1.1 VYHĽADÁVANIE A SŤAHOVANIE DATASETOV

Tento modul zabezpečuje vyhľadávanie relevantných datasetov na portáli a ich sťahovanie z úložiska Google Cloud Storage.

1.1.1 Vyhľadávanie datasetov a používateľov

Riešiteľ: Martin Žalondek, Helmut Posch, Richard Belan

Analýza

Modul vyhľadávanie datasetu umožňuje používateľovi vyhľadať dataset na základe jeho atribútov. Google v rámci technológie App Engine obsahuje nástroj Google Search API. Tento nástroj poskytuje priamočiary model na indexovanie, vyhľadávanie a zobrazenie dát. Je určený prioritne pre implementáciu fulltextového vyhľadávania. Možnosti, ktoré poskytuje sú tie, ktoré potrebujeme aj pre našu prácu, a preto sme sa rozhodli použiť práve tento mechanizmus.

Každý záznam musí byť definovaný ako dátový typ *document* a musí mať definované atribúty. Dokument je uložený do databázy ako súčasť vopred definovaného indexu. Všetky dokumenty v danom indexe je neskôr možné vyhľadať na základe ich ID alebo ostatných atribútov. Atribúty v dokumentoch môžu obsahovať tieto dátové typy:

- *Textové pole*: String s maxmálnou dĺžkou 1024^2 znakov
- *HTML pole*: String vo formáte HTML s maximálnou dĺžkou 1024^2 znakov
- *Atomické pole*: String s maximálnou dĺžkou 500 znakov
- *Číselné pole*: Float s hodnotou medzi -2,147,483,647 a 2,147,483,647
- *Dátum*: Pole na zobrazenie dátumu použitím Python tried - datetime.date alebo datetime.datetime
- *Geografické pole:* Bod na mape opísaný svojou zemepisnou šírkou a dĺžkou

Obsah textových a HTML polí je tokenizovaný. Reťazec je rozdelený na tokeny podľa toho, kde sa v ňom nachádzajú biele alebo špeciálne znaky (interpunkcia, mriežka a pod.). Index obsahuje záznam pre každý token čím umožní vyhľadávať kľúčové slová a frázy obsiahnuté iba v časti poľa atribútu. Napríklad vyhľadávací reťazec „berkley“ by umožnil vyhľadanie dokumentu s textovým atribútom „Berkley image dataset“.

Tagy obsiahnuté v HTML poliach nie sú tokenizované. Preto dokument, ktorý obsahuje v atribúte text „biggest collection of datasets“ bude vyhovovať vyhľadávaniu slova „collection“, avšak nie slova „strong“.

Atomické polia nie sú tokenizované. Dokument s hodnotou „bad weather“ v atomickom poli nebude možné vyhľadať pomocou reťazca „bad“, ale bude nutné vypísať jeho celú hodnotu „bad weather“.

Pravidlá podľa ktorých hodnoty atribútov tokenizované:

- Znak podčiarkovník (_) a ampersand (&) nerozdeľujú slová na tokeny

- Na obr. sú interpunkčné znaky, ktoré rozdeľujú slová na tokeny

Obrázok 1 Znaky rozdeľujúce slová na tokeny

- Všetky ostatné 7-bitové znaky okrem písmen a čísel (‘A-Z’, ‘a-z’, ‘0-9’) sú brané ako interpunkcia a rozdeľujú slová na tokeny
- Všetko ostatné je brané ako UTF-8 znak

Vyhľadávaciemu dopytu môžeme nastaviť tieto nastavenia:

- *Limit* – maximálny počet dokumentov, ktoré má dopyt vrátiť
- *Number_found_accuracy* – presnosť výsledku, ktorý vracia funkcia na zistenie počtu nájdených dokumentov (`SearchResults.number_found()`)
- *Offset* – pozícia dokumentu v databáze, ktorý nám vráti vyhľadávanie. Vhodné nastavovať ak máme viac výsledkov ako sa zmestí na jednu stránku. Na ďalšie stránky sa môže pokračovať od pozície stanovenej v atribúte offset a nemusia sa načítavať výsledky, ktoré uz pred tým načítané boli.
- *Cursor* – funguje podobne ako *offset*, avšak nepodporuje spätné stránkovanie, čo predstavuje nevýhodu pre náš projekt
- *Sort_options* – nastavenie kritérií podľa ktorých sa výsledky majú usporiadať

