

České vysoké učení technické v Praze
Fakulta Informačních technologií

Vyber dobrých knih pro čtení

Semestrální práce
BI-BIG
Sukha Viktoriia
6.12.2020

Konec stránky

Rejstřík

Rejstřík	2
Úvod	3
Hlavní část	3
Popis dat	3
Ukázka dat z každého datasetu	4
Načtení datasetu	4
Agregace datasetu	5
Agregace z 1 původního datasetu	5
Agregace ze 2 datasetů najednou, z čehož jeden bude výsledkem předchozí agregace	5
Agregace ze 2 původních datasetů najednou	5
Save files	6
Kibana	6
Dotazy do indexu	6
Filtrování	6
Třídění	7
Wildcard hledání	7
Grafické zobrazení v Kibane	8
Závěr	10

Úvod

Pro semestrální práci vybrala jsem datasety z kibanny, které obsahují informace o dobrých knihách za celou historie. Tento dataset obsahuje 10000 knih z plnou informace o nich, také je hodnoceni od jiných lide, a seznam knih, které lidé nejčastěji chtějí přečíst. Na základě těchto datasetu vytvořila jsem jiné datasety a různé vizualizace ze kterých, je vidět kdo napsal nejvíce dobrých knih a v jakých rocích. V jakých rocích psaly nejvíce knih, jaké knihy chtějí číst lidé teď a mnoho jiného.

Hlavní část

Popis dat

Mám tři datasety: **books**, **ratings** a **want_to_read**.

V hlavním datasetu books mám takové sloupce:

- id (integer) - číslo řádky
- book_id (integer) - id knihy
- books_count (integer) - počet vydání
- authors (string) - jména autorů
- original_publication_year (integer) - data publikace knihy
 - omezení: menší než aktuální rok
- original_title (string)- originální název
- title (string) - název na jazyce překladu
- language_code (string) - kód jazyku na kterém napsaná kniha

V datasetu ratings je tři sloupce:

- book_id (integer) - index knihy
- user_id (integer) - index uživatele
- rating (integer) - hodnocení knihy uživatelem
 - omezení: čísla od 1 do 5

V datasetu want_to_read je dva sloupce:

- book_id (integer) -index knihy
- users_want_to_read (integer) - počet lidí které chtějí přečíst knihu z daným indexem

Ukázka dat z každého datasetu

books

id	book_id	count_of_edition	authors	original_publication_year	original_title	title	language_code
1	2767052	272	Suzanne Collins	2008.0	The Hunger Games	The Hunger Games ...	eng
2	3	491	J.K. Rowling, Mar...	1997.0	Harry Potter and ...	Harry Potter and ...	eng
3	41865	226	Stephenie Meyer	2005.0	Twilight	Twilight (Twiligh...	en-US
4	2657	487	Harper Lee	1960.0	To Kill a Mocking...	To Kill a Mocking...	eng
5	4671	1356	F. Scott Fitzgerald	1925.0	The Great Gatsby	The Great Gatsby	eng

only showing top 5 rows

ratings

book_id	user_id	rating
1	314	5
1	439	3
1	588	5
1	1169	4
1	1185	4

only showing top 5 rows

want_to_read

book_id	users_want_to_read
1645	97
1591	95
1238	289
471	375
7754	71

only showing top 5 rows

Načtení datasetu

Ve Sparku nacetla jsem 3 datasety:

```
val books = spark.read.format("csv")  
  .option("sep", ",")  
  .option("inferSchema", "true")
```

```
.option("header", "true")
.load("books.csv")

val ratings = spark.read.format("csv")
.option("sep", ",")
.option("inferSchema", "true")
.option("header", "true")
.load("ratings.csv")

val to_read = spark.read.format("csv")
.option("sep", ",")
.option("inferSchema", "true")
.option("header", "true")
.load("to_read.csv")
```

Agregace datasetu

Agregace z 1 původního datasetu

První agregace udělala jsem z tabulky ratings. Měla jsem hodnocení uživatelů o knihách a výpočítala střední rating každé knihy, pomocí grupování po book_id a agregační funkce. Tím vytvořila tabulku **average_rating**.

```
val average_rating = ratings.groupBy("book_id").agg(avg("rating"))
```

