-trivial dataset from the web.
- Process, create indexes and provide fast scalable search over the data (ElasticSearch) on selected use case
- Data - min 500MB+ (wikipedia, dbpedia...)
- Evaluate your solution and highlight some features
 - You CAN use Kibana
 - Need to have several nontrivial CUSTOM QUERIES (use advanced features)

Získavanie dát

Dáta sme získavali technikou zoškrabúvania (z angl. scraping, scrape). Použili sme programovací rámec Scrapy¹, ktorý bol vytvorený pre programovací jazyk Python za účelom extrahovania dát z webu. Získavané dáta sú články zo štyroch svetových webov:

- [The Verge](#)
- [Wired](#)
- [The Guardian](#)
- [Mashable](#)

Pomocou Scrapy sme napísali program, ktorý rekurzívne prelieza tieto štyri weby. Scrapy umožňuje jednoduché nastavenie pavúka (angl. spider, terminológia Scrapy pre program, ktorý extrahuje dáta z webu). Pavúk mal obmedzenie na navštevovanie len vyššie spomenutých domén. Naviac, aby sme sa vyhli navštíveniu stránok ako „Kontakt“, každá navštívená adresa musí spĺňať regulárny výraz:

```
pattern = 'http://www.theverge.com/[0-9]{4}/[0-9]{1,2}/[0-9]{1,2}/[0-9]{8}/.*|' \
          'https://www.wired.com/[0-9]{4}/[0-9]{2}/.*|' \
          'https://www.theguardian.com/.*/[0-9]{4}/[a-z]{2,4}/[0-9]{2}/.*|' \
          'http://mashable.com/[0-9]{4}/[0-9]{1,2}/[0-9]{2}/.*|'
regex = re.compile(pattern)
```

Zacyklenie pavúka rieši Scrapy ukladaním navštívených adries do medzi-pamäte, pričom druhý krát nenavštívi tú istú adresu. Pomocou XPath selektorov sme extrahovali dáta priamo do výsledného formátu a vyhli sme sa tak ukladaniu čistých stránok v HTML formáte. Ukážka extrahovania dát pomocou XPath selektorov:

```
url = response.xpath('//link[@rel="canonical"]/@href').extract_first()
result['title'] = response.xpath('//meta[@property="og:title"]/@content').extract_first()
result['author'] = response.xpath('//meta[@name="author"]/@content|//meta[@name="Author"]/@content').extract()
```

¹ <https://scrapy.org/>

Extrahované dáta:

- *url*, webová adresa článku.
- *title*, nadpis článku.
- *author*, autor alebo autori článku, ak ich je viac.
- *timestamp*, deň publikovania článku.
- *image_link*, odkaz na hlavný obrázok článku.
- *article*, text článku.
- *categories*, kategórie do ktorých článok patrí.

Dáta boli ukladané priamo do súboru vo formáte JSON², pričom jeden riadok predstavuje jeden záznam (článok). Takto pripravené dáta je možné použiť na hromadný (angl. bulk) import do Elasticsearch-u. Ukážka dát je v súbore *data_sample.json*. Výsledný dataset má veľkosť 85 000 článkov, na disku zaberá 350MB, v Elasticsearch-i po indexovaní 650MB.

Indexovanie dát

Pred hromadným importom dát do Elasticsearch-u je potrebné nastaviť mapovanie a analyzátor pre jednotlivé atribúty datasetu.

Analyzátor a tokenizátor

Pre správne fungovanie full-textového vyhľadávania sme použili niekoľko textových analyzátorov, ktoré sú poskytnuté Elasticsearch-om. Každý analyzátor používa štandardný tokenizátor a filtre na odstránenie stop slov a štandardný stemmer pre anglický jazyk. Ascii-folding filter slúži na konverziu Unicode znakov, ktoré nie sú z rozsahu 127 ASCII do ich ASCII ekvivalentu

Autocomplete analyzátor bol vytvorený za účelom automatického dopĺňovania slov pri písaní dopytu počas vyhľadávania. Na konci avšak nebol použitý, pretože bol aplikovaný na celý text článku a neposkytoval dostatočne dobré výsledky. Fungoval dobre pri dopyte „more like this“, ktorý bude opísaný detailne neskôr. Navyše, používa vlastný *autocomplete_filter* typu *edge_ngram*.

