Najčastejšie chyby/výsledky sú uvedené červenou farbou

Zadanie ku skúške z predmetu Vyhľadávanie informácií, rok 2012

Dokumenty

Číslo	**Text Dokumentu** (linky sú podčiarknuté, na konci je číslo dokumentu, kde ukazujú)	Číslo	**Text Dokumentu** (linky sú podčiarknuté, na konci je číslo dokumentu, kde ukazujú)
1	Zomrel československý exprezident a dramatik Václav Havel Zomrel človek, ktorý sa o pád komunistického režimu v Československu zaslúžil zrejme najviac. PRAHA, BRATISLAVA. Vo veku 75 rokov zomrel dnes Václav Havel. Novoročný prejav 1990 (2) štvrtok 18. 12. 2011, SME © Petit Press, a.s.	2	Novoroční projev prezidenta ČSSR Václava Havla Praha, Pražský hrad, 1. ledna 1990 Milí spoluobčané, čtyřicet let jste v tento den slyšeli z úst mých předchůdců v různých obměnách totéž: jak naše země vzkvétá, kolik dalších miliónů tun oceli jsme vyrobili, jak jsme všichni šťastni, jak věříme své vládě ... Předpokládám, že jste mne nenavrhli do tohoto úřadu proto, abych vám i já lhal. Naše země nevzkvétá. Velký tvůrčí a duchovní potenciál našich národů není smysluplně využit.
3	Václav Havel Zoznam správ: • Zomrel Václav Havel (1) [18.12.2011] • Novoročný prejav 1990 (2) [1.1.1990] • Hríb o Havlovi (4) [18.12.2011] narozen: 5. 10. 1936 v Praze zemřel: 18. 12. 2011 Hrádeček otec: ing. Václav M. Havel, architekt matka: Božena Havlová, roz. Vavrečková bratr: Ivan M. Havel (1938), vědec	4	Václav Havel .štefan Hríb 18. december 2011 Zomrel človek, ktorý zmenil môj život. Pretože si ctil pravdu viac ako väčšina. Je tu teraz smutno. Pán Václav Havel, pán prezident, ďakujem Vám. Copyright © 2009 - 2012 Vydavateľstvo W Press a.s., Partizánska 2, 811 03 Bratislava, Slovensko / ISSN: 1336-653X

1. Sťahovače (3b)

a) Aká je najlepšia stratégia sťahovania? (0,5b)

b) Ako sa definujú obmedzenia pre sťahovače a aké poznáme obmedzenia? (0,5b)
nespomenutie MIME typov (neuberali sa body)

c) Čo je FocusedCrawler a ako funguje? (1b)
Nespomenutie využitia head requestu
častá zlá odpoveď: sťahovanie dynamických stránok bez rovnakého obsahu – toto platí aj pre bežný stahovač že nechceme to isté.

d) Nakreslite orientovaný graf liniek medzi dokumenatmi a definujte v akom poradí sa stiahnu dokumenty pri vyhľadávaní do šírky a v akom poradí do hĺbky, keď začneme od dokumentu 3 a v rámci stránky sú linky objavené v poradí, akom sa nachádzajú v texte dokumentu. (1b)

2. Textové operácie (5b)

a) Tokenizujte všetky číselné informácie v dokumentoch 1 a 2 pomocou inteligentného tokenizátora. (1b)
Chýbajúce dátumy a textové čísla *čtyřicet* a *miliónů*

b) Lematizujte text správy v dokumente 4 (*Zomrel* ...). (1b)
je = byť, vám = vy, môj = *môj* (privlastňovacie zámeno) nie *ja* (osobné zámeno)

c) Stemujte prvú vetu z textu správy v dokumente 4 (*Zomrel .. ... život.*). (1b)
zomrel = *zomr*, môže byť aj odrezanie predpony, ale pri *mr* bude rovnaký stem pre veľa slov.

d) Tokenizujte posledné 3 riadky dokumentu 3, tak ako by ste mali inteligentný tokenizátor. Vezmite do úvahy rozdiel medzi tokenmi a termami. (1,5b)
Mená osôb ako treba ako jeden token – pre indexovanie potom ale treba normalizovať
e) Tokenizujte posledný riadok dokumentu 3 pomocou whitespace tokenizátora. (0,5b)
Správne: [bratr:] [Ivan] [M.] [Havel] [(1938),] [vědec]
Ide len o jeden typ tokenu nemá zmysel písať word a podobne, pričom word to často nie je.