Tieto nastavenia určujú ktoré polia dokumentu sa budú nachádzať vo výsledkoch:

- *Ids_only* – ak je nastavený na *True*, metóda vráti iba atribút ID pre každý dokument
- *Returned_fields* – určuje ktoré atribúty dokumentov vráti metóda na vyhľadávanie
- *Returned_expressions* – každý dokument sa vypočíta hodnota výrazu. Napr. min, max, count.
- *Snippeted_fields* – zoznam textových polí dokumentu, pre ktoré sa vygeneruje *snippet* (útržok). Vhodné použiť ak je dĺžka textového poľa príliš veľká a my požadujeme iba časť z nej.

Nastavenie usporadúvania dokumentov je veľmi dôležitá súčasť vyhľadávacieho mechanizmu. Google Search API umožňuje nastaviť tieto atribúty pri usporadúvaní:

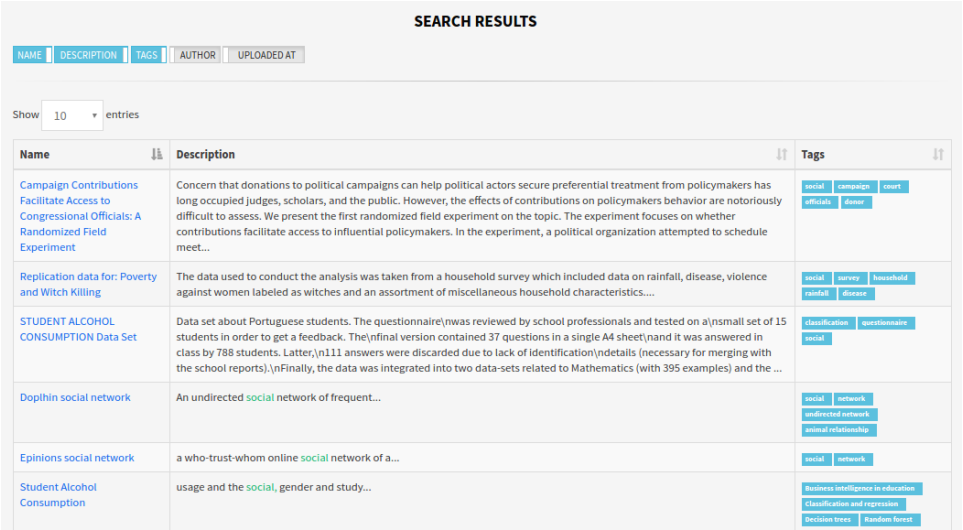
- *Expression* – výraz, ktorý sa vyhodnotí pri usporadúvaní nájdených dokumentov
- *Match_scorer* – usporiadanie dokumentov podľa frekvencie výskytu hľadaných slov v dokumente
- *Limit* – maximálny počet objektov, ktoré sa majú usporiadať. Maximálne 10 000 výsledkov

Implementácia vyhľadávania datasetov

V súčasnej verzii umožňujeme vyhľadávať datasety podľa názvu, autorov, citácií, tagov, popisu a dátumu nahratia. Keďže aktuálna verzia Search API nepodporuje vyhľadávanie podreťazcov, rozhodli sme sa implementovať vlastné riešenie. Pomocou metódy *tokenize_string()* je reťazec pred uložením rozdelený na všetky možné podreťazce. Tieto podreťazce sú následne uložené do dokumentovej databázy. Atribúty, ktoré evidujeme pre každý dataset sú:

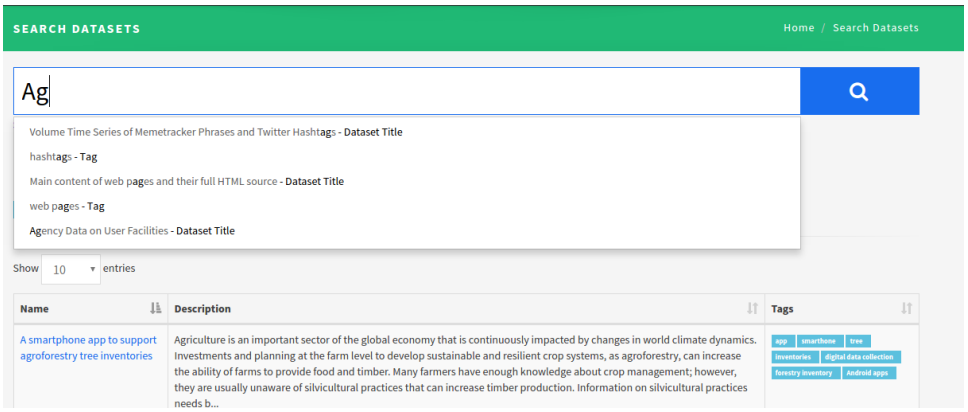
- *Autor*
- *Autor_tokenized*
- *Citation*
- *Description*
- *Id_datastore*
- *Tag*
- *Tag_tokenized*
- *Title*
- *Title_tokenized*
- *Uploaded*
- *Url_alias*

Ak sa hľadaný reťazec nachádza v popise datasetu, daný reťazec je farebne vyznačený vo výsledkoch vyhľadávania. Výsledky sú zoradené podľa relevancie. Používateľ si môže zvoliť atribúty, ktoré chce pre každý dataset zobraziť.



Obrázok ? Výsledky vyhľadávania

Vyhľadávanie datasetov zahŕňa aj automatické dopĺňanie možností počas zadávania vyhľadávacieho reťazca. Možnosti sú rozdelené na názov datasetu alebo tag. 200 milisekúnd po stlačení klávesy je odoslaná AJAX správa na server a ten následne vráti výsledok vyhľadávania, ktorý sa používateľovi zobrazí vo vysúvacom menu.

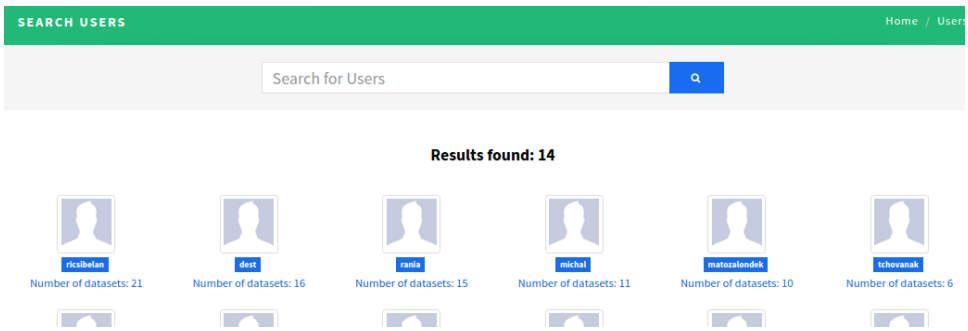


Obrázok ? Automatické dopĺňanie možností

Implementácia vyhľadávania používateľov

Vyhľadávanie používateľov je umožnené podľa ich používateľského mena. Výsledky sú zoradené podľa počtu nahraných datasetov od daného používateľa. Atribútu, ktoré evidujeme pre každého používateľa sú:

- *User_name*
- *Username_tokenized*
- *Profile_image*
- *First_name*
- *Last_name*
- *Datastore_id*
- *Dataset_counter*



Obrázok ? Vyhľadávanie používateľov

2

Zobrazenie najčastejšie použitých tagov

Možnosť zobrazenia najčastejšie používanych tagov môže uľahčit novému návštevníkovi zorientovať sa pri hľadaní datasetov. Údaje o počtoch jednotlivých tagov získavame pomocou možnosti *Faceted Search,* ktorú obsahuje Search API. Pre odľahčenie vyhľadávania sú záznamy o tagoch ukladané do Datastore tabuľky *FacetsTagsModel* spolu s časovou pečiatkou. Ak je časová pečiatka staršia ako 12 hodín, záznamy sa považujú za neaktuálne a pri najbližšom načítaní stránky s tagmi sú aktualizované.

