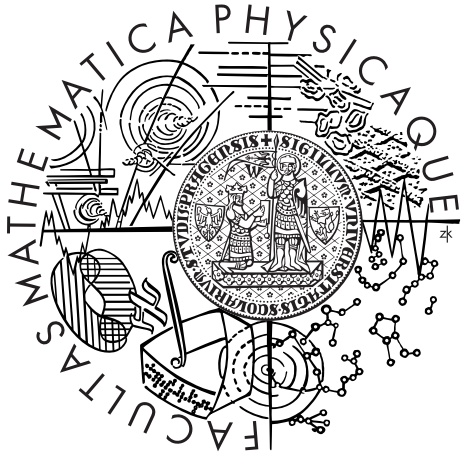
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Bc. Ondřej Odcházel

Automatické doporučování ilustračních snímků

Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: RNDr. Pavel Pecina, Ph.D.
Studijní program: Informatika
Studijní obor: Matematická lingvistika

Praha 2014

děkuji máje, že se nebojí létat

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne ..... Podpis autora

Název práce: Automatické doporučování ilustračních snímků

Autor: Bc. Ondřej Odcházel

Katedra: Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: RNDr. Pavel Pecina, Ph.D., Ústav formální a aplikované lingvistiky

Abstrakt:

Klíčová slova: vyhledávání obrazových informací

Title: Automatic suggestion of illustrative images

Author: Bc. Ondřej Odcházel

Department: Institute of Formal and Applied Linguistics

Supervisor: RNDr. Pavel Pecina, Ph.D., Institute of Formal and Applied Linguistics

Abstract:

Keywords: information retrieval, image retrieval

Obsah

Úvod (zatím opsané zadání)

Většina zpravodajských serverů často opatřuje publikované články tzv. ilustračními snímky, jejichž úkolem je vizuálně dokreslovat obsah článku a upoutat na něj čtenářovu pozornost. Ilustrační snímky většinou pocházejí z rozsáhlých fotografických databází, jsou vybírány autory článku a s obsahem článku souvisejí jen relativně volně. Výběr ilustračních snímků probíhá nejčastěji na základě porovnávání klíčových slov specifikovaných autorem textu a popisků, kterými jsou obrázky v databázi opatřeny (typicky svými autory).

Proces výběru ilustračních snímků (dotazování ve fotografické databázi) je obtížný jednak pro samotný vyhledávací systém (hledání relevantních fotografií na základě uživatelských dotazů), jednak pro autory, kteří musí dotazy vytvářet. Konstrukce dotazů spočívá v několika krocích: uživatel nejdříve musí identifikovat ústřední téma (či témata) článku, které chce ilustrovat vhodnou fotografií, a ta potom popsat vhodnými klíčovými slovy, zvolit a zkombinovat je tak, aby vedla k nalezení vhodného obrázku. Tento proces by mohl být zjednodušen tím, že konstrukce dotazů pro vyhledávání bude prováděna automaticky pouze na základě textu článku.

Cílem diplomové práce je navržení a implementace komfortní webové aplikace pro automatické navrhování ilustračních snímků na základě textu článku, bez nutnosti explicitně konstruovat vyhledávací dotazy. Součástí práce bude i uživatelská evaluace celého systému. Pro experimenty bude použita kolekce ilustračních snímků od společnosti Profimedia.

1. Teorie: Jak najít vhodné obrázky

1.1 Popis datové sady

Datová sada poskytnutá firmou Profimedia obsahuje 20 014 394 oanotovaných obrázků ve formě CSV souboru "profi-text-cleaned.csv". CSV obsahuje sloupce "locator","title","description","keywords". Sloupec locator obsahuje ID obrázku v databázi Profimedia. Sloupec description je prázdný. Sloupce title a keywords obsahují řetězce anglických slov popi sujících obrázek a oddělených mezerou.

Příklad jednoho řádku souboru profi-text-cleaned.csv:

```
"0000000980","hradec kings holy ghost cathedral","",
    "outdoors nobody urban scenes architecture houses
    towers czech czech republic europe buildings build
    history historical churches church fronts holy ghost
    cathedral spirit ceska republika cathedrals sv hradec
    kralove"^M
```

Na příkladu je vidět, že data obsahují fráze jako "holy ghost cathedral", tyto fráze však nejsou strojově čitelně vyznačené. Dalším problémem je špatný překlad dat do angličtiny. Fráze "hradec kings" vznikla evidentně doslovným překladem názvu "hradec králové".

Důležitým aspektem dat je jejich nepodobnost běžnému novinovému textu. Anotované texty neobsahují většinu nejfrekventovanějších anglických slov.