3. *Indexovanie, váhovanie a podobnosť (8b)*

a) Vytvorte jednoduchý invertovaný index vyššie uvedených dokumentov. Berte do úvahy iba podčiarknuté slová (slová v linkách) a slová v nadpisoch dokumentov, ostatné vynechajte. Slová v rôznych tvaroch berte, akoby boli rovnaké. Indexujte tieto slová aj na miestach, kde nie sú podčiarknuté alebo nie sú v nadpise (napr. *Hríb* v dokumente 4) (1b)
b) Vytvorte invertovaný index, kde vezmete do úvahy aj počet výskytu (frekvenciu) termov v dokumente, za rovnakých podmienok ako v úlohe a). Anchor text (text liniek) priraďte aj tým dokumentom, na ktoré ukazuje, pričom termom z anchor textu dajte pre odkazované dokumenty dvojnásobnú váhu. Váhy nemusíte normalizovať. (3b)
c) Vypočítajte váhu termu *Havel* pomocou miery tf-idf (term frequency–inverse document frequency) v dokumentoch 1 a 2. Do úvahy berte celú kolekciu dokumentov (dokumenty 1 až 4) a lematizáciu. (2b)
výsledok je 0 lebo term Havel sa nachádza v každom dokumente kolekcie
d) Vypočítajte kosínusovú mieru medzi dopytom *Václav Havel* a dokumentmi 1, 3. Použite euklidovskú normalizáciu. Kolekciu tvoria iba dokumenty 1 a 3, uvažujte s lematizáciou a ignorujte číselné termy. (2b)

4. *Usporiadanie (5b)*

a) Akým spôsobom je možné kombinovať usporiadania, napr. kosínusovú mieru a PageRank? (1b)
b) Opíšte vlastnými slovami princíp PageRank (1b)
c) Napíšte vzťah pre výpočet PageRank pomocou Google Matice.(1b)
d) Určte Google Maticu a spravte prvú iteráciu. Damping factor je 0,6. Pravdepodobnosť že používateľ odskočí na ľubovoľnú stránku z dangling nódu a personalizačný vektor je (1/4, 1/4, 1/4, 1/4). (2b)

5. *Extrakcia informácií (6b)*

a) Akých je 5 základných úloh extrakcie informácií? (Definované konferenciami MUC) (1b)
Lepšie je po slovensky vysvetliť každú úlohu pár slovami, ako písať zle naspamäť anglické názvy.
Úlohy extrakcie nie sú tokenizácia, indexovanie, lematizácia, a pod. aj keď tieto techniky môžem pri extrakcii využiť.
b) Identifikujte názvoslovné entity v dokumente 1 a 3 a definujte ich typ. (1b)
Osoby, mestá, krajina, firma = Petit Press a.s., médium = SME
c) Opíšte, akým spôsobom sa dajú extrahovať mená osôb z textu a extrahujte ich zo všetkých dokumentov. (1b)
Neextrahované osoby z dokumentov
d) Preveďte všetky úlohy extrakcie informácií na dokumente 1. (3b)
NE: pozri b); malo by byť aj 75 rokov (vek) ale môže byť aj ako TE
CO, Aliasy, referencie: Václav Havel – človek, može byť asi aj exprezident
TE, Vlastnosti: Václav Havel - exprezident, dramatik; exprezident – československý
TR, sú to relácie NE (nie TE) v učebnom texte je chyba: Václav Havel = 75 rokov (vek); Václav Havel - exprezident (môže to byť aj TE ale skôr tu)
ST, Udalosť: zomrel (kto: Havel, kedy: dnes – dá sa odvodiť dátum); mohlo by byť ešte dátum a rok novoročného prejavu ale nehodnotené.
Niektorí aplikovali na jednotlivé kategórie slovné druhy, toto nie je úplne zle, lebo to má súvislosť (podstatné mená sú NE, CO zámená, slovesá sú ST, prídavné mená TE) ale nedá sa automaticky aplikovať.
☺ funny: TR: Václav Havel – československý človek

6. *Regulárne výrazy (5b)*

a) Na aké úlohy sa dajú v oblasti vyhľadávania informácií (information retrieval) použiť regulárne výrazy (regex)? (1b)
b) Napíšte regexy na vyhľadanie sídel (miest a dedín), firiem, PSČ, dátumov, ľudí a rokov v uvedených dokumentoch tak, aby boli aj všeobecnejšie použiteľné. Stačí definovať 3 regexy, musia správne fungovať minimálne na uvedených textoch. Je potrebné napísať regex na dátumy. Regexy vysvetlite, uveďte ktoré časti

čo extrahujú. Uveďte ktoré entity z uvedených textov extrahujú. (4b)
Nevysvetlené regexy, neuvedené entity
Väčšina regexov napísaných pre roky extrahovala aj číslo 1336 z ISSN
regex pre dátum nevedel extrahovať mesiace definované slovom.
regex pre meno osoby bez stredného mena alebo strednej iniciály
extrakcia sídel pomocou regexu (v|pri|na ….), ktorý nenájde nič v textoch
Poznámka: ruské PSČ nás tento rok nezaujíma