Obrázok ? Zobrazene 20 najpoužívanejších tagov

Testovanie

Vzhľadom na to, že tento modul je zatiaľ len prototyp, vykonali sme na ňom iba jednoduché testovanie. Vložili sme doň 30 záznamov fiktívnych datasetov, zadali vyhľadávací dopyt na stránke a vypísané výsledky porovnali s očakávanými. Testovanie sa ukázalo ako úspešné. Otestovali sme aj funkcionalitu stránkovania vyhľadaných výsledkov , ktoré taktiež dopadlo úspešne.

Pri testovaní systému v lokálnom prostredí sme objavili problém, pretože uložené záznamy v databáze sa pri každom reštarte počítača vymažú. Ako riešenie tohto problému sme navrhli zálohovanie dát z databázy alebo zmenu priečinku, do ktorého sa dáta bežne ukladajú. Pri testovaní systému priamo na Google Cloud, sme tento problém neregistrovali.

1.1.2 Sťahovanie datasetu

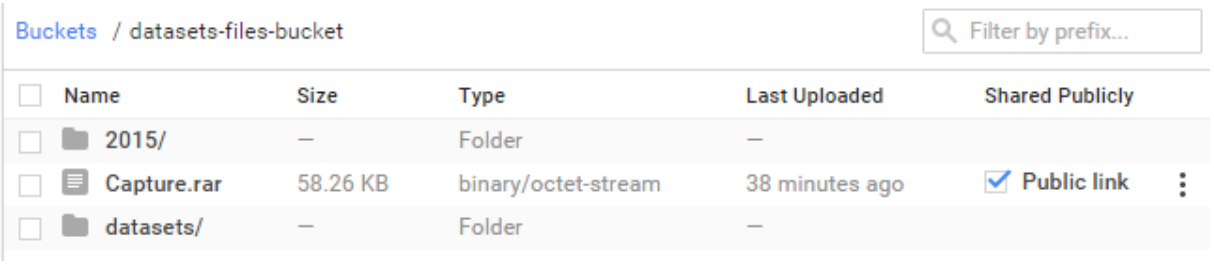
Riešiteľ: Michal Palatinus, Richard Belan, Rania Daabousová

Analýza

Modul Stiahnutie datasetu umožňuje používateľovi prostredníctvom webovej aplikácie stiahnuť vybraný dataset. Google Cloud Storage poskytuje návod na stiahnutie objektov z úložiska prostredníctvom GET požiadaviek. GET požiadavka sa vytvára a odosiela na strane klienta (webová aplikácia).

Riešenie

GET požiadavka musí špecifikovať objekt (v našom prípade dataset) a tzv. *bucket*, v ktorom je dataset uložený. Odpoveď na požiadavku vracia požadovaný objekt. V prípade, že vytvárame GET požiadavku na neexistujúci objekt, odpoveďou je *404 Not Found.* Aby bolo možné objekt stiahnuť, musí byť v Google Cloud Storage povolené verejné zdielanie zakliknutím políčka *Public link.*



Obrázok 3.: Povolenie verejného zdielania objektu prostredníctvom verejného odkazu.

```
function loadFile() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
    if (xhttp.readyState == 4 && xhttp.status == 200) {
        var obj = JSON.parse(xhttp.responseText);
        window.open(obj.mediaLink);
    }
}
xhttp.open("GET", "https://www.googleapis.com/storage/v1/"
        +"b/datasets-files-bucket/o/Capture.rar", true);
xhttp.send();
}
```

Obrázok 4.: Funkcia loadFile() na stiahnutie datasetu.