1.2 Teoretické cíle

V teoretické části je hlavním cílem práce nalézt nejvhodnější metodu extrakce klíčových slov textu. Tato klíčová slova pak budou použita při vyhledávání ilustračních obráyků v databázi Profimedie. Celá práce, pokud nebude uvedeno jinak, označuje za slova stemy vstupních slov. Jako stemmer se využívá ??? stemmer.

Vstupní text tedy nejprve rozdělíme na slova. Čísla a interpunkce nás v této úloze nezajímají, jelikož se v datech nenachází. Ze slov pak získáme stemy. Vstupem algoritmu pro nalezení klíčových slov tedy bude množina vhodných stemů. Ke každému stemu si ještě uložíme jednu jeho nestemovou variantu, kterpu pak můžeme zobrazit uživateli.

Nyní můžeme použít některý z algoritmů na extrakci klíčových slov uvedených v další kapitole.

1.3 Rešerše, vhodné algoritmy

Algoritmy na extrakci klíčových slov lze rozdělit do dvou kategorií. V jedné máme k dispozici korpus podobných dokumentů, druhá kategorie tento korpus ke své práci nepotřebuje.

1.3.1 TF-IDF

Technika TF-IDF je známý algoritmus na měření významnosti slov v textu. Využívá korpusu dokumentů D a dvou složek.

$$TFIDF(t, d, n, N) = TF(t, d) \times IDF(n, N) \tag{1.1}$$

Složka TF znamená $TERM\ FREQUENCY$ a pokud t je slovo a $d \in D$ je dokument, je TF

$$TF(t, d) = \begin{cases} 1 & \text{pokud } t \in d \\ 0 & \text{jinak} \end{cases} \tag{1.2}$$

$$TF(t, d) = \sum_{slovo \in d} \begin{matrix} 1 & \text{pokud } slovo = t \\ 0 & \text{jinak} \end{matrix} \tag{1.3}$$

Jedná se tedy o frekvenci slova (stemu) v dokumentu.

Složka IDF, tedy $INVERSE\ DOCUMENT\ FREQUENCY$ vyjadřuje, jak moc daný termín popisuje dokument. Pokud je N počet všech dokumentů v D, tedy $N = |D|$ a n je počet dokumentů, ve kterých se vyskytuje slovo t, je IDF tohoto slova

$$IDF(n, N) = \log\left(\frac{N}{n}\right) \tag{1.4}$$

$$IDF(n, N) = \log\left(\frac{N - n}{n}\right) \tag{1.5}$$

Čím je tedy slovo v korpusu častější, tím více se s logaritmem snižuje jeho informační hodnota. Slova, která jsou velmi běžná většinou klíčovými slovy.

Výsledný vzorec pak jde shrnout jako:

$$TFIDF(t, d, n, N) = \left(\sum_{slovo \in d} \begin{matrix} 1 & \text{pokud } slovo = t \\ 0 & \text{jinak} \end{matrix} \right) \times \log\left(\frac{N - n}{n}\right) \tag{1.6}$$

Problémem tohoto algoritmu pro je odlišný charakter korpusu a vstupních dat. Vstupní data jsou typicky novinový článek. Pokud bychom jako korpus použily anotované obrázky, získáme špatné výsledky. Běžná anglická slova, jako "the", nebo "a" se v takovém korpusu vyskytují velmi zřídka, jejich IDF tedy bude vysoká. Naopak TF v běžném novinovém textu je vysoké. Takovýto korpus nám pak označuje jako klíčová slova běžná anglická slova.

1.3.2 Extrakce bez korpusu

Dalším druhy algoritmů ke své práci korpus nepotřebují a pracují pouze se vstupním textem.

1.4 Řešení: jaké algoritmy zvoleny, získání tréninkových dat

Jako nejvhodnější řešení byl nakonec zvolen TF-IDF algoritmus. Jako kandidáti jsou odfiltrována slova, která se nenacházejí v datech Profimedie. Jako korpus k měření IDF byla použita data článků z Wikipedie.

Pro rychlé testovací účely bylo oanotováno pár článků z anglických wikinews. V každém článku jsem označil pět klíčových slov. Porovnávání algoritmů na extrakci klíčových slov pak vzalo pět nejpravděpodobnějších klíčových slov podle algoritmu a porovnalo v kolika procentech se algoritmus trefil s anotací.

Zbývá: Zkusit otestovat další metody. Překlad do češtiny.

1.5 Evaluace výsledků

Výsledkem práce by mělo být rozhraní pro anotaci obrázků využívající algoritmus na hledání klíčových slov v textu.

