**Slovenská technická univerzita v Bratislave**

**Fakulta informatiky a informačných technológií**

**Ilkovičova 2 , 842 16 Bratislava**

**Zadanie č.1**

Crawler a ElasticSearch

**Cvičiaci:​** doc. Ing. Michal Kompan, PhD.

Cvičenie: Ut, 8:00  
**Autor:​** Bc. Jozef Varga   
**Študijný program:​** Inteligentné softvérové systémy  
**Predmet:​** Vyhľadávanie informácií  
**Akad. rok:​** 2019/2020

Obsah

[Crawler stránky csfd.cz 3](#_Toc22571563)

[Csfd.cz crawlovanie dát 3](#_Toc22571564)

[Uložený záznam 4](#_Toc22571565)

[Vyhľadávanie a ElasticSearch 5](#_Toc22571566)

[Scenáre 6](#_Toc22571567)

[Prvý scenár 6](#_Toc22571568)

[Druhý scenár 7](#_Toc22571569)

[Tretí scenár 8](#_Toc22571570)

[Štvrtý scenár 8](#_Toc22571571)

[Piaty Scenát 8](#_Toc22571572)

[Šiesty scenár 8](#_Toc22571573)

[Vizualizácie 9](#_Toc22571574)

[Vizualizácia č.1 9](#_Toc22571575)

[Vizualizácia č.2 10](#_Toc22571576)

[Vizualizácia č.3 11](#_Toc22571577)

[Príloha A) Mapovanie 12](#_Toc22571578)

[Príloha B) Query scenárov 16](#_Toc22571579)

[Query č. 1 16](#_Toc22571580)

[Query č. 2 17](#_Toc22571581)

[Query č. 3 18](#_Toc22571582)

[Query č. 4 19](#_Toc22571583)

[Query č. 5 20](#_Toc22571584)

[Query č. 6 21](#_Toc22571585)

[Príloha C) Zoznam súborov 22](#_Toc22571586)

# Crawler stránky csfd.cz

Dáta ktoré som na tento projekt potreboval som crawloval na stránke csfd.cz (URL stránky: <https://www.csfd.cz>). Crawler, ktorý tieto dáta stiahol, bol vytvorený v jazyku Python s využitím framework-u scrapy (<https://scrapy.org>). Ku sťahovaniu dát som pristupoval dvoma spôsobmi. Prvým spôsobom som si stiahol všetky celé html. Druhým som si rovno pri sťahovaní vyextrahoval potrebné dáta (názov filmu, hercou, obsah ...) a uložil do JSON formátu. Prvý spôsob bol len ako poistka, ak by sa niečo nepodarilo pri extrahovaní, aby som to potom mohol extrahovať priamo nad uloženými dátami. Vďaka využitiu Python-u bola práca s JSON formátom veľmi jednoduchá.

# Csfd.cz crawlovanie dát

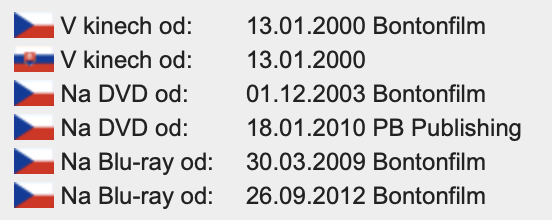
Dáta z tejto stránky som sťahoval priamo cez ich interné vyhľadávanie. Tu nastal hneď prvý problém. Pri vyhľadaní všetkých filmov, sa zobrazí na stránke 50 záznamov, avšak stránkovanie s filmami je povolené len do strany 20. Tým pádom pri jednom vyhľadaní viem nájsť len 1000 záznamov (50\*20 = 1000). Tento problém som vyriešil tak, že pri každom vyhľadávaní zadám iný rok a prejdem celých 20 stránok. Pri vyhľadávaní som zadával roky od 2000 po 2012 a dokopy som stiahol 12986 záznamov. Tieto záznamy tvorili informácie z html ktoré som extrahoval napríklad takto:

'film\_genre': response.css('.genre::text').extract\_first().split(' / '),

Hore vidíme extrahovanie žánra filmu. Extrahované to bolo podľa class v html, ktorá má názov „genre“ a následne, keďže sa jedná o viacero žánrov, bol text rozdelený podľa „/“ aby sme dostali jednotlivé žánre (viď html nižšie).