Edge_ngram analyzátor používa vlastný tokenizátor *my_edge_ngram*, ktorý je typu *edgeNGram* a slúži na analyzovanie slov po znakov, pričom maximálna dĺžka gramu je 10 znakov. Navyše, používa filter, ktorý transformuje všetky znaky na malé písmená.

Keyword analyzátor používa keyword tokenizátor.

Dopyt na vytvorenie nastavení analyzátorov a tokenizátorov:

² <http://www.json.org/>

```
{
  "settings": {
    "analysis": {
      "analyzer": {
        "autocomplete": {
          "type": "custom",
          "tokenizer": "standard",
          "filter": [
            "asciifolding",
            "my_stop",
            "my_stemmer",
            "autocomplete_filter"
          ]
        },
        "keyword": {
          "type": "custom",
          "tokenizer": "keyword",
          "filter": [
            "asciifolding",
            "my_stop",
            "my_stemmer"
          ]
        },
        "edge_ngram": {
          "type": "custom",
          "tokenizer": "my_edge_ngram",
          "filter": [
            "asciifolding",
            "my_stop",
            "my_stemmer",
            "lowercase"
          ]
        }
      },
      "tokenizer": {
        "my_edge_ngram": {
          "type": "edgeNGram",
          "min_gram": 1,
          "max_gram": 10,
          "token_chars": []
        }
      },
      "filter": {
        "my_stop": {
          "type": "stop",
          "stopwords": "_english_"
        },
        "my_stemmer": {
          "type": "stemmer",
          "name": "english"
        },
        "autocomplete_filter": {
          "type": "edge_ngram",
          "min_gram": 1,
          "max_gram": 20
        }
      }
    }
  }
}
```

Mapovanie

Mapovanie vie odhadnúť Elasticsearch zo štruktúry dát. Toto však často nestačí, ak potrebujeme nastaviť iné dátové typy, či rôzne analyzátory pre atribúty. V našom prípade bolo potrebné nastaviť iné ako predvolené analyzátory. Naš index používa iba jeden typ *articles* a má niekoľko vlastností (angl. properties), ktoré predstavujú atribúty datasetu. Nastavenie `_size` v mapovaní umožní neskôr analyzovať dĺžku uložených reťazcov. Dopyt pre vytvorenie mapovania:

```
{
  "articles": {
    "_size": {
      "enabled": true
    },
    "properties": {
      "article": {
        "type": "string",
        "analyzer": "autocomplete"
      },
      "author": {
        "type": "string",
        "analyzer": "edge_ngram"
      },
      "categories": {
        "type": "string",
        "analyzer": "keyword"
      },
      "image_link": { "type": "string" },
      "timestamp": {
        "type": "date",
        "format": "strict_date_optional_time||epoch_millis"
      },
      "title": {
        "type": "string",
        "analyzer": "edge_ngram"
      },
      "url": { "type": "string" }
    }
  }
}
```

Do takto vytvoreného indexu s mapovaním môžeme pomocou metódy `_bulk` hromadne importovať náš dataset. Elasticsearch automaticky vytvorí invertovaný index, ktorý si udržiava v hlavnej pamäti počítača, pričom vytvára repliku dát na disk pre prípad poruchy. Dáta boli importované programom *curl*³:

```
curl -s -XPOST localhost:9200/_bulk --data-binary "@articles.json"; echo
```

Pre zrýchlenie opakovaného vyhľadávania sme aktivovali ukladanie výsledkov do medzi-pamäte:

```
{
  "settings": { "index.requests.cache.enable": true }
}
```

³ <https://curl.haxx.se/>

Aplikácia

Výsledkom projektu je funkčná webová aplikácia, ktorá poskytuje pohodlný spôsob zadávania dopytov vo formáte známych vyhľadávacích služieb ako Google, či Yahoo. Dopyty sú na pozadí preložené do formátu vhodného pre Elasticsearch, ktorý tiež zabezpečuje full-textové vyhľadávanie a získavanie výsledných dokumentov k používateľskému dopytu. Aplikácia poskytuje:

- automatické dopĺňovanie slov počas písania dopytu (angl. autocomplete).
- full-textové vyhľadávanie, ktoré zohľadňuje preklepy.
- listovanie vo výsledkoch vyhľadávania.
- zobrazenie obsahu článku a šesť podobných článkov.
- vyhľadávanie článkov publikovaných len vo zvolenom dátume.
- popisné štatistiky datasetu.