Testovanie

Momentálne je sťahovanie vo verzií prototypu. Objekt je staticky špecifikovaný svojou cestou v rámci úložiska.

Testovacie GET požiadavky majú nasledujúci tvar:

```
GET /storage/v1/b/datasets-files-bucket/o/cesta_k_datasetu HTTP/1.1
Host: www.googleapis.com
```

a bolo vykonané na objekte Capture.rar.

```
GET /storage/v1/b/datasets-files-bucket/o/Capture.rar HTTP/1.1
Host: www.googleapis.com
```

V nasledujúcom vývoji plánujeme pridať dynamické skladanie URL pre ľubovoľné datasety.

1.1.3 Sťahovanie datasetu v archíve

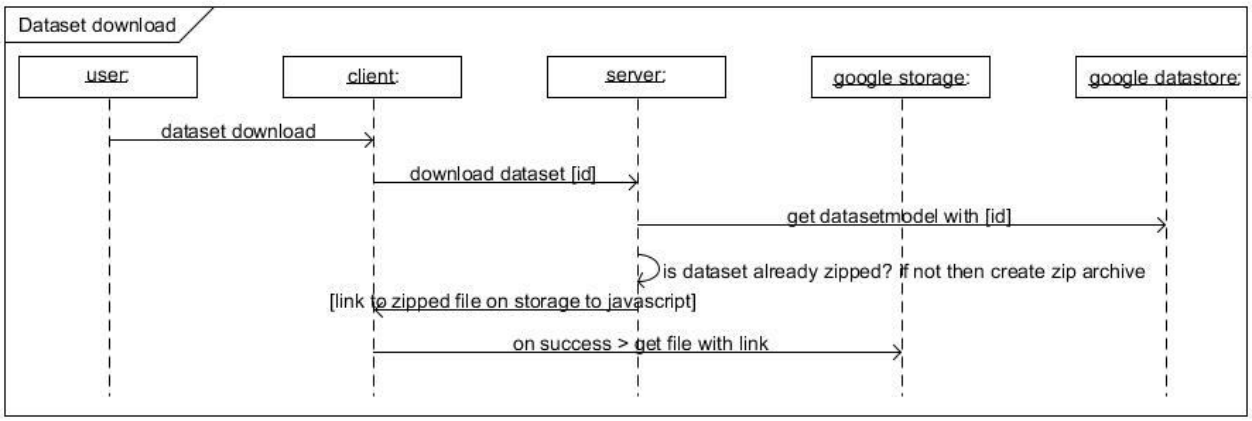
Riešiteľ :Tomáš Chovaňák, Richard Belan

Analýza

Pri datasetoch s väčším počtom súborov sme sa rozhodli poskytnúť možnosť stiahnutia datasetu ako zip archívu. Pričom archív sa má vytvárať vždy pri prvej požiadavke (niečo ako "lazy" prístup).

Riešenie

Api funkcia v module *download_app* a pohľade *DatasetArchiveDownload* umožňuje javascript klientovi pomocou odoslania ajax post požiadavky s id datasetu, ktorý má byť stiahnutý ako archív, získať zo servera link potrebný pre stiahnutie zip archívu. Logika sa odohráva na serveri, kde sa kontroluje či už daný dataset má vytvorený archív (ak áno tak sa len jednoducho vráti link). Ak ešte archív nie je vytvorený tak sa v Google cloud storage otvoria všetky súbory datasetu a vložia do archívu pomocou knižnice *zipfile.* Následne sa vytvorený archív uloží k danému datasetu v Google Cloud Storage a v modeli datasetu sa nastaví link na tento archív, ktorý sa vráti aj klientovi.



Obrázok 5 Sekvenčný diagram pre scenár stiahnutia archivovaného datasetu.

Testovanie

Ručne testované tri scenáre

- download datasetu, ktorý je už archivovaný.
- Download datasetu, ktorý ešte nebol archivovaný a pozostáva z viacerých malých súborov

- Download datasetu, ktorý ešte nebol archivovaný, ale pozostáva iba z jedného väčšieho súboru a nemal by teda byť archivovaný.