<p class="genre">Akční / Dobrodružný / Krimi / Thriller</p>

Niektoré informácie ktoré som crawloval nebolo tak ľahké vytiahnuť, keďže sa jednotlivé html stále menilo s počtom informácií ako napríklad premiéry:



Obrázok Premiéry filmov

Problém s premiérami bol v tom že nie každý film obsahuje všetky tipy premier (na Blue-ray, Na DVD, ...) a tým pádom aj inač vyzerá html. Ja som sa rozhodol sťahovať pre každý typ práve jeden dátum teda:

        for i, element in enumerate(release\_date):

            help\_element = element.replace('\t','').replace('\n','').replace(' ', '#', 1).split('#')[0]

            help\_element = help\_element.split('.')[2] + '-' + help\_element.split('.')[1] + '-' + help\_element.split('.')[0]

            if release\_name[i \* 2 + 1] == 'V kinech od:':

                if alt\_data[i] == 'Česko':

                    in\_cinema\_cz = help\_element

                if alt\_data[i] == 'Slovensko':

                    in\_cinema\_sk = help\_element

                if alt\_data[i] == 'USA':

                    in\_cinema\_usa = help\_element

            if release\_name[i \* 2 + 1] == 'Na DVD od:':

...

            if release\_name[i \* 2 + 1] == 'Na Blu-ray od:':

                ...

Posledným problémom bolo spracovať účinkujúcich, teda herci, hudba, kamera ... Avšak tento problém bol podobný ako pri premiérach. Tu som však ukladal všetkých z danej kategórie.

Do JSON-u som ukladal tieto informácie: Názov, tagy, obsah, priemerné hodnotenie, žáner, dĺžka filmu, krajina pôvodu, rok vydania, réžia, scenár, kamera, hudba, herci, dátum premiéry v kine, na DVD, na Blue-ray a to na Slovensku, Česku a USA.

## Uložený záznam

Takto vyzeral uložčený záznam:

{"film\_name": "Imp\u00e9rium", "film\_genre": ["Krimi", "Drama", "Thriller"], "film\_origin": ["USA"], "film\_length": "96", "film\_year": "2002", "film\_average": "51", "film\_director": ["Franc. Reyes"], "film\_tags": [], "film\_content": ["Hlu\u010dn\u00e9 ulice ji\u017en\u00edho Bronxu se setk\u00e1vaj\u00ed s elegantn\u00edm sv\u011btem Wall Streetu v ak\u010dn\u00edm p\u0159\u00edb\u011bhu Imp\u00e9rium o honb\u011b za rychl\u00fdm ziskem a vysok\u00e9 cen\u011b za chtivost a nenasytnost. Nap\u00ednav\u00fd a poutav\u00fd ak\u010dn\u00ed thriller s Johnem Leguizamo (Moulin Rouge, Carlitova cesta), Denise Richards (James Bond: Jeden sv\u011bt nesta\u010d\u00ed), Isabellou Rossellini (Modr\u00fd samet, Zb\u011bsilost v srdci) a rapov\u00fdmi superstar Fat Joe a Treach. Imp\u00e9rium je rychl\u00fd, strhuj\u00edc\u00ed film, kter\u00fd zaru\u010duje z\u00e1bavu a nap\u011bt\u00ed."], "film\_administration": ["Franc. Reyes"], "film\_camera": ["Kramer Morgenthau"], "film\_scenario": ["Franc. Reyes"], "film\_music": ["Rub\u00e9n Blades"], "film\_actors": ["John Leguizamo", "Peter Sarsgaard", "Denise Richards", "Vincent Laresca", "Isabella Rossellini", "S\u00f4nia Braga", "Delilah Cotto", "Nestor Serrano", "Anthony 'Treach' Criss", "Rafael B\u00e1ez", "Fat Joe", "Jean-Luke Figueroa", "Mike Figueroa", "Nicole Gomez Fisher", "Carlos Leon", "Felix Solis", "Elliot Santiago", "Derrick Simmons", "Sam Coppola", "Monica Steuer", "Granville Adams", "Karina Arroyave"], "film\_in\_cinema\_sk": "0000-00-00", "film\_on\_dvd\_sk": "0000-00-00", "film\_on\_bluray\_sk": "0000-00-00", "film\_in\_cinema\_cz": "0000-00-00", "film\_on\_dvd\_cz": "2013-01-13", "film\_on\_bluray\_cz": "0000-00-00", "film\_in\_cinema\_usa": "2002-12-06", "film\_on\_dvd\_usa": "0000-00-00", "film\_on\_bluray\_usa": "0000-00-00"},