7. Hodnotenie (5b)

a) Aké sú základné miery hodnotenia systémov pre vyhľadávanie informácií (IR - Information Retrieval)? (1b)
b) Napíšte ich vzorce. (1b)
c) Definujte, aké dokumenty vráti dopyt *prezident Havel* bez a s lematizáciou. Vypočítajte základné miery hodnotenia pre IR systém, ktorý pre dopyt *prezident Havel* (s a bez lematizácie) vráti dokumenty 2, 3. (3b)

8. Sémantický web (5b)

a) Opíšte vlastnými slovami, čo je sémantický web a aké sú jeho ciele. Opíšte štandard RDF slovne čo je jeho podstatou, aj na konkrétnom príklade (1b)
Chýbal konkrétny príklad. Bolo tiež treba spomenúť že je založený na trojiciach: subjekt, predikát, objekt
XML je len forma zápisu, podstatné sú trojice.
b) Z extrakcie informácií dostaneme jednoduché objekty typov ako People (ľudia) a ich vzťahy. Vytvorte graf inštancií týchto objektov získaných v úlohe 5b) z dokumentu 3. Tieto inštancie zároveň majú vlastnosť *title*, kde je uvedený textový reťazec zistený z dokumentu po lematizácii. Inštancie typu People obsahujú vlastnosti *firstname*, *lastname*. (2b)
c) Napíšte SPARQL dopyt na získanie všetkých inštancií typu People. Výstupom je vlastnosť *firstname* týchto inštancií. Napíšte aj výsledok, ktorý vráti. (2b)
Ako môžete mať query a nemať úlohu b)? Znovu použitý geo prefix z minulých testov ...
Nebol uvedený výsledok ktorý vráti SPARQL dopyt

9. Softvérové knižnice a systémy (3b)

a) Opíšte na čo je možné použiť knižnice Lucene, Nutch a Tika. Aké sú ich základné vlastnosti? (1b)
Lucene nie je vyhľadávač. Nemá užívateľské rozhranie. Je to knižnica ktorá umožňuje analýzu textu (tokeny, termy), indexovanie a vyhľadávanie. Lucene nie je sám osebe škálovateľný. Iba v kombinácii s Hadoop pomocou Nutch. Ani Nutch nie je škálovateľný ak nebeží na Hadoop.
b) Opíšte, aké vlastnosti a časti musí obsahovať systém na vyhľadávanie informácií o ľuďoch v slovenskom webe. Opíšte, pomocou akých softvérových knižníc je možné takýto systém vytvoriť a čo je nutné doprogramovať. (2b)
spomenutie využitia info zo sociálnych sietí je fajn
Jedna z odpovedí: Beagle[1]? Keď sa učíte alebo robíte ťaháky zo starých testov treba tomu rozumieť mám chuť za takéto veci dávať mínus body .....

10. Vyhľadávanie informácií na internete a MapReduce (5b)

a) Uveďte vlastnosti vyhľadávača Google, ktorými sa už v roku 1998 odlíšil od dovtedy známych systémov pre vyhľadávanie na internete. (1b)
PageRank, využitie anchor textov liniek
škálovateľná architektúra založená na PC (Podľa Moorovho pravidla)
oddelenie sponzorovaných odkazov od výsledkov vyhľadávania.
b) V čom sa líši vyhľadávanie na internete od vyhľadávania napríklad v knižniciach (2b)
dôveryhodnosť informácie (každý môže napísať hocičo) a teda treba riešiť aj inú ako textovú relevanciu.
Dôležité je aj to čo iný hovoria o informácii (linky, anchor texty) a nie len to čo autor tvrdí (samotná info)
c) Napíšte aspoň dva systémy postavené nad Hadoop a opíšte na čo slúžia. (1b)
Malo byť Nutch, Hive, Pig, HBase, Mahout... a popis (Lucene, Cassandra je zlá odpoveď)
Zlé odpovede: Google a podobne. Ešte Yahoo! By sa dalo uznať aj keď system sa nemyslela priamo firma ale niečo čo sad á použiť.

[1] http://en.wikipedia.org/wiki/Beagle_(software)

d) Opíšte princíp MapReduce a uveďte aspoň dve jeho výhody. (1b)
Princíp: každá úloha ako Map a Reduce metódy kde vstupom a výstupom sú asociatívne polia. Map alebo sekvencia Map úloh rieši problém nad konkrétnou časťou dát. Reduce kombinuje, spracúva a triedi výsledky z viacerých úloh (musí kopírovať dáta, výstup z Map spracovaný pomocou Reduce by mal byť čo najmenší). Zložitejšia úloha je riešená pomocou sequencie Map a Reduce úloh. Úloha (program) ide k dátam nie opračne ako pri niektorých úlohách.
Výhody: škálovateľnosť; riešený failover; Nie je nutné aby programátor rozumel distribuovanému systému, stačí keď problém naprogramuje ako Map a Reduce metódy; Možnosť ladiť program lokálne. Vhodná architektúra pre spracovanie rozsiahlych dát.