VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

PŘEDZPRACOVÁNÍ A TRANSFORMACE TEXTOVÝCH KOLEKCÍ DAT

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE VIKTOR MARUNA
AUTHOR

BRNO 2013

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

PŘEDZPRACOVÁNÍ A TRANSFORMACE TEXTOVÝCH KOLEKCÍ DAT

PREPROCESSING AND TRANSFORMATION OF TEXT DATA COLLECTIONS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE VIKTOR MARUNA
AUTHOR

VEDOUCÍ PRÁCE Ing. VLADIMÍR BARTÍK, Ph.D.
SUPERVISOR

BRNO 2013

Abstrakt

Tato bakalářská práce se zabývá problematikou získávání znalostí z textů, především se zaměřením na předzpracování a transformaci. V teoretické části práce jsou obsaženy informace o vývoji a metodach procesů získávání znalostí z textů, textových kolekcí dat a využití v praxi. Další část této práce detailně popisuje jednotlivé kroky procesu předzpracování a transformace textových kolekcí dat. V závěrečných částech je přehled o vývoji aplikace, testování a osobní zhodnocení práce.

Abstract

This bachelor thesis deals with the issue of text-mining, mostly focused on preprocessing and transformation. In theoretical part there are contained information about development and principles of text-mining processes, text data collections and use in practice. The next part of this thesis describes in detail single steps of preprocessing and transformation of text data collections. In the final parts there are reviews of application development, testing and personal view on this thesis.

Klíčová slova

získávání znalostí z textů, textové kolekce dat, předzpracování, tokenizace, označování slovních druhů, izolace kořene slova, stop-slova

Keywords

text-mining, text data collections, preprocessing, tokenization, Part-Of-Speech tagging, stemming, stop-words

Citace

Viktor Maruna: Předzpracování a transformace textových kolekcí dat, bakalářská práce, Brno, FIT VUT v Brně, 2013

Předzpracování a transformace textových kolekcí dat

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Vladimíra Bartíka, Ph.D..
Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Viktor Maruna
28. dubna 2013

Poděkování

Rád bych se poděkoval Ing. Vladimíru Bartíkovi, Ph.D., vedoucímu mé bakalářské práce, za vedení, odbornou pomoc a věcné připomínky, které mi ochotně poskytoval.

© Viktor Maruna, 2013.
Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1 Úvod **3**

2 Objavovanie znalostí **5**
2.1 História vzniku 5
2.2 Proces objavovania znalostí 5
2.3 Objavovanie znalostí z textov 6
2.3.1 Porovnanie procesu objavovania znalostí z databáz a z textov 7
2.3.2 Procesy získavania znalostí z textov 8
2.4 Textové kolekcie dát 9
2.5 Textové dokumenty 9
2.5.1 Základné črty textových dokumentov 10
2.6 Získavanie znalostí v praxi 11

3 Predspracovanie a transformácia textových kolekcií dát **13**
3.1 Konverzia elektronických dokumentov na čistý textový formát 14
3.1.1 Obsah textu 14
3.1.2 Formáty 14
3.1.3 Strata relevantných informácií 14
3.1.4 Metainformácie 15
3.2 Tokenizácia 15
3.2.1 Token a term 15
3.2.2 Špecifické problémy 15
3.3 Morfologická analýza tokenu 17
3.3.1 Český a slovenský jazyk 17
3.3.2 Anglický jazyk 17
3.4 Lematizácia, izolácia koreňa slova 18
3.4.1 Izolácia koreňa slova 18
3.4.2 Český a slovenský jazyk 19
3.4.3 Anglický jazyk 19
3.5 Eliminácia stop-slov 22
3.5.1 Spôsoby odstraňovania stop-slov 22
3.6 Váhovanie termov a reprezentácia dokumentov 23
3.6.1 Lokálne váhovanie termov 23
3.6.2 Globálne váhovanie termov 23
3.6.3 Reprezentácia dokumentov 24

4 Implementácia **25**
4.1 Všeobecné ciele práce 25
4.2 Návrh 25
4.3 Výber implementačného jazyka aplikácie 26
4.4 Modul Natural Language Toolkit 26
4.5 Načítanie a konverzia webových elektronických dokumentov na čistý textový formát 27
4.6 Tokenizácia 27
4.7 Part-Of-Speech značkovanie, identifikácia fráz 27
4.7.1 Identifikácia fráz 29
4.8 Izolácia koreňa slova 29
4.9 Eliminácia stop-slov 29
4.10 Grafické užívateľské rozhranie 29
4.10.1 Modul PyQt4 29
4.10.2 Návrh grafického užívateľského rozhrania 29
4.11 Váhovanie termov a reprezentácia výsledkov 30
4.12 Testovanie aplikácie a dosiahnuté výsledky 31

5 Záver **32**

A Obsah CD **34**

Kapitola 1

Úvod

Na základe jedného z prieskumov spoločnosti *International Data Corporation*, ktorá sa zaoberá prieskumom trhu predovšetkým v oblasti informačných technológií, bol v marci roka 2011 publikovaný odborný článok s názvom *Extrahovanie hodnôt z chaosu* [3]. Podľa odhadov spoločnosti ľudstvo vyprodukovalo v roku 2011 približne 1.8 zettabytu dát, pre lepšiu predstavu táto hodnota zodpovedá 1.8 triliónu gigabytov, pričom predpokladajú, že v nasledujúcich rokoch sa môže táto hodnota niekoľkonásobne zvyšovať. Pod týmito dátami si môžeme predstaviť rozličné záznamy z informačných systémov, multimediálne dáta či rôzne textové informácie. V dôsledku toho vznikajú obrovské databázy. S obrovským objemom týchto databáz a zvyšujúcim sa trendom pribúdania dát je stále náročnejšie vyhľadať požadované informácie.

Preto vznikla potreba zrodu technológií pre efektívne vyhľadávanie a triedenie informácií. Vzhľadom k obrovskému objemu dát nie je manuálna analýza možná a rovnako by nebola ani efektívna. Novým postupom v tejto oblasti je vyhľadávanie skrytých informácií z databáz a textov, ktoré sa nazývajú *znalosti*. Znalosti predstavujú taký typ informácie, ktorý nie je možné získať z dát jednoduchým dotazom čo znamená, že tento typ informácie nie je explicitne uložený v dátach. Medzi procesy, ktoré sa zaoberajú získavaním takéhoto typu informácie, môžeme zaradiť napr. proces objavovania znalostí z databáz alebo proces objavovania znalostí z textov.

Táto bakalárska práca sa primárne zaoberá vyššie spomenutou problematikou objavovania znalostí z textov, pričom je zameraná hlavne na jednu z počiatočných fáz dolovania textových znalostí, a to problematiku predspracovania a transformácie textových kolekcií dát. V jednotlivých kapitolách sú popísané teoretické, odborné východiská a viaceré spôsoby týkajúce sa predspracovania a transformácie textových kolekcií dát.

Po prvej kapitole (1), ktorá obsahuje len stručné uvedenie do danej problematiky nasleduje popis kľúčových oblastí týkajúcich sa práce.

Druhá kapitola (2) zachytáva históriu, vývoj procesov a disciplín týkajúcich sa objavovania znalostí z textov a databáz. Vysvetľuje význam viacerých pojmov akými sú textová kolekcia dát alebo textové dokumenty. Popisuje základné charakteristiky procesu objavovania znalostí, ale aj odvodených procesov, a to proces objavovania znalostí z databáz a proces objavovania znalostí z textov. V každej z podkapitol sú vysvetlené pojmy, ktoré sú dôležité pre pochopenie problematiky objavovania znalostí. Niekedy sú vysvetlenia doplnené príkladmi tak, aby bola práca jednoducho a jednoznačne pochopiteľná pre viaceré skupiny čitateľov. Posledná z podkapitol opisuje sposôby využitia procesov objavovania znalostí v praxi.

Tretia kapitola (3) je zameraná na jednu z počiatočných fáz procesu získavania znalostí z textov, a to fázu predspracovania a transformácie textových kolekcií dát. Popisuje viaceré metódy, algoritmy, operácie a pojmy, ktoré sú príznačné pre túto fázu. Jednotlivé podkapitoly sa zaoberajú tokenizáciou, lematizáciou, morfologickou analýzou, izoláciou koreňa slova, vytváraním fráz a stop-slovami a reprezentáciou výsledkov.

Štvrtá kapitola (4) popisuje vlastnú implementáciu. V úvode je popísaný celkový návrh aplikácie, vymenúva použité moduly, spôsob implementácie metód predspracovania textu a návrh grafického užívateľského rozhrania. Rovnako sú vysvetlené použité technológie.

Poslednou kapitolou je záver (5), v ktorom je uvedené zhrnutie a zhodnotenie postupu práce a dosiahnuté výsledky, ale rovnako aj osobný dojem pri tvorbe a vypracovaní práce. Zároveň sú tu načrtnuté možnosti vylepšenia práce, postupy a možnosti využitia práce v budúcnosti.

V prílohe je popísaný obsah CD (A), ktoré bolo priložené k tomuto dokumentu.

Kapitola 2

Objavovanie znalostí

Nasledujúca kapitola popisuje kľúčové prvky a základné procesy týkajúce sa objavovania znalostí. Informácie k tejto kapitole boli čerpané predovšetkým z publikácií [2][4][7][8].

2.1 História vzniku

Dolovanie textových dát je pomerne novou oblasťou výskumu v oblasti informačných technológií. Ešte v nedávnej dobe bola väčšina textových informácií uchovávaná v klasickej forme a to tak, že tieto textové informácie boli zaznamenané na papieri. S obrovským nástupom informačných technológií sa v súčasnej dobe tento spôsob razantne zmenil.

S postupom času sa pre uchovanie týchto textových informácií stále viac presadzuje podoba elektronických dokumentov. Ako príklad sa dá uviesť informačné systémy v podnikoch či iných odvetviach, elektronické kartotéky, archívy a knižnice. V dnešnej dobe je z týchto podôb na vrchole web, s ktorým sa stretávajú milióny ľudí denne.

Dôvodov, prečo vytvárať dokumenty v elektronickej podobe či uskutočňovať transformácie dokumentov do elektronickej podoby je mnoho. Hlavnou výhodou a dôvodom ukladania textových informácií v tejto podobe je zníženie nákladov na archiváciu. Pre dokumenty uložené v tejto podobe je charakteristická aj ich znovupoužiteľnosť. Dôležitou výhodou je možnosť použitia sofistikovaných softvérových nástrojov a prostriedkov umožňujúcim prístup k elektronickým dokumentom. Výrazne sa zjednodušuje a urýchľuje práca s takýmto typom dokumentov, kedy je možné efektívne analyzovať texty, navzájom ich porovnávať, triediť ich do rozličných kategórií, štruktúrovať a integrovať s inými typmi údajov.

Práve tieto typy úloh komplexne rieši oblasť dolovania znalostí z textových dát. Medzi úlohy komplexného dolovania znalostí z textových dát zaraďujeme metódy predspracovania a transformácie, kategorizácie, zhlukovania, extrakcie informácií či mnohé iné.

2.2 Proces objavovania znalostí

Objavovaním znalostí sa nazýva proces *semiautomatickej*, príp. *automatickej extrakcie* znalostí z rôznych typov dát.

Najčastejšie ide o extrakciu dát z databáz (angl. *Knowledge Discovery in Databases*, skr. *KDD*), ale poznáme aj ďalšie typy akými sú napr. dolovanie znalostí z webu, dolovanie znalostí z multimediálnych dát alebo extrakciu dát z textov (angl. *Knowledge Discovery in Texts*, skr. *KDT*), označovaná aj ako dolovanie znalostí z textov (angl. *Text Mining*, skr. *TM*).

Pre extrahované znalosti je charakteristické, že sú:

- doposiaľ neznáme,
- potenciálne užitočné,
- platné (v štatistickom zmysle).

Ako bolo už naznačené, proces objavovania znalostí je *proces*. Znamená to, že sa skladá z viacerých deterministicých krokov. Tieto kroky budú podrobnejšie popísané v časti 2.3.2 tohto dokumentu. Aj keď v teoretickej rovine je postupnosť týchto krokov presne daná, tak samotná realizácia a implementácia už nie. Realizácia samotného procesu už nepredstavuje len jednoduchý automatický prechod medzi viacerými jednotlivými deterministickými krokmi, ale má dve podstatné vlastnosti. Proces objavovania znalostí či už z databáz alebo z textov označujeme ako *iteratívny* a *interaktívny*.

Pri plnej automatizácii procesu objavovania znalostí iba ťažko možno dosiahnuť optimálne výsledky a zisk takých informácií a znalostí, ktoré budú skutočne cenné, a ktoré budú schopné reálneho prínosu. Práve z tohto dôvodu sa proces objavovania znalostí neuskutočňuje automaticky, ale semiautomaticky. Pri realizácii samotného procesu je účasť a asistencia človeka výhodná, dokonca v niektorých prípadoch až nevyhnutná. Človek je schopný rozhodnúť o výbere vhodných operácií pri predspracovaní, rozhoduje aj o výbere algoritmov a rovnako aj parametrov v rámci jednotlivých krokov.

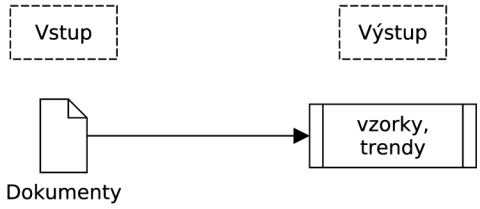
Pre proces objavovania znalostí je dôležitý aj ten fakt, že len človek je schopný jednoznačne rozhodnúť o tom, ktorá z objavených znalostí spĺňa požadované parametre (ak je objavená znalosť platná, doposiaľ neznáma a dostatočne užitočná), aby ju bolo možné s úspechom aplikovať a využiť v danej skúmanej aplikácii.

Ako je už známe, proces objavovania znalostí je v rámci svojich jednotlivých fáz nedeterministický. V každom zo svojich krokov sú možné viaceré možnosti a alternatívne rozhodnutia. Následkom toho sa pri výbere alternatívnej operácie, algoritmu alebo zmene parametrov operácie ovplyvňujú nasledujúce kroky realizácie procesu, ktoré sú spôsobené rozličnými výstupmi. V dôsledku toho dochádza často v rámci procesu objavovania znalostí k iteráciám v rámci jedného alebo viacerých krokoch, pričom hlavným cieľom je dosiahnutie čo najlepšieho výsledku.

Práve tieto fakty vedú k tomu, že proces objavovania znalostí je označovaný ako iteratívny a interaktívny.

2.3 Objavovanie znalostí z textov

Základná a zároveň najjednoduchšia definícia dolovania v textoch je odvodená podľa základnej definície pre objavovanie znalostí a to tak, že je to iteratívny a interaktívny proces získavania platných, pre danú aplikáciu doposiaľ neznámych a užitočných znalostí, pričom algoritmy, ktoré sa používajú, získavajú platné vzory rôznych typov a ktorých interpretácia a posúdenie užitočnosti je ponechané na človeka (Obrázok 2.1). Preto je interakcia používateľa s analytickým systémom nevyhnutná pre úspech realizácie procesu.



Obrázok 2.1: Proces objavovania znalostí z textov.

2.3.1 Porovnanie procesu objavovania znalostí z databáz a z textov

Proces objavovania znalostí z textov môže byť definovaný ako proces, v ktorom užívateľ interaguje s dokumentom, resp. s kolekciou dokumentov v čase, pričom s použitím vhodných analytických nástrojov sa získavajú užitočné znalosti. Spoločnou charakteristickou vlastnosťou objavovania znalostí z databáz a objavovania znalostí z textov je snaha o extrahovanie užitočných informácií a znalostí z dátových zdrojov prostredníctvom vyhľadávania a identifikácie vzoriek (Obrázok 2.1).

Vyhľadávanie vzoriek

Pre proces objavovania znalostí z textov je typické, že zdrojom sú textové kolekcie dát. Vzorky nie sú vyhľadávané medzi formalizovanými štruktúrovanými záznamami v databázach, ale v neštruktúrovaných textových informáciách uložených v textových dokumentoch, ktoré sú súčasťou textových kolekcií dát.

Napriek tomu, že pojmy ako objavovanie znalostí z databáz a objavovanie znalostí z textov majú spoločné vlastnosti a úzko spolu súvisia, tak proces objavovania znalostí z textov sa vzhľadom k neurčitosti jazyka a jeho inherentnej neštruktúrovanosti preukazuje ako omnoho zložitejší než proces objavovania znalostí v databázach.

Architektúra systémov

Architektúra systémov pre získavanie znalostí z databáz a systémov pre získavanie znalostí z textov je veľmi podobná. Oba typy systémov sú založené na operáciách pre predspracovanie, algoritmy pre vyhľadávanie vzoriek a prezentačnú vrstvu. Prezentačnú vrstvu reprezentujú technické nástroje pre vizualizáciu nájdených vzoriek.

Predspracovanie dát

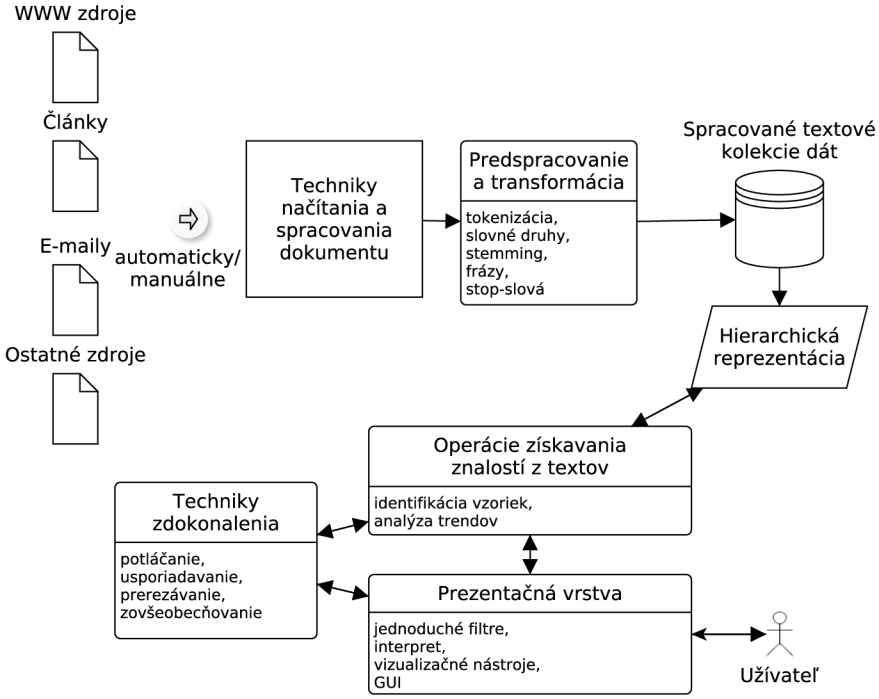
Z hľadiska predspracovania dát sú operácie predspracovania u systémov pre získavanie znalostí z textov zamerané predovšetkým na identifikáciu a extrahovanie vzoriek z dokumentov prirodzeného jazyka.

Úlohou operácií predspracovania je transformácia neštruktúrovaných dát (text zložený z prirodzeného jazyka) uložených v kolekcii dokumentov do štrukúrovaného (často označovaného aj ako prechodného) formátu.

Systémy pre získavanie znalostí z textov teda využívajú niektoré operácie, postupy a algoritmy ako systémy pre získavanie znalostí z databáz, ale rovnako ich dopĺňajú o techniky a metodológie z oblastí vyhľadávania informácií (angl. *information retrieval*), extrahovaní informácií a korpusovo založených lingvistikách.

2.3.2 Procesy získavania znalostí z textov

Pre procesy získavania znalostí z textov bol navrhnutý *systém získavania znalostí z textov*. Architektúra takéhoto systému je znázornená na obrázku 2.2.



Obrázok 2.2: Architektúra systému pre získavanie znalostí z textov.

Celkový proces získavania znalostí z textov je možné (z obrázku 2.2) rozdeliť do 6 fáz:

- *Identifikácia aplikačnej domény* - užívateľ musí identifikovať kľúčové koncepty aplikačnej domény a stanoviť cieľ procesu získavania znalostí z textov.
- *Získanie relevantnej množiny dokumentov* - dokumenty sa získavajú z existujúcich zdrojov, a to buď manuálne, alebo nástrojmi pre získavanie informácií, príp. sa môžu oba predchádzajúce spôsoby kombinovať.
- *Predspracovanie textových kolekcií dát* - obsahuje operácie, procesy a metódy potrebné k príprave dát pre operácie získavania znalostí z textov, ktoré sú implementované v jadre systémov získavania znalostí z textov. Problematika predspracovania a transformácie textových kolekcií dát bude podrobnejšie vysvetlená v kapitole 3.
- *Operácie získavania znalostí z textov* - tvoria jadro systémov získavania znalostí z textov. Sú zložené z metód pre vyhľadávanie vzoriek, analýzu trendov a rôzne ďalšie metódy pre vyhľadávanie a získavanie znalostí. Táto časť systému predstavuje samotnú aplikáciu zvoleného algoritmu pre získavanie znalostí z textov. Môže ísť o operácie pre extrahovanie informácií, zhlukovanie, kategorizáciu alebo klasifikáciu textov. Operácie pre extrahovanie informácií sú však častejšie zaraďované a chápané ako súčasť predchádzajúcej fázi, a to predspracovania textových kolekcií dát.

- *Prezentačná vrstva a jej komponenty* - predstavuje súbor technických nástrojov na vhodnú vizuálnu reprezentáciu nájdených vzoriek alebo modelov, ktoré boli vytvorené v predchádzajúcom kroku, teda pomocou metód a operácií pre získavanie znalostí z textov. Dôležitým faktom však je, že samotnú interpretáciu výsledkov musí uskutočniť užívateľ.

- *Techniky zdokonalenia* - obsahujú metódy, ktoré sú schopné filtrovať redundantné a neužitočné informácie a zároveň zhlukujú úzko súvisiace dáta. Takýmito technikami sú potláčanie (angl. *supression*), usporiadavanie, prerázavnie v stromových štruktúrach (angl. *pruning*) a zovšeobecňovanie. Tieto techniky sú častokrát označované aj ako pospracovanie (angl. *postprocessing*).

Najdôležitejšími a najkritickejšími časťami architektúry systémov pre získavanie znalostí z textov sú fázy predspracovanie dát a následne operácie získavania znalostí z textov.

2.4 Textové kolekcie dát

Jedným z kľúčových prvkov a dôležitým elementom v oblasti získavania znalostí z textov sú *textové kolekcie dát.*

Najjednoduchšou definíciou, ktorá popisuje textové kolekcie dát je, že je to komplexné zoskupenie viacerých textovo založených dokumentov. V praxi to predstavuje desiatky až stovky miliónov textových dokumentov združených do kolekcií, v ktorých najčastejšie prebieha vyhľadávanie vzoriek.
Textové kolekcie dát môžeme deliť na 2 skupiny:

- *statické* - predstavujú také textové kolekcie dát, ktorých stav, teda obsah, zostaváva nezmenený.

- *dynamické* - sú charakterizované ako textové kolekcie dát, ktorých dokumenty sa môžu v čase meniť. Znamená to, že buď sa môže obsah týchto dokumentov aktualizovať, alebo môžu pribúdať úplne nové dokumenty.

Extrémne veľké textové kolekcie dát a rovnako aj textové kolekcie dát s mnohými aktualizáciami v obsahu podliehajú veľmi často rôznym optimalizačným metódam a postupom. Deje sa tak v systémoch pre získavanie znalostí z textov.

Ako ilustrácia pre skutočné textové kolekcie dát využívané v praxi môže poslúžiť repozitár *PubMed*, teda repozitár obsahujúci práce biomedicínskeho výskumu, ktorý spravuje *Národná knižnica medicíny Spojených štátov amerických* na Národnom inštitúte zdravia v štáte Maryland. Bližší popis využitia textových kolekcií dát a systémov pre získavanie znalostí z textov v praxi bude predstavený podkapitole 2.6.

2.5 Textové dokumenty

Ďalším základným a podstatne dôležitým elementom v oblasti získavania znalostí z textov sú *textové dokumenty.*

Pre praktické účely môžu byť textové dokumenty definované neformálne ako jednotka diskrétnych textových dát, ktorá je súčasťou textových kolekcií dát. Zároveň je táto jednotka dát obvykle (nie vždy a nevyhnutne) vo vzájomnom vzťahu s nejakým skutočným dokumentom. Najčastejšie sa jedná o dokumenty typu e-mail, rôzne správy a novinové články, články z oblasti výskumu a mnohé iné.

Jeden konkrétny textový dokument môže byť v rovnakom čase súčasťou aj viacerých rozdielnych textových kolekcií dát a zároveň textový dokument sa može vyskytovať vo viacerých typoch textových kolekcií dát, a to od formálne organizovaných až po rôzne ad hoc typy.

Hoci sa na textové dokumenty hľadí a označujú sa ako neštruktúrované dáta, tak z viacerých perspektív sa môžu javiť ako dáta štruktúrované. Z pohľadu lingvistiky sa dá na tieto dáta pozerať ako na určitú syntaktickú a sémantickú štruktúru, ktorá je implicitná a skrytá vo svojom textovom kontexte.

Zároveň sa v texte vyskytujú rôzne typografické elementy ako napr. diakritika, číselné údaje, veľké alebo malé písmená a rozličné špeciálne znaky. Častokrát je text a jeho umiestnenie obohatené o viaceré prvky a artefakty, medzi ktoré môžeme zaradiť biele (neviditeľné) znaky, stĺpce, tabuľky, podčiarknutie, odrážky atď.. Všetky predchádzajúce spomenuté elementy bývajú označované ako druh jazyka (v anglickej terminológii ako *soft markup language*), ktorý pomáha identifikovať dôležité subkomponenty v textových dokumentoch (titulky, odseky, hlavičky a nadpisy a pod.).

Z tohto hľadiska sa preto textové dokumenty zvyknú deliť do troch skupín:

- *neformátované textové dokumenty* - vedecké a výskumné práce, obchodné správy.
- *slaboštruktúrované textové dokumenty* - e-mail, PDF súbory, HTML web stránky alebo serializovaná dáta vo forme napr. XML či JSON súboru. Obsahujú formátovacie elementy, ktoré predstavujú metadáta.
- *semištruktúrované textové dokumenty* - textové dokumenty, resp. súbory z textových procesorov s viacerými šablónami dokumentov.

2.5.1 Základné črty textových dokumentov

Medzi základné črty textových dokumentov zaraďujeme znaky, slová, termy a koncepty.

Znaky

Sú to komponenty textových dokumentov ako písmená, medzery, číslice alebo špeciálne znaky. Predstavujú základné stavebné prvky pre vyšší level sémantiky, a to slová, termy a koncepty.

Slová

Špecifické slová, ktoré sú vyberané priamo z textových dokumentov, a ktoré bývajú označované ako základná úroveň sémantiky.

Termy

Termy sú osamotené slová alebo viacslovné frázy vyberané z korpusu textových dokumentov pomocou rôznych postupov a metodológií na extrahovanie termov. Viaceré metodológie na extrahovanie termov dokážu konvertovať neštruktúrovaný text textového dokumentu do tzv. *normalizovaných termov.*

Normalizované termy sú sekvencie jedného alebo viacerých tokenizovaných a lematizovaných slov, doplnené o značkovanie slovných druhov.

Problematika termov ako osamotených slov a viacslovných fráz je rozvedená na nasledujúcom príklade:

Novak Djokovic, born in Socialist Federal Republic of Yugoslavia, is a Serbian professional tennis player.

Zoznam termov reprezentujúcich osamotené slová môže byť nasledovná:

Novak, *Djokovic*, *born*, *Socialist*, *Federal*, *Republic*, *Yugoslavia*, *Serbian*, *profesional*, *tennis*, *player*

Naopak zoznam termov reprezentujúcich viacslovné frázy by mohol mať viacero podôb. Jednou z nich je nasledujúci zoznam:

Novak Djokovic, *Socialist Federal of Republic Yugoslavia*, *Serbian professional tennis player*

Koncepty

Koncepty sú vlastnosti textových dokumentov generované pomocou prostriedkov kategorizačných metodológií, medzi ktoré patria manuálne, štatistické, hybridné a na pravidlách založené metodológie. Nakoľko problematika konceptov prekračuje rámec tejto práce, v nasledujúcich kapitolách jej už nebude venovaná pozornosť.

Pre čitateľov, ktorí majú záujem o hlbšie naštudovanie a pochopenie tejto oblasti môže poslúžit práca Dr. Paula Losiewicza, Dr. Douglasa W. Oarda a Dr. Ronalda N. Kostoffa s názvom *Veda a technológia získavania znalostí z textov: Základné koncepty* [4].

2.6 Získavanie znalostí v praxi

Ako bolo v tejto kapitole už viackrát spomenuté, oblasti získavania znalostí z databáz a získavania znalostí z textových dokumentov majú v dnešej doba čoraz väčší význam a prikladá sa im stále väčšia dôležitosť. Získavanie znalostí môže byť užitočné a dokonca niekedy aj nevyhnutné vo viacerých aplikačných oblastiach, kde sa sústreďuje veľké množstvo dát. Či už sa jedná o oblasti bežného života, alebo oblasti vedy a výskumu, získavanie znalostí si získava svoju opodstatnenosť. Dôkazom toho je aj vznik *Národného centra pre získavanie znalostí z textov* sídliaceho na Manchesterskom Inštitúte Biotechnológií.

Biomedicína

V tejto časti sa opät dostávame k pojmu PubMed, teda repozitáru obsahujúcemu práce biomedicínskeho výskumu. Podľa posledných údajov [11] obsahuje tento repozitár viac ako 22 miliónov výskumných dokumentov publikovaných už od roku 1966, a preto PubMed predstavuje najobsiahlejšiu a najkomplexnejšiu kolekciu vedeckých článkov, pričom toto množstvo článkov neustále stúpa. V dôsledku toho sa repozitár PubMed stal jednou z najvýznamnejších textových kolekcií dát, ku ktorej je upriamená pozornosť viacerých výskumníkov a vedcom venujúcim sa problematike získavania znalostí z textových kolekcií dát.

Astronómia

V oblasti astronómie ide o systémy pre získavanie znalostí, ktoré analyzujú dáta prijaté z vesmíru k objavovaniu nových astronomických objektov, najmä objavovanie hviezd a galaxií. Získavaním znalostí z textových kolekcií dát za podrobnejšie zaoberá aj americký *Národný úrad pre letectvo a vesmír*, ktorý sa vo svojom výskume *Dashlink* [1] za pomoci metód ako kategorizácia a zhlukovanie snaží prispievať do oblasti bezpečnosti komerčných, ale aj vesmírnych letov.

Marketing

Procesy získavania znalostí významnou mierou zasahujú aj do sféry marketingu. Nástroje pre získavanie znalostí sú prispôsobené do takej miery, aby vyhľadávaním a kategorizáciou informácií boli schopné dopomôcť k efektívnym výsledkom reklamných kampaní.

Riadenie podnikov a podpora obchodovania

Spadajú sem nástroje pre strategické rozhodovanie v riadiacej sfére podnikov, kedy je ich hlavnou úlohou uskutočňovať podporu pre podnikateľské zámery. Pre podporu obchodovania boli vyvinuté systémy, ktoré sa snažia zvýšiť zisky obchodných spoločností. Najčastejšie ide o systémy, ktoré analyzujú nákupy zákazníka v prostredí elektronických obchodov, príp. aj v kamenných obchodch, kedy sa snažia podľa nakúpeného tovaru určiť o aký tovar by mohol mať zákazník ešte záujem, čím zvyšujú pravdepodobnosť ďalšieho nákupu a teda aj zisku spoločnosti.

Kriminalita

Pre oblasť kriminality boli navrhnuté viaceré komplexné systémy, ktorých hlavným cieľom je odhaľovanie podvodov. Medzi najdôležitejšie úlohy takýchto systémov patrí odhaľovanie podvodov hlavne v bankovom sektore, a to pri podvodoch s platobnými kartami či neoprávnenými platbami prostredníctvom elektronického bankovníctva.

Oblasť internetovej bezpečnosti

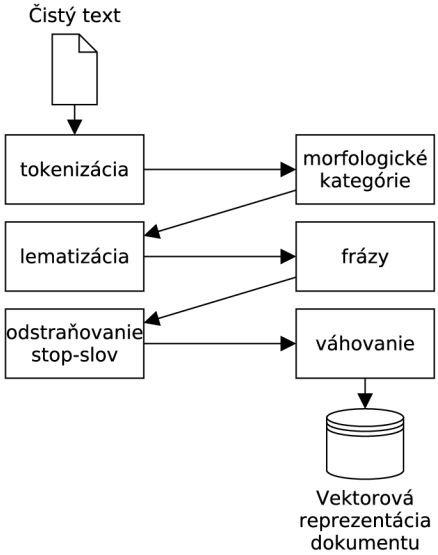
Viaceré nástroje získavania znalostí z textových dokumentov sú prispôsobené tak, aby boli schopné bezpečnostnej analýzy a monitoringu texových dát na webe. Využívajú sa najmä v oblastiach, kde pribúda každým dňom množstvo nového obsahu. Tieto nástroje sa využívajú predovšetkým na monitoring internetoých novín, blogov a pod., za účelom ochrany užívateľa pred potenciálne nebezpečným obsahom. Niekedy za zvyknú tieto nástroje využívať aj za účelom národnej bezpečnosti a ochrany.

Kapitola 3

Predspracovanie a transformácia textových kolekcií dát

Predspracovanie a transformácia textových kolekcií dát je jednou z prvých fáz procesu získavania znalostí z textov. Kedže v tejto fázi nedochádza priamo k použitiu špecifických algoritmov, metód a operácií pre získavanie znalostí, odborníkmi je často pomenovaná ako prípravná fáza aplikácie metód a postupov pre získavanie znalostí.

Samotná fáza predspracovania sa častokrát zvykne označovať a definovať aj ako získavanie príznakových popisov textových dokumentov z textových kolekcií dát. Táto fáza pozostáva zo sekvencie viacerých čiastkových krokov, ktoré sú znázornené na obrázku 3.1.

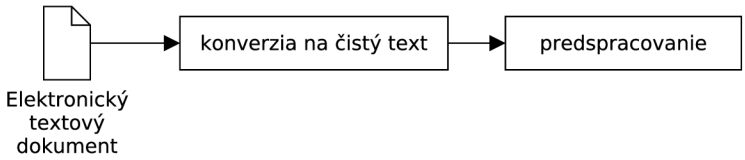


Obrázok 3.1: Proces predspracovania a transformácie textových kolekcií dát.

V nasledujúcich podkapitolách budú podrobnejšie popísané jednotlivé fázy predspracovania a transformácie textových kolekcií dát, spôsoby konverzie elektronických dokumentov na textový formát a možnosti reprezentácie textových dokumentov ako vstupov pre hlavný proces získavania znalostí. Pre bližšie porozumenie problematike získavania znalostí mi boli nápomocné publikácie [2][7][8].

3.1 Konverzia elektronických dokumentov na čistý textový formát

Jednou z predprípravných fáz k predspracovaniu textových kolekcií dát je proces konverzie vstupných elektronických dokumentov na *čistý textový formát* (Obrázok 3.2). Čistý textový formát alebo čistý text možno popísať ako sekvenciu alfanumerických, interpunkčných či rôznych iných oddeľovacích grafických znakov alebo špeciálnch symbolov, ku ktorým môžme zaradiť znaky ako napr. percento(%), ampérsand(&), zavináč(@).



Obrázok 3.2: Predprípravná fáza k predspracovaniu textových kolekcií dát.

3.1.1 Obsah textu

Každá z týchto skupín znakov plní v texte svoju funkciu. Sekvencie alfanumerických znakov predstavujú lexikálnu a fonetickú hodnotu. V dôsledku toho ich možno označiť za priamych nositeľov obsahu textu. K členeniu obsahu textu sa využívajú interpunkčné a oddeľovacie znaky akými sú napr. bodka, čiarka, lomka, zátvorky, pomlčka, medzera, tabelátor.

Hlavnou úlohou procesu konverzie elektronických dokumentov na čistý textový formát je teda odstránenie všetkých netextových informácií a typografických značiek zo vstupného textu. V praxi ide najčastejšie o odstránenie formátovacích značiek (napr. typ, veľkost, farba písma, odsadzovanie textu a pod.), ale častokrát aj odstraňovanie obrázkov, grafických elementov, grafov, tabuliek.

3.1.2 Formáty

Oblasť získavania znalostí z textových kolekcií dát sa zaoberá dokumentami rôznych formátov, pričom pre každý z týchto formátov sú špecifické iné vlastnosti a rôzny spôsob konverzie elektronického dokumentu do čistého textového formátu. Najčastejšie používné vstupné formáty vo fázi konverzie elektronických dokumentov na čistý textový sú: *HTML*, *XML*, *DOC*, *DOCX*, *RTF*, *PDF*, *PS*. Ďalšími formáty, ktoré je možné využívať v procese konverzie elektronických dokumentov na čistý textový formát sú *ODT*, *TeX*, *XLS*, či mnohé iné.

3.1.3 Strata relevantných informácií

Počas procesu konverzie elektronických dokumentov na čistý textový formát sa v mnohých prípadoch strácajú dôležité a relevantné informácie o obsahu textu a jeho štruktúre. Niektoré z týchto informácií môžu byť v neskoršom procese získavania znalostí z textov dôležité a užitočné, a preto sa zvyknú určitým spôsobom zachovávať. Možnosťou je zakódovanie dôležitých informácií v modifikovanom formáte do čistého textu, napr. pomocou XML štruktúry alebo archivovanie pôvodného dokumentu spolu s uloženou referenciou na novovytvorený čistý textový formát.

3.1.4 Metainformácie

Údaje, ktoré sa pri konverzii elektronických dokumentov na čistý textový formát strácajú, avšak dajú sa odvodiť z pôvodného elektronického textového dokumentu nazývame *metainformácie*. Často sa zvyknú tieto údaje v procese konverzie uchovávať (napr. vo formáte XML) a využívať v neskorších fázach procesu získavania znalostí z textov.

Medzi metainformácie môžeme zaradiť:

- *informácie o štruktúre textu* - štruktúrovanie textu na odseky, kapitoly, nadpisy, zalomenia riadok a strán alebo typografické informácie ako veľkosť a typ písma, odsadzovanie atď..
- *metaúdaje* - informácie, ktoré charakterizujú obsah textu a sú zväčša vkladané do dokumentu autorom.
- *bibliografické údaje* - autor, miesto a dátum publikovania, zdroj, vydavateľ.
- *vlastnosti dokumentu* - názov, umiestnenie alebo veľkosť elektronického textového dokumentu.

3.2 Tokenizácia

V prvotnej fázi procesu predspracovania a transformácie textových kolekcií prichádza ku tokenizácii, ktorá by sa dala označiť aj ako fáza identifikácie. Tokenizácia je tvorená inicializačnými textovými operáciami, pričom ich hlavnou úlohou je identifikácia základných textových lexikálnych jednotiek (slová, slovné spojenia, vety, odseky a pod.) vo vstupnom texte. Elementárne textové jednotky sú konvertované na lexikálne jednotky - *tokeny* (značky).

Proces tokenizácie je závislý od jazyka vstupného textu. V dôsledku toho je pre automatizovanú tokenizáciu potrebné zisťovať jazyk takýchto vstupných textov, a to rôznymi heuristickými operáciami a algoritmami.

3.2.1 Token a term

Veľmi dôležitým faktom je aj ujasnenie rozdielu medzi termom a tokenom, ktorých definície bývajú mnohokrát nejednoznačné alebo zamenené.

Termy sú označované ako kľúčové slová, ktoré sú súčasťou reprezentácie dokumentov. Sú vytvárané priamo z tokenov vo fázach lematizácie, jazykovej analýzy a rovnako aj vo fáze eliminácie stop-slov.

Na rozdiel od toho tokeny predstavujú vymedzenú časť vstupného textu s určitým konkrétnym slovníkovým významom a jednoznačnou pozíciou v texte. Nie každý token je v konečnej fázi transformovaný na term, ale každý term je tvorený z jedného, príp. viacerých tokenov.

3.2.2 Špecifické problémy

V samotnom procese tokenizácie sa môže vyskytnúť viacero špecifických problémov a javov. V mnohých prípadoch však neexistuje jednoznačná odpoveď na to, akým spôsobom vzniknuté problémy riešiť. Preto je častokrát riešenie problému závislé na konkrétnej implementácii, alebo je ponechané do rúk užívateľa, ktorý s takýmto systémom pre získavanie znalostí z textov pracuje.

Medzi takéto javy zaraďujeme:

- spájanie voľnejších zložených jednotiek
- zložené slovné spojenia
- veľkosť písma
- interpunkcia
- čísla a alfanumerické reťazce

Spájanie voľnejších zložených jednotiek

Špecifickým problémom a pomerne častým javom v procese tokenizácie je spájanie voľnejších zložených jednotiek akými sú napr. zložené slovné spojenia "in the case of" alebo "data flow". V takýchto prípadoch, kedy