Tento algoritmus se dá uživatelsky testovat několika způsoby. Uživatel vidí text a několik (cca 5) vrácených obrázků algoritmem. Uživatel vybere množinu relevantních obrázků. Další možností je mezi 5 vrácených obráyků vložit jeden náhodný. Úkolem anotátora je pak vybrat ten náhodně vybraný. Přesnost algoritmu pak jde měřit pomocí toho, kolikrát se anotátor trefí do špatného obrázku (potřeba zdrojový článek)

2. Praktická část: Implementace moderní webové aplikace

Práce je z velké části implementační. Snažil jsem se tedy vytvořit moderní webovou aplikaci s využitím co nejvíce frontendových novinek v novém standardu HTML5.

2.1 Databáze: jak uložit 20M metadat obrázků

Jak uložit takové množství dat, aby se dalo rychle vyhledávat. Škálovatelnost. Dostupnost knihoven pro práci s databázema. Proč jsem si vybral nakonec ES. SQL vs NoSQL.

2.2 Backend a úprava dat: Komunikace s databází, implementace algoritmů

Proč jsem zkoušel golang a proč jsem nakonec použil Ruby a Ruby on Rails. Základní popis MVC frameworku. Dostupnost knihoven pro získání stemů a prací s databází.

2.3 Frontend: AJAXová aplikace na zobrazování obrázků

Jaké jsou dnešní možnosti vývoje frontendu. Single page aplikace. Možnosti moderních prohlížečů. JavaScriptové knihovny. Proč to nedělám v jQuery, ale používám Google Closure. Google Closure Library, Templates, Compiler.

Návrh rozhraní bez jediného tlačítka. Responzivní webdesign.

2.4 Anotační rozhraní

Jak lze v Ruby on Rails vyrobit jednoduše anotační rozhraní s uživateli a s ukládáním do databáze.

2.5 Návod k použití

Popis prvků. Screenshoty aplikace.

2.6 Preklad

Dva postupy jak pouzit anglicka data v jinych jazycich. Bud je mozne prelozit vzdy zadany cesky dotaz do ciloveho jazyka. Nebo je mozne prelozit vsechna data

u fotek. Pak je nutne pouzit pro kazdy jazyk nejaky lemmatizer, nebo stemmer. Pro cestinu jsme nakonec zvolili druhou variantu.

Jak prelozit 20 milionu popisku? Jednou moznosti je preklad slovo od slova. Pouzit pouze slovnik. Preklad pomoci google je drahy. Stalo to zhruba 1300Kc. Preklad celych frazi by byl lepsi (automaticky objevi fraze), ale pomoci google velmi drahy. Rozchodil jsem tedy prekladovy nastroj Moses s modelem prilozenym ve verzi 2.1 (http://www.statmt.org/moses/RELEASE-2.1/models/en-cs/model/). Po nekolikahodinovem nacitani se model nacetl (i kdyz mam SSD disk). Preklady v prolozenem modelu jsou velmi pomale (jeden segment trva priblizne 3s). Preklad neni idealni a mam spustu —UNK slov.

2.7 Poznamky

Pouziji data z wiki dumpu.

Cesky dump: http://dumps.wikimedia.org/cswiki/20140612/ a cswiki-20140612-pages-articles-multistream.xml.bz2 /data/wiki$_{d}ump_{c}s.xmlrakees : extract_{w}ords_{f}rom_{w}iki[cs]e$
$60MBtextucestinaobsahujenazacatkuhodnekratkychvygenerovanychclankutypu >$
$1.leden <$

Anglicky dump: torrent z piratebay http://thepiratebay.se/torrent/8114722/Wikipedia$_{2}$013
$extract_{w}ords_{f}rom_{w}iki[en]extrakcedoid10000 > 70MBtextu$

application.rb obsahuje konstanty aplikace

vyrobim nejcastejsi trigramy pro cestinu a anglictinu z wiki dat pomoci: rake es:extract$_{m}ost_{f}requent_{t}rigrams$

Language data jsem stahnul z: http://dumps.wikimedia.org/cswiki/latest/

Ceska data jsou z: /net/seznamdata/profiset/profi-text-cleaned.csv

prekladova data: moses - http://www.statmt.org/moses/RELEASE-2.1/binaries/macosx-mavericks/bin/

prekladovy model: http://www.statmt.org/moses/RELEASE-2.1/models/en-cs/

3. Evaluace výsledků

3.1 Metodika

Jak budu měřit. Porovnání naivního a nejlepšího algoritmu. Nejlepší algoritmus vybrán pomocí oanotovaných klíčových slov ve Wikinews článcích. Pak anotátoři se budou snažit najít nejméně vhodný obrázek z nabízených hodnot.

Evaluace jenom pro angličtinu, nebo i pro češtinu?

3.2 Výsledky

Grafy, tabulky.

3.3 Možná zlepšení

Jak bychom kmohli mít lepší data a co omezuje použitý algoritmus.

Závěr

Seznam použité literatury

Seznam tabulek

Seznam použitých zkratek

Přílohy