Pred tým ako som však vložil dáta do ElasticSearchu, som prišiel na dve chyby. Prvou chybou bol dátum, ktorý ak záznam neobsahoval, som uložil ako "0000-00-00“. Tento problém som vyriešil nahradením všetkých záznamov za rok "1900-01-01“. Keďže prvý rok, ktorý beriem do úvahy, musí byť väčší ako rok 2000. Teda staršie filmy ne-crawlujem. Druhý problém bol zlý formát dĺžky filmu, roku vydania a priemerného hodnotenia. Tento problém som vyriešil pri úprave JSON-nu na formát, ktorý akceptuje „\_bulk“. Upravené boli nasledovne:

for i, value in enumerate(data\_json):

    try:

        data\_json[i]['film\_length'] = int(value['film\_length'])

        data\_json[i]['film\_year'] = int(value['film\_year'])

        data\_json[i]['film\_average'] = int(value['film\_average'])

    except:

        del data\_json[i]

# Vyhľadávanie a ElasticSearch

Dáta v ktorých som vyhľadával som stiahol podľa informácií vyššie. Aby ich ElasticSearch akceptoval v \_bulk potreboval som ich upraviť na tvar:

{"index":{"\_id":"1"}}

{"film\_name": "Formule.....

{"index":{"\_id":"2"}}

{"film\_name": "Velk.....

{"index":{"\_id":"3"}}

{"film\_name": "Vian...

Tento tvar som dosiahol pomocou Pythonu kde som si celý JSON roz-parsoval a pridal potrebné riadky.

Na prácu s ElasticSearchom som využil aplikáciu Postman (<https://www.getpostman.com/>) a kibana Dev Tools.

Následne som potreboval vytvoriť index s mapping-om a základnými nastaveniami (viď. Prílohu A)

Potom som naplnil ElasticSearch dátami pomocou príkazu, ktorý v tele obsahoval dáta môjho JSON-u:

localhost:9200/varga\_vi\_zadanie\_1/\_bulk?pretty&refresh

Pri mappingu som priamo aj nastavil analyzer. Ďalšie možnosti nastavenia analyzátoru môžu byť napríklad využitie slovníka pre češtinu Hunspell (<http://hunspell.github.io> - využíva to aj OpenOffice). Ja som využil české nastavenia priamo od ElasticSearchu. Taktiež som využil lowercase a only\_on\_same\_position kvôli duplicitným slovám, kde bolo potrebné zabezpečiť zakázanie mazania duplicít v celom indexovanom texte. Taktiež som odstránil stop slová vďaka funkciám, ktoré nám ElasticSearch ponúka. Čo sa týka tokenizera, ten som prestavil zo standardu na whitespace. Ako možné vylepšenie by som zvolil namiesto asciifoldingu filter icu\_folding, ktorý oproti asciifoldingu počíta aj s významom jednotlivých slov. Dokonca vie, že ak po sebe ide „c“ a „h“ tak je to „ch“. Ja som si vybral asciifolding pretože využitie icu\_folding, je podstatne drahšie aj keď presnejšie.

Nastavenie analyzátora a filtra:

"settings": {

"analysis": {

"analyzer": {

"my\_analyzer\_czech": {

"type": "czech",

"tokenizer": "whitespace",

"filter": [

"czech\_stop",

"lowercase",

"on\_same\_position"

]

}

},

"filter": {

"czech\_stop": {

"type": "stop",

"stopwords": [

"\_czech\_"

]

},

"on\_same\_position": {

"type": "unique",

"only\_on\_same\_position": true

}

}

}

}

# Scenáre

Každý scenár obsahuje inú query s využitím iných funkcií a parametrov. Dôvodom je vyskúšanie si viacerých prístupov a druhov vyhľadávanie v ElasticSearch.

## Prvý scenár

Prvý scenár (Viď. Príloha B) Query č.1) nám vyhľadáva záznamy, ktoré obsahujú v obsahu slovo „Komedie“, daný záznam nesmie mať dĺžku filmu medzi 10-100 minút a súčasne by mal obsahovať slovo „holka“ v nadpise.

Z vyhľadaných záznamov vieme usúdiť že analyzér nám pracuje správne, keďže ako prvý záznam nám vrátil tento:

"\_index" : "varga\_vi\_zadanie\_1",

"\_type" : "\_doc",

"\_id" : "1633",

"\_score" : 11.030911,

"\_source" : {

"film\_content" : [

"V hlavních rolích rozpustilé komedie "

],

"film\_name" : "Holky z kalendáře",

"film\_length" : 108

}

V zázname je vidno, že vyhľadávač nemal problém s veľkými a malými písmenami (komedie = Komedie) a taktiež prirovnal slovo „holka“ ku „Holky“.