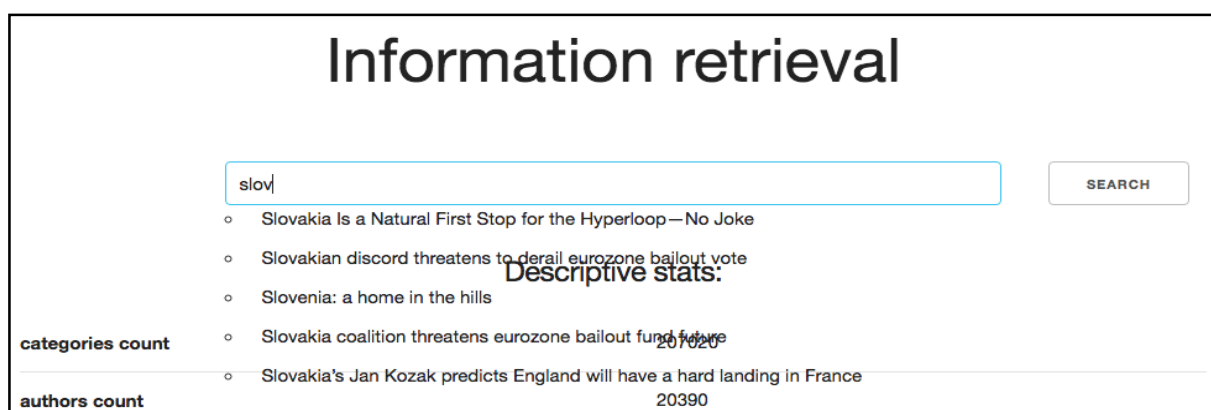
Aplikácia bola implementovaná v jazyku Python, pre API bol použitý micro web-rámec Flask⁴, na zobrazenie výsledkov HTML šablóna Skeleton⁵, jQuery⁶ a Highcharts⁷.

Automatické dopĺňovanie slov počas písanie dopytu

V momente keď používateľ začne písať vyhľadávací dopyt aplikácia začne navrhovať slová, ktoré tento dopyt dopĺňajú. Dopyt je vykonaný nad nadpisom článku. Existujú dve možnosti ako túto funkcionality docieľiť v Elasticsearch-i:

- vyhodnotenie v čase dopytu (angl. query-time) použitím špeciálneho dopytu *match_phrase_prefix*, ktorý má dva parametre *query* a *max_expansions*.
- príprava počas vytvárania indexu (angl. index-time) s použitím analyzátora.

My sme zvolili druhú možnosť s použitím vlastného analyzátora *edge_ngram*. Z výsledku získavame 5 atribútov *title*, ktoré sú zoradené podľa skóre a dátumu publikácie článku. Screenshot aplikácie:



Obrázok 1: Automatické doplnenie slova slov

⁴ <http://flask.pocoo.org/>

⁵ <http://getskeleton.com/>

⁶ <https://jquery.com/>

⁷ <http://www.highcharts.com/>

Dopyt vykonaný nad Elasticsearch-om:

```
{
  "size": 5,
  "fields": [ "title" ],
  "query": {
    "match": { "title": query }
  },
  "sort": [
    {
      "_score": {
        "order": "desc"
      }
    },
    {
      "timestamp": {
        "order": "desc",
        "mode": "max"
      }
    }
  ]
}
```

Full-textové vyhľadávanie, ktoré zohľadňuje preklepy

Pri full-textovom vyhľadávaní je potrebné zohľadniť ľudské chyby a preklepy pri písaní. Ďalšou výzvou je usporiadanie výsledkov vyhľadávania. Okrem toho, že výsledky vyhľadávania môžu byť do istej miery personalizované, najviac relevantné výsledky by sa mali umiestniť najvyššie.

Prvú požiadavku sme zabezpečili pridaním fuzzy faktora do vyhľadávacieho dopytu. Znamená to, že dopyt zohľadňuje preklepy ako napríklad: *slovaakia* bude Elasticsearch-om chápaný ako *slovakia*.





Relevantné usporiadanie výsledkov vyhľadávania je zabezpečené zvyšovaním (angl. boosting) váhy výsledkov, ktoré sú novšie. Toto zabezpečí, že novšie články budú na lepších pozíciách, pričom usporiadanie najprv podľa skóre zhody dokumentu, zohľadní relevanciu výsledkov. Vyhľadávanie je vykonávané na atribútoch: *title*, *article* a *author*. Pri každom výsledku je zobrazený čas trvania dopytu, ktorý bol vo väčšine prípadov menej ako 300ms. Ukážka výsledkov vyhľadávania pre dopyt *slovvak*:

IR

slovvak

SEARCH

Results: 69202 (0.364s)

	<p>Slovakia Is a Natural First Stop for the Hyperloop—No Joke</p> <p>Author: Alex Davies,</p> <p>When Elon Musk proposed his wild idea for the Hyperloop almost four years ago, he billed it as an unbelievably cool way to get from San Francisco to Los Angeles in 30 minutes. But ...</p>	<p>Tags: Elon Musk, Future Transport, hyperloop, Infrastructure, post-Soviet</p> <p>Date: 2016-03-11T07:00:31+00:00</p>
	<p>Slovakian discord threatens to derail eurozone bailout vote</p> <p>Author:</p> <p>Hopes of securing agreement on reforms to the eurozone's bailout package will go down to the wire on Tuesday after the warring partners in Slovakia's ruling coalition failed to rea...</p>	<p>Tags: Eurozone crisis Europe Slovakia</p> <p>Date: 2011-10-10T19:02:34.000Z</p>
	<p>Slovakia coalition threatens eurozone bailout fund future</p> <p>Author: Helen Pidd,</p> <p>Slovakia's governing coalition has failed to strike a deal to prevent the collapse of a continent-wide plan to rescue heavily indebted European nations. Prime Minister Iveta Radicov...</p>	<p>Tags: Slovakia Eurozone crisis Europe Euro (World news) Euro (Business) Currencies European Union Economics</p> <p>Date: 2011-10-10T23:46:52.000Z</p>
	<p>Slovakia's Jan Kozak predicts England will have a hard landing in France</p> <p>Author: David Hytner,</p> <p>s arrivals go it was hardly one that sent a grand statement. The plane that carried Jan Kozak and his squad from Bratislava to Dublin on Monday morning entered Irish air space to ...</p>	<p>Tags: Slovakia Republic of Ireland Friendlies</p> <p>Date: 2016-03-28T21:48:57.000Z</p>

Obrázok 2: Výsledky vyhľadávania pre dopyt *slovvak*

Dopyt vykonávaný nad Elasticsearch-om:


```
{
  "from": start, "size": size,
  "query": {
    "bool": {
      "must": {
        "multi_match": {
          "fields": ["title", "article", "author"],
          "query": query,
          "fuzziness": "AUTO"
        }
      },
      "should": [
        {
          "range": {
            "timestamp": {
              "boost": 5,
              "gte": "now-90d/d"
            }
          },
          "range": {
            "timestamp": {
              "boost": 2,
              "gte": "now-12m/d"
            }
          },
          "range": {
            "timestamp": {
              "boost": 1,
              "gte": "now-24m/d"
            }
          }
        }
      ]
    }
  },
  "sort": [
    {"_score": {"order": "desc"}},
    {"timestamp": {
      "order": "desc",
      "mode": "max"
    }}
  ],
  "highlight": {
    "fields": {
      "title": {},
      "article": {},
      "author": {}
    }
  }
}
```


Listovanie vo výsledkoch vyhľadávania je zabezpečené zmenou parametrov *start* a *size*, pričom *size* je v našom prípade vždy 10. V odpovedi Elasticsearch-u sú tiež zvýraznené (angl. highlight) zhodné termíny.


Zobrazenie článku a jemu šesť podobných článkov


Po kliknutí na článok z výsledkov vyhľadávania je zobrazený tento článok aj s jeho obsahom. Kvôli nastaveniu analyzátorov na full-textové vyhľadávanie spolu s automatickým dopĺňaním, je vyhľadávanie presného článku realizované pomocou hľadania zhody v *url*, ktorá by mala byť vždy unikátna. V tomto prípade by nefungoval správne ani dopyt *exact_match*, ktorý síce poskytuje Elasticsearch, pretože na atribúty je použitý *edge_ngram* analyzátor. Pri každom článku je zobrazených šesť jemu podobných článkov na základe obsahu článku (atribút *article*). Screenshot z aplikácie pre článok s nadpisom „Slovakia Is a Natural First Stop for the Hyperloop—No Joke“:


More like this:


[Hyperloop pens deal that could connect Slovakia, Austria and Hungary](#)

[These central European countries may be the next to get a Hyperloop](#)

[Slovakia Is a Natural First Stop for the Hyperloop—No Joke](#)

[Take a look inside this Hyperloop pod to allay your claustrophobia](#)