Agregace ze 2 datasetů najednou, z čehož jeden bude výsledkem předchozí agregace

V druhé agregace spojila jsem tabulky average_rating a books pomocí book_id pak zgrupovala podle autorů a spočítala střední rating autorů. Tím vytvořila tabulku **rating_of_autors**.

```
al books_with_rating = filtred_books.join(average_rating, filtred_books("book_id") ===
average_rating("book_id")).withColumnRenamed("avg(rating)","rating")
val rating_of_autors = books_with_rating.groupBy("authors").agg(avg("rating"))
```

Agregace ze 2 původních datasetů najednou

V třetí agregace spojila jsem tabulky want_to_read a books pomocí book_id pak zgrupovala podle roku publikace a spočítala kolik lidí chtějí přečíst knihy, napsané v nějakém roce. A tím vytvořila tabulku **popular_year**.

```
val books_with_to_read = filtered_books.join(want_to_read, filtered_books("book_id") === want_to_read("book_id"))

val popular_year =
books_with_to_read.groupBy("original_publication_year").agg(sum("count(1)").withColumnRenamed("sum(count(1))", "count_to_read"))
```

Save files

Pak stáhla jsem vytvořené soubory na disk:

```
popular_year.coalesce(1).write.format("csv")
.option("sep", ",")
.option("header", "true")
.save("popular_year")

rating_of_authors.coalesce(1).write.format("csv")
.option("sep", ",")
.option("header", "true")
.save("rating_of_authors")
```

Kibana

Připojila jsem Kibana k indexu v Elasticsearch pomocí LogStash. Vytvorila jsem tři indexy: `all_books`, `popular_year`, `rating_of_authors` z datasetu `books` (původní dataset), `popular_year` a `rating_of_authors` (agregační datasety), konfigurační soubor je v příloze.

Dotazy do indexu

V console elasticsearch - "Dev Tools" udělala jsem dotazy nad indexy.

Filtrování

Z indexu `all_books` vypsala 5 knih, které byly napsané z 2005 po 2010 rok

```
1 GET /_search
2 {
3   "query": {
4     "match_all": {}
5   }
6 }
7
8 GET all_books/_search
9 {
10  "query": {
11    "range": {
12      "original_publication_year": {"gte": 2005, "lte": 2010}
13    }
14  },
15  "size": 5
16 }
17
18 GET rating_of_authors/_search
19 {
20   "size": 10,
21   "sort": [
22     { "avg(rating)": "desc" }
23   ]
24 }
25
26 GET all_books/_search
27 {
28   "query": {
29     "wildcard": {
30       "title": "potter*"
31     }
32   }
33 }
34
35
36
37
38
```

```
1 {
2   "took": 17,
3   "timed_out": false,
4   "_shards": {
5     "total": 5,
6     "successful": 5,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": 2114,
12    "max_score": 1,
13    "hits": [
14      {
15        "_index": "all_books",
16        "_type": "doc",
17        "_id": "99NTOHBS4ato8KYUK0_",
18        "_score": 1,
19        "_source": {
20          "title": "The Name of the Wind (The Kingkiller Chronicle, #1)",
21          "language_code": "eng",
22          "books_count": 123,
23          "message": "'192,186074,123,Patrick Rothfuss,The Name of the Wind,The Name of the Wind (The Kingkiller Chronicle, #1),eng,2007'",
24          "tags": [
25            "books"
26          ],
27          "id": 192,
28          "original_title": "The Name of the Wind",
29          "authors": "Patrick Rothfuss",
30          "book_id": 186074,
31          "timestamp": "2020-12-06T13:52:31.814Z",
32          "path": "/usr/share/logstash/data/books.csv",
33          "version": "1",
34          "original_publication_year": 2007,
35          "host": "d17b4ec65c88"
36        },
37      },
38      {
39        "_index": "all_books",
40        "_type": "doc",
41        "_id": "99NTOHBS4ato8KYUK5_",
42        "_score": 1,
43        "_source": {
44
```

Třídění

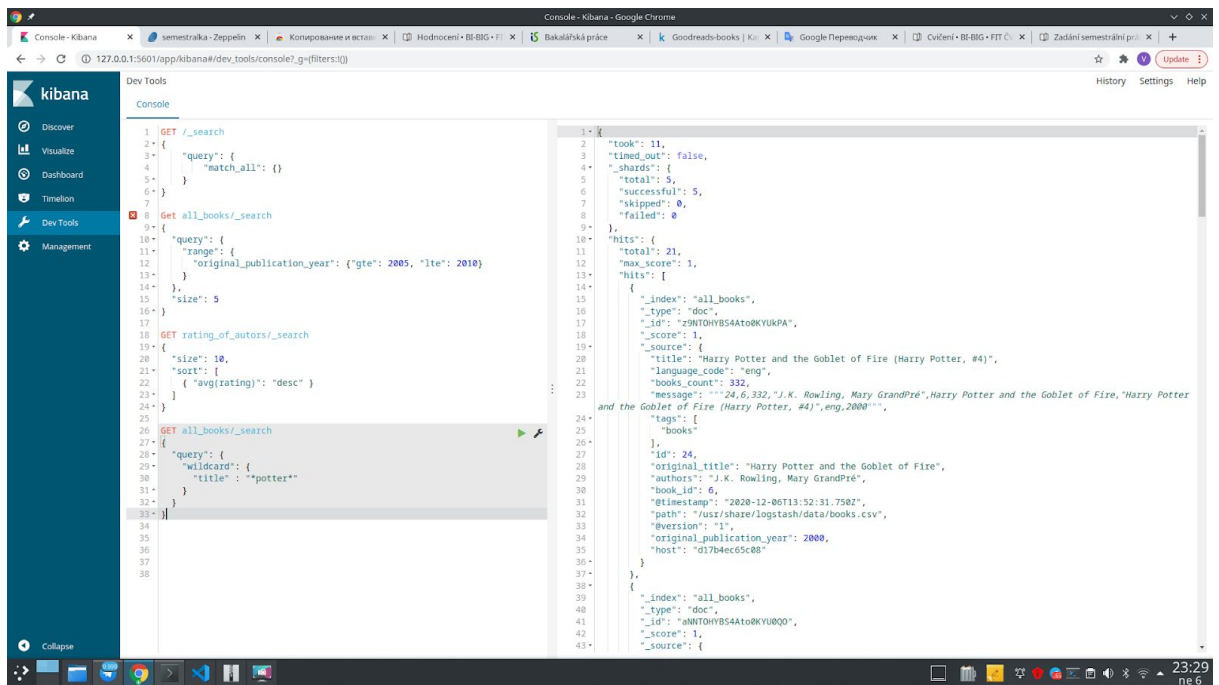
Index rating_of_authors setřídila podle hodnocení

```
1 GET /_search
2 {
3   "query": {
4     "match_all": {}
5   }
6 }
7
8 GET all_books/_search
9 {
10  "query": {
11    "range": {
12      "original_publication_year": {"gte": 2005, "lte": 2010}
13    }
14  },
15  "size": 5
16 }
17
18 GET rating_of_authors/_search
19 {
20   "size": 10,
21   "sort": [
22     { "avg(rating)": "desc" }
23   ]
24 }
25
26 GET all_books/_search
27 {
28   "query": {
29     "wildcard": {
30       "title": "potter*"
31     }
32   }
33 }
34
35
36
37
38
```

```
1 {
2   "took": 17,
3   "timed_out": false,
4   "_shards": {
5     "total": 5,
6     "successful": 5,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": 439,
12    "max_score": null,
13    "hits": [
14      {
15        "_index": "rating_of_authors",
16        "_type": "doc",
17        "_id": "99NTOHBS4ato8KYVEU3_",
18        "_score": null,
19        "_source": {
20          "authors": "Dave Barry, Ridley Pearson, Greg Call",
21          "message": "'4.557142857142857,4.557142857142857'",
22          "timestamp": "2020-12-06T13:52:31.821Z",
23          "avg(rating)": 4.557142857142857,
24          "path": "/usr/share/logstash/data/rating_of_authors.csv",
25          "version": "1",
26          "tags": [
27            "authors"
28          ],
29          "host": "d17b4ec65c88"
30        },
31        "sort": [
32          4.5571427
33        ]
34      },
35      {
36        "_index": "rating_of_authors",
37        "_type": "doc",
38        "_id": "99NTOHBS4ato8KYVEZE_",
39        "_score": null,
40        "_source": {
41          "authors": "Howard Zinn",
42          "message": "Howard Zinn,4.54",
43          "timestamp": "2020-12-06T13:52:31.893Z",
44          "avg(rating)": 4.54,
45          "path": "/usr/share/logstash/data/rating_of_authors.csv",
46          "version": "1",
47
```

Wildcard hledání

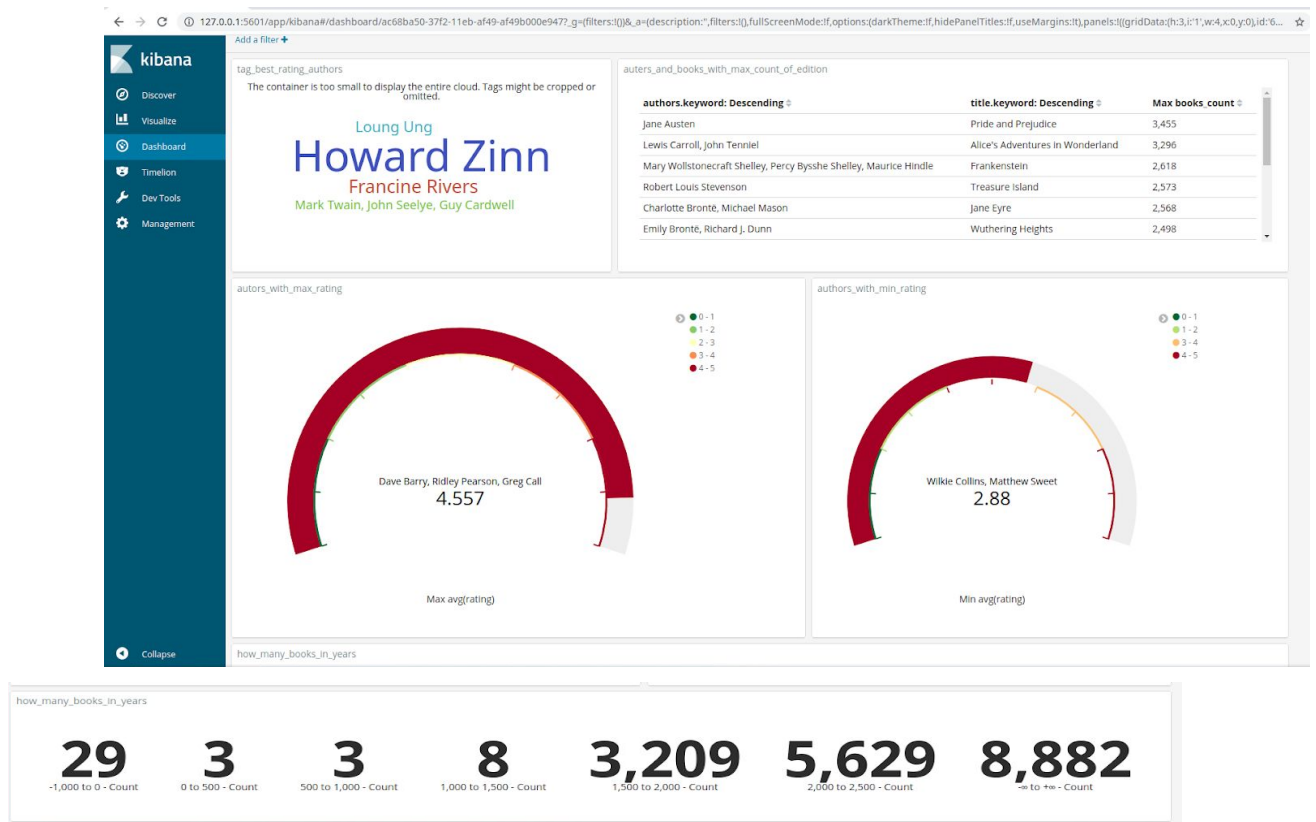
V indexu all_books nasla vsechny knihy, ktere v title obsahuji slovo "potter"



Graficke zobrazeni v Kibane

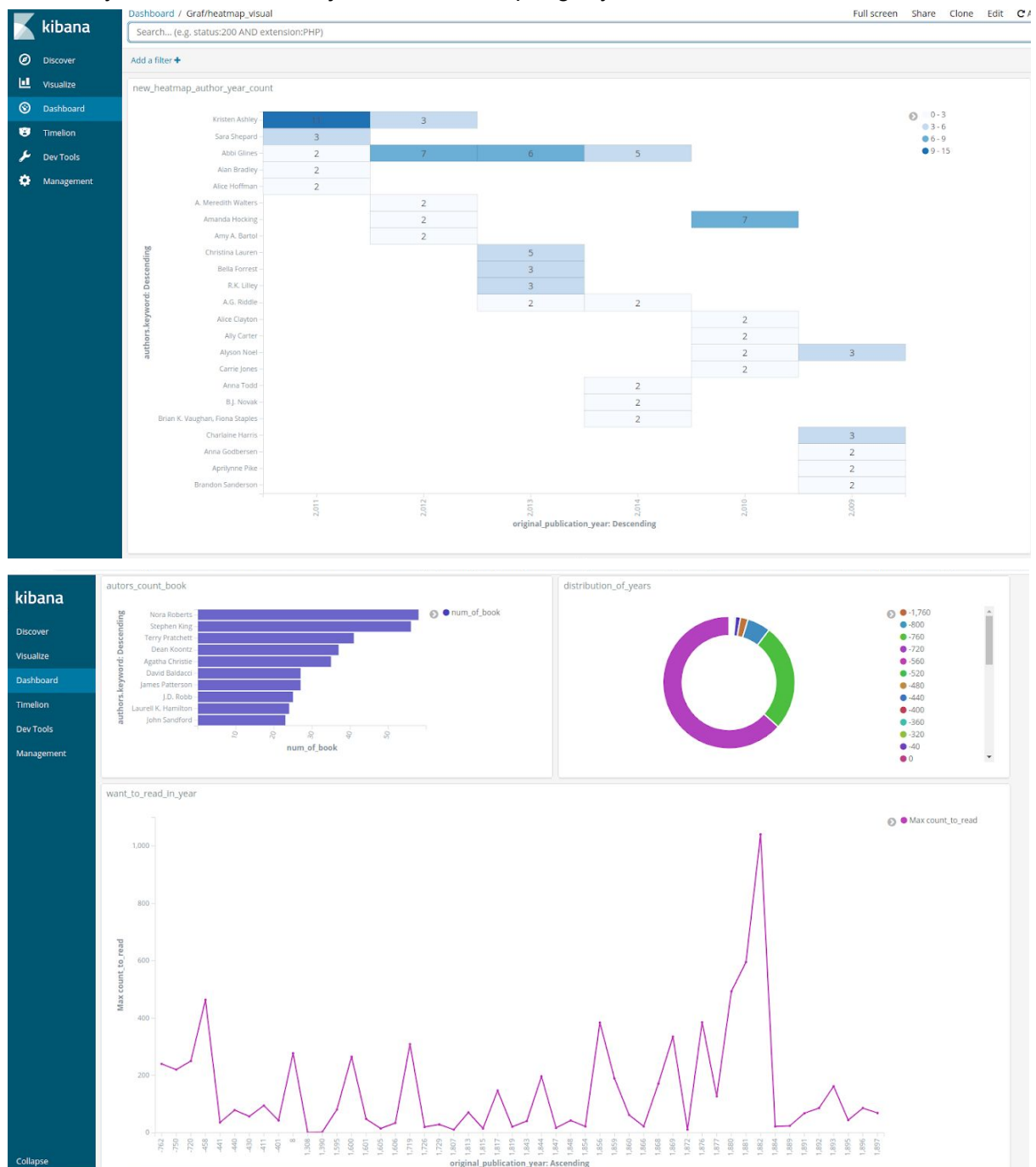
Vytvorila jsem dva dashbordy.

Prvni obsahuje ruzne pohledy:



Tady na prvním pohledu je vidět tag cloud z autory z nejlepším hodnocením, na druhém tabulku z autory a knihy z maximálním počtem vydání na třetím a čtvrtém autory z nejlepším a nejnižším hodnocením. Pak počet knih které byly vydané v různých rozmrzích roky.

Pak druhý dashboard obsahuje různé heatmap a grafy.



Je vidět heatmapu ze které můžeme dozvědět jaké autory kolik knih napsaly v jakých rocích.

Pak je graf na kterém je videt jaký procent knih v jakem roce napsali. Pak je graf s zobrazením autorů které napsali nejvíce knih a kolik přesně oni je napsali.

Pak je line graf, na kterém je vidět kolik knih v jakem roce napsali a můžeme vidět v jakých rocích napsaly knih nejvíc

Zaver

V teto prace vytvorila jsem datasety a vizualizace ze kterych je mozne vybirat dobre knihy pro cteni. Zlepsila jsem svoji umeni v Kibane, Elasticsearch a logStash