## Druhý scenár

Tento scenár (Viď. Príloha B) Query č.2) nám vráti všetky filmy, ktoré obsahujú v jednom zo svojich tagov slovo „vražda“ a mal by obsahovať aj slovo „drogy“. Pri tejto query som si vyskúšal aj agregačné funkcie. Na konci dát nám zobrazí top 5 žánrov s počtom filmov v jednotlivých rokoch.

Orezaný výsledok pre pochopenie:

"aggregations" : {

"genre" : {

"doc\_count\_error\_upper\_bound" : 8,

"sum\_other\_doc\_count" : 752,

"buckets" : [

{

"key" : "Thriller",

"doc\_count" : 674,

"year" : {

"buckets" : [

{

"key" : 2000.0,

"doc\_count" : 40

},

{

"key" : 2001.0,

"doc\_count" : 26

},..................

]

}

"key" : "Horor",

"doc\_count" : 476,

.....................

}..............

## Tretí scenár

Tento scenár (Viď. Príloha B) Query č.3) hľadá v obsahu slovo „pes“ a súčasne obsah nesmie obsahovať „dvojice“. Aby som si vyskúšal ovplyvňovanie výsledkov, zadal som do query podmienku, že ak obsah obsahuje slovo „animovany“ tak sa boost nastavý na 1,5. Môžme to vidieť na zoradení výsledkov, prvý záznam má skóre 16.885044, no avšak ak by sme boost dali preč ten istý záznam má skóre len 13.894733.

"max\_score" : 16.885044,

"hits" : [

{

"\_index" : "varga\_vi\_zadanie\_1",

"\_type" : "\_doc",

"\_id" : "8715",

"\_score" : 16.885044,

"\_source" : {

"film\_content" : [

"Animovaný film o chlapci, který právě ztratil svoji matku a který má důvod ke starostem, když jeho pes onemocní."

],

"film\_name" : "Pes"

}

},

## Štvrtý scenár

Štvrtý scenár (Viď. Príloha B) Query č.4) skúša zmenu score, multi\_match a využitie query\_string-u. Dopyt hľadá záznamy, ktoré obsahujú v obsahu alebo názve „americky prezident“ s tým, že ak je zhoda v obsahu nastaví sa boost 2. Využíva aj phrase, ktoré zaručuje najlepšiu zhodu ak sa nájde zhoda vo viacerých poliach (porovnáva score). Taktiež je tam využitý slop, ktorý určuje vzdialenosť. Ďalšia časť query vyhľadáva či sa v hercoch alebo režiséroch nachádza „Barack Obama“ alebo „McDonald“. A nakoniec sa záznamy zoradia aj podľa priemeru, keďže výpočet score ešte prebieha podľa = sqrt(1.5 \* priemer).

## Piaty Scenát

Piaty scenár (Viď. Príloha B) Query č.5) vyhľadáva v obsahu „psa“ s tým že dané filmy musia mať premiéru v slovenských kinách v dátumoch od „2006-01-01“ do „2016-01-01“. Záznamy nesmú obsahovať herca „Ronja Arvilommi“. Za záznamami sa ešte zobrazia „tagy“ s počtom zhôd daných v našom vyhľadaní.

## Šiesty scenár

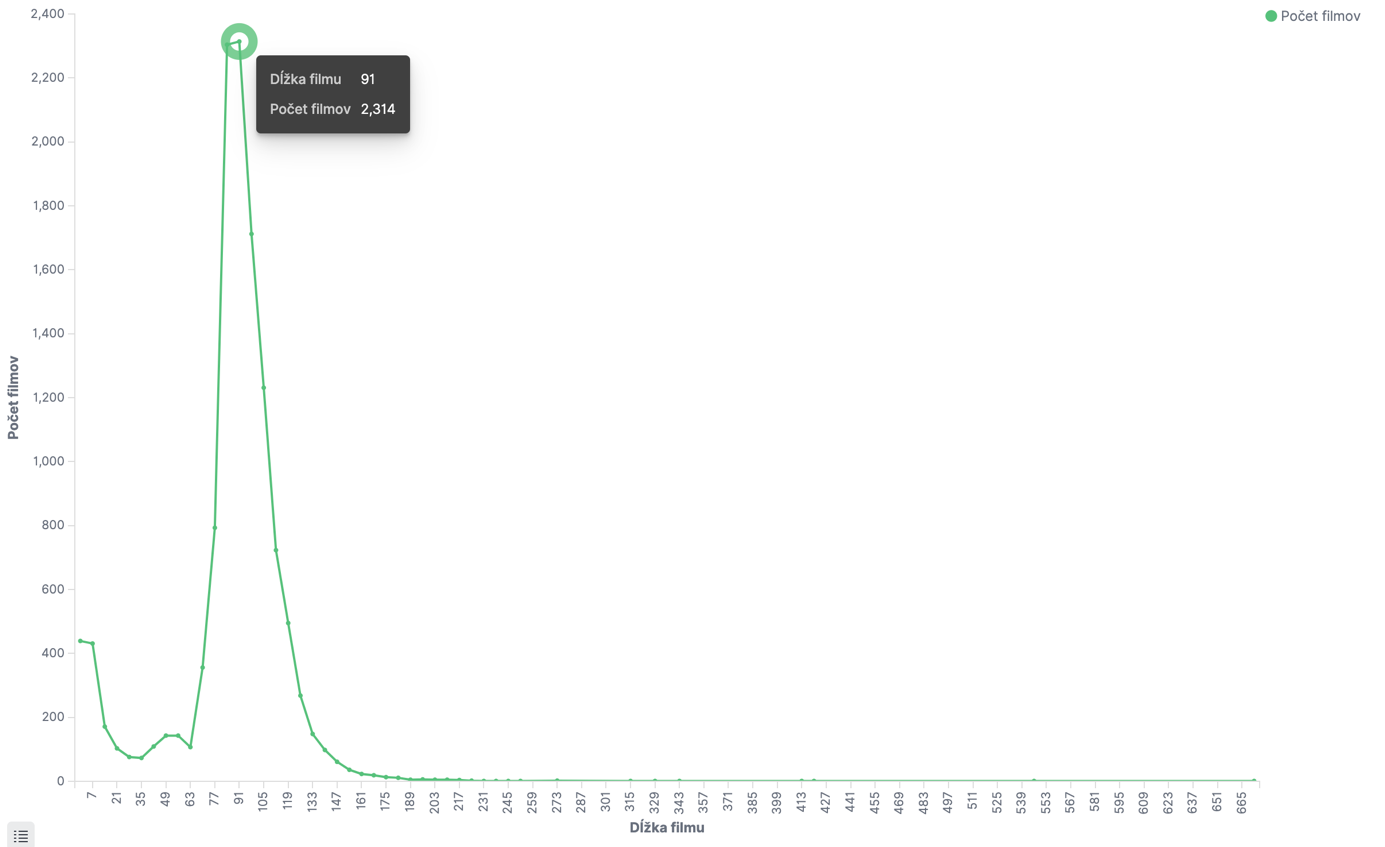
Šiesty scenár (Viď. Príloha B) Query č.6) vyhľadáva v hercoch jedno zo zadaných mien: „Angelina Jolie“, „Brad Pitt“, „George Clooney“, „Antonio Banderas“, „Jennifer Aniston“, „Courteney Cox“, „Robert Downey Jr.“ alebo „Chris Evans“. Z vyhľadaných záznamov sa následne vytvorí štatistika pre všetky roky a pre každý rok sa vypíše minimálna, maximálna a priemerná hodnota hodnotenia filmu.

# Vizualizácie

Vizualizácie boli vytvorené z vlastných crawlovaných dát. Na vytvorenie vizualizácií bola využitá Kibana.

## Vizualizácia č.1

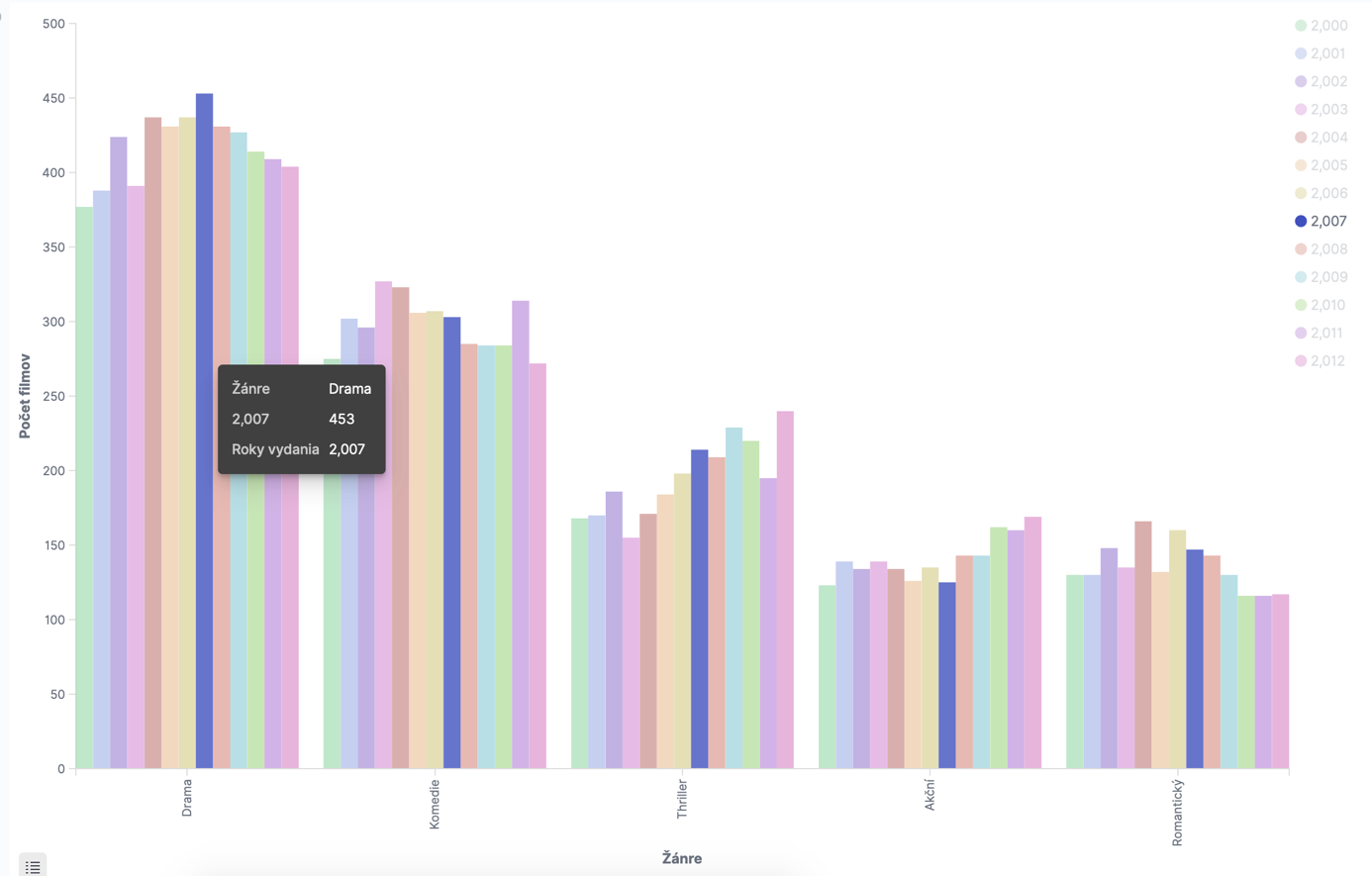
Na vizualizácií je vidieť koľko filmov má akú dĺžku. Zaujímavé je práve z tohoto grafu vyčítať, že najviac filmov (2314) má práve 91 minút.



Obrázok Vizualizácia 1

## Vizualizácia č.2

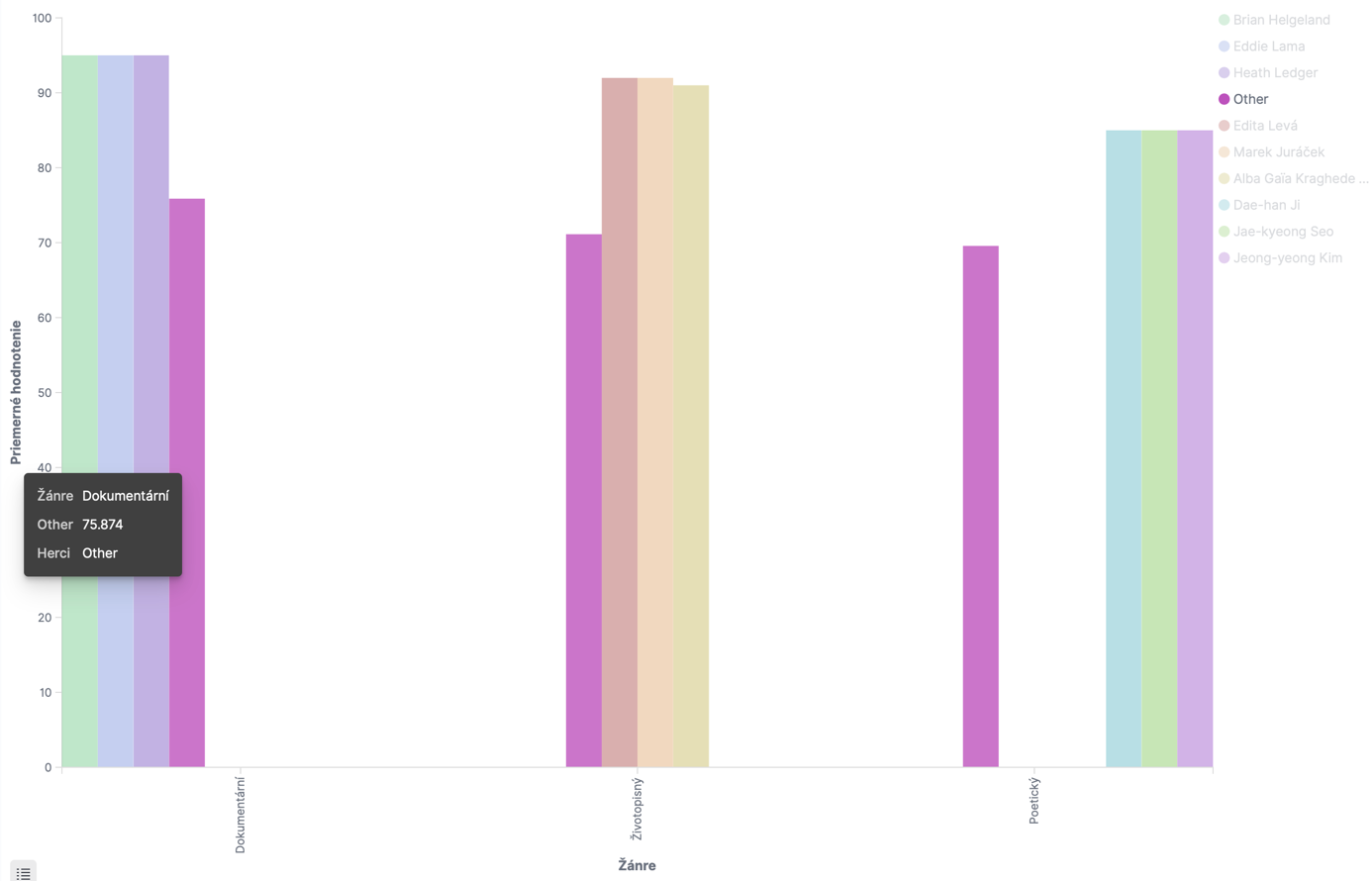
Táto vizualizácia zobrazuje top 5 žánrov podľa počtu filmov s tým, že pre každý žáner zobrazuje jednotlivé roky s počtom vydaných filmov obsahujúcich daný žáner. Z tohoto grafu je zaujímavé vyčítať, že v roku 2007 bolo najviac vydaných filmov v žánre „dráma“.



Obrázok Vizualizácia 2

## Vizualizácia č.3

Táto vizualizácia zobrazuje, ktoré sú top 3 žánre s najlepším priemerným hodnotením. Pre každý žáner zobrazí top 3 spisovateľov aj s ich priemerom. Taktiež pre každý žáner zobrazí aj priemer ostatných. Je zaujímavé vidieť, že priemer pre „Dokumentárne“ filmy je až 75,874.



Obrázok Vizualizácia 3

# Príloha A) Mapovanie

{

"settings": {

"analysis": {

"analyzer": {

"my\_analyzer\_czech": {

"type": "czech",

"tokenizer": "whitespace",

"filter": [

"czech\_stop",

"lowercase",

"on\_same\_position"

]

}

},

"filter": {

"czech\_stop": {

"type": "stop",

"stopwords": [

"\_czech\_"

]

},

"on\_same\_position": {

"type": "unique",

" only\_on\_same\_position": true

}

}

}

},

"mappings": {

"properties": {

"film\_actors": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_administration": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_average": {

"type": "long"

},

"film\_camera": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_content": {

"type": "text",

"analyzer": "my\_analyzer\_czech",

"search\_analyzer": "my\_analyzer\_czech",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_director": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_genre": {

"type": "text",

"analyzer": "my\_analyzer\_czech",

"search\_analyzer": "my\_analyzer\_czech",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_in\_cinema\_cz": {

"type": "date"

},

"film\_in\_cinema\_sk": {

"type": "date"

},

"film\_in\_cinema\_usa": {

"type": "date"

},

"film\_length": {

"type": "long"

},

"film\_music": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_name": {

"type": "text",

"analyzer": "my\_analyzer\_czech",

"search\_analyzer": "my\_analyzer\_czech",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_on\_bluray\_cz": {

"type": "date"

},

"film\_on\_bluray\_sk": {

"type": "date"

},

"film\_on\_bluray\_usa": {

"type": "date"

},

"film\_on\_dvd\_cz": {

"type": "date"

},

"film\_on\_dvd\_sk": {

"type": "date"

},

"film\_on\_dvd\_usa": {

"type": "date"

},

"film\_origin": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_scenario": {

"type": "text",

"analyzer": "standard",

"search\_analyzer": "standard",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_tags": {

"type": "text",

"fields": {

"keyword": {

"type": "keyword",

"ignore\_above": 256

}

}

},

"film\_year": {

"type": "long"

}

}

}

}

# Príloha B) Query scenárov

## Query č. 1

GET /\_search

{

"query": {

"bool": {

"must": {

"match": {

"film\_content": "Komedie"

}

},

"must\_not": {

"range": {

"film\_length": {

"gte": 10,

"lte": 100

}

}

},

"should": [

{

"match": {

"film\_name": "holka"

}

}

]

}

},

"\_source": [

"film\_name",

"film\_content",

"film\_length"

]

}

## Query č. 2

GET /\_search

{

"query": {

"bool": {

"must": [

{

"term": {

"film\_tags": "vražda"

}

}

],

"should": [

{

"term": {

"film\_tags": "drogy"

}

}

]

}

},

"aggs": {

"genre": {

"terms": {

"field": "film\_genre.keyword",

"order": {

"\_count": "desc"

},

"size": 5

},

"aggs": {

"year": {

"histogram": {

"field": "film\_year",

"interval": 1,

"min\_doc\_count": 1

}

}

}

}

},

"\_source": [

"film\_name",

"film\_tags",

"film\_year",

"film\_genre"

]

}

## Query č. 3

GET /\_search

{

"query": {

"bool": {

"must": [

{

"match": {

"film\_content": "pes"

}

}

],

"must\_not": [

{

"match": {

"film\_content": "dvojice"

}

}

],

"should": [

{

"match": {

"film\_content": {

"query": "animovany",

"boost": 1.5

}

}

}

]

}

},

"\_source": [

"film\_name",

"film\_content"

]

}

## Query č. 4

GET /\_search

{

"query": {

"function\_score": {

"query": {

"bool": {

"must": [

{

"multi\_match": {

"query": "americky prezident",

"fields": [

"film\_content^2",

"film\_name"

],

"slop": 2,

"type": "phrase"

}

},

{

"query\_string": {

"query": "(Barack Obama) OR (Mac McDonald)",

"fields": [

"film\_actors",

"film\_director"

]

}

}

]

}

},

"field\_value\_factor": {

"field": "film\_average",

"factor": 1.5,

"modifier": "sqrt",

"missing": 1

}

}

},

"\_source": [

"film\_name",

"film\_content",

"film\_actors",

"film\_director",

"film\_average"

]

}

## Query č. 5

GET /\_search

{

"query": {

"bool": {

"must": [

{

"match": {

"film\_content": "pes"

}

},

{

"range": {

"film\_in\_cinema\_sk": {

"gte": "2006-01-01",

"lte": "2016-01-01",

"boost": 2

}

}

}

],

"must\_not": [

{

"match\_phrase": {

"film\_actors.keyword": "Ronja Arvilommi"

}

}

],

"filter": [

{

"exists": {

"field": "film\_in\_cinema\_sk"

}

}

]

}

},

"aggs": {

"tags": {

"terms": {

"field": "film\_tags.keyword",

"order": {

"\_count": "desc"

},

"size": 5

}

}

},

"\_source": [

"film\_name",

"film\_content",

"film\_in\_cinema\_sk",

"film\_length"

]

}

## Query č. 6

GET /\_search

{

"query": {

"query\_string" : {

"query": "(Angelina Jolie) OR (Brad Pitt) OR (George Clooney) OR (Antonio Banderas) OR (Jennifer Aniston) OR (Courteney Cox) OR (Robert Downey Jr.) OR (Chris Evans)",

"fields": ["film\_actors"]

}

},

"aggs": {

"year": {

"histogram": {

"field": "film\_year",

"interval": 1,

"min\_doc\_count": 1

},

"aggs": {

"average": {

"avg": {

"field": "film\_average"

}

},

"maximum": {

"max": {

"field": "film\_average"

}

},

"minimum": {

"min": {

"field": "film\_average"

}

}

}

}

},

"size": 0

}

# Príloha C) Zoznam súborov

crawler\_full\_page.py - crawler ktorý stiahne celé HTML

crawler\_only\_data.py - crawler ktorý stiahne len potrebné dáta do JSON súboru

convert\_json\_for\_ElasticSearch.py - script ktorý vytvorený JSON prevedie na \_bulk formát

data\_for\_ElasticSearch\_example.json - časť dát ktoré sú už upravené na \_bulk

data\_for\_parser\_example.json - časť dát ktoré sú stiahnuté crawlerom (neupravené na \_bulk)

HTML\_page\_1.html - príklad HTML stránky odkiaľ sú dáta

VI\_2019\_Z1\_XVARGAJ4.pdf - dokumentácia zadania