[Hyperloop startup selects Vibranium for pods because it's good enough for Captain America](#)

[Hyperloop Transportation says it will use a 'cheaper, safer' form of magnetic levitation](#)

Obrázok 3: Podobné články k článku *Slovakia Is a Natural First Stop for the Hyperloop—No Joke*

Dopyt pre vyhľadanie konkrétneho článku (*unquote_plus* je funkcia Python-u):

```
{
  "size": 1,
  "query": {
    "match": {
      "url": unquote_plus(query)
    }
  }
}
```

V rámci zobrazenia článku sa vykoná na pozadí dopyt pre získanie podobných článkov:

```
{
  "size": size,
  "query": {
    "more_like_this": {
      "fields": [
        "article"
      ],
      "like_text": query,
      "min_term_freq": 1,
      "max_query_terms": 12
    }
  }
}
```

Vyhľadávanie článkov pre zvolený dátum

Dopyt, ktorý je použitý pre vyhľadanie článkov publikovaných len v určitý deň je tiež použitý pre vykreslenie grafu, ktorý zobrazuje počet publikovaných článkov v čase. Tento dopyt je natoľko generický, že sa dá použiť na rôznych miestach. V tomto prípade aplikácia zabezpečí správne nastavenie parametrov:

- *size*: 10,
- *date_from*: dopytovaný dátum,
- *date_to*: dopytovaný dátum + 1 deň,
- *interval*: day

Dopyt, zoradenie od najnovších článkov po najstaršie:

```
{
  "size": size,
  "query": {
    "range": {
      "timestamp": {
        "from": date_from,
        "to": date_to
      }
    }
  },
  "aggregations": {
    "articles_over_time": {
      "date_histogram": {
        "field": "timestamp",
        "interval": interval
      }
    }
  },
  "sort": [ { "timestamp": { "order": "asc" } } ]
}
```

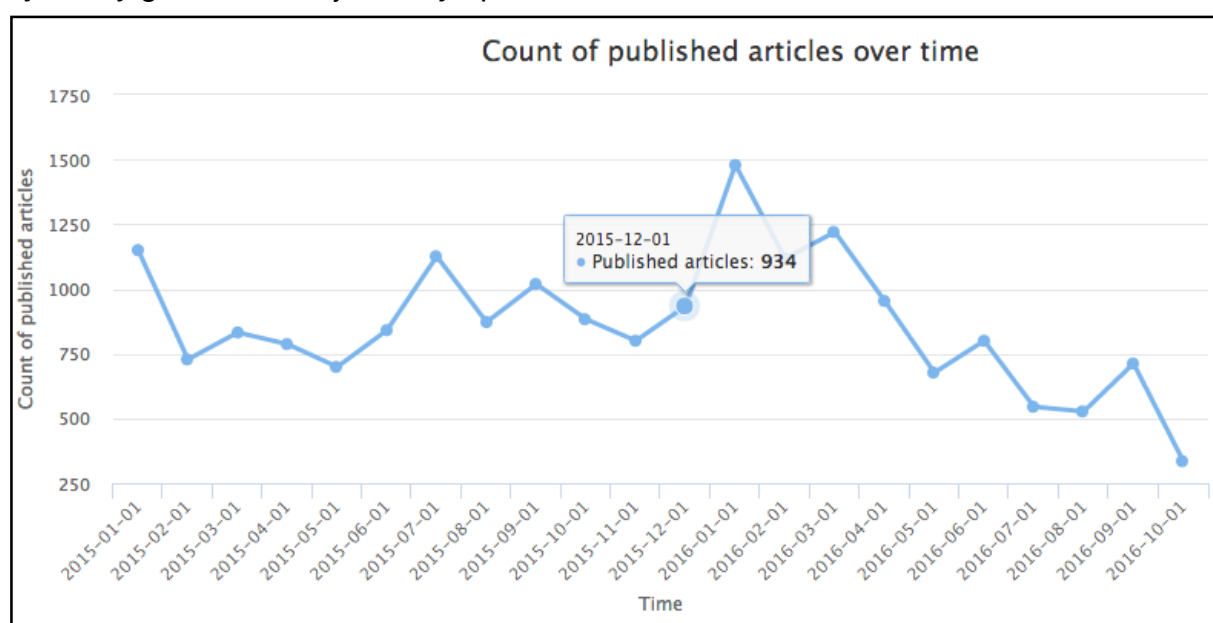
Popisné štatistiky datasetu

Počet publikovaných článkov v čase vyjadruje čiarový graf, ktorý zobrazuje koľko bolo publikovaných článkov v jednotlivých mesiacoch od roku 2015. Štruktúra dopytu je rovnaká ako v prípade vyššie spomenutého vyhľadávania pre konkrétny dátum.

Parametre dopytu:

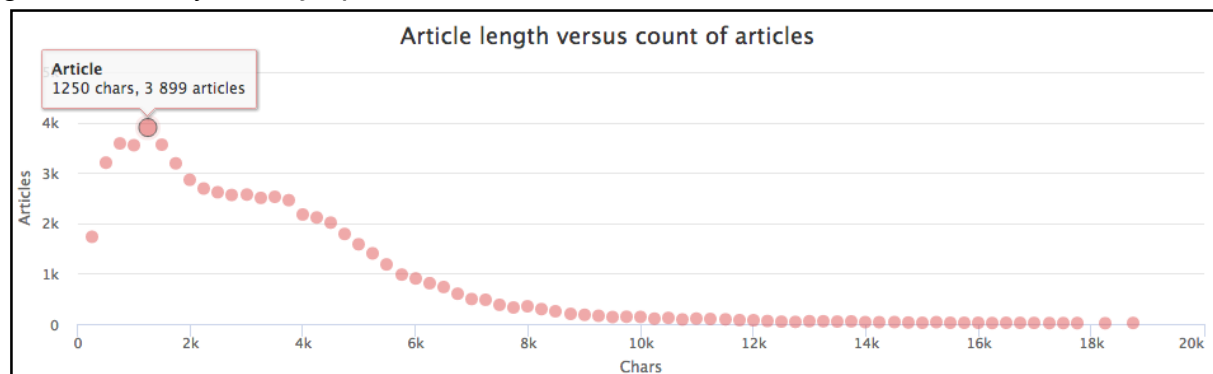
- *size*: 0, - pretože chceme len agregáty,
- *date_from*: 2015,
- *date_to*: 2017,
- *interval*: months – pretože chceme agregáty po mesiacoch.

Výsledný graf z úvodnej stránky aplikácie:



Obrázok 4: Počet publikovaných článkov v čase

Dĺžka článku verzus počet článkov vyjadruje koľko článkov má určitú dĺžku článku. Dĺžka článku je rozdelená po 250 znakov, to znamená 250, 500, 750, ..., N. Výsledný graf z úvodnej stránky aplikácie:



Obrázok 5: Dĺžka článku verzus počet článkov

Dopyt do Elasticsearch-u používa možnosť použitia skriptu v dopyte pre zistenie dĺžky článku (atribút *article*), histogram je rozdelený po intervaloch 250 znakov. Vo výslednom grafe sú orezané outliers. Dopyt:

```
{
  "size": 0,
  "aggs" : {
    "articles_ln_histogram" : {
      "histogram" : {
        "script" : "_source.article.toString().length()",
        "interval" : 250
      }
    }
  }
}
```

Sumárne štatistiky datasetu hovoria o počte dokumentov (článkov), počte autorov a kategórií, priemernej, maximálnej a agregovanej dĺžke článkov:

Descriptive stats:	
categories count	207020
authors count	20390
articles count	85284
max article length	59127 chars
sum article length	231339766 chars
avg article length	2713 chars

Dopyt do Elasticsearch-u opäť používa funkcionálnosť skriptovania pre získanie dĺžky článku:

```
{
  "size": 0,
  "query" : {"match_all" : {}},
  "aggs":{
    "articles_stats" : {
      "stats" : {
        "script" : "_source.article.toString().length()"
      }
    },
    "authors_count" : {
      "cardinality" : {
        "field" : "author"
      }
    },
    "categories_count" : {
      "cardinality" : {
        "field" : "categories"
      }
    }
  }
}
```

Zhodnotenie

V projekte sa nám úspešne podarilo extrahovať dáta článkov zo štyroch svetových webov. Získané dáta boli importované do Elasticsearch-u, ktorý vytvoril nad obsahom dát invertovaný index. Nad takýmto indexom sme mohli jednoducho vykonávať full-textové vyhľadávanie s automatickým dopĺňaním počas písania dopytu. Tiež sme boli schopný jednoducho nájsť obsahovo podobné články k aktuálne prezeranému článku, či zobraziť základné opisné štatistiky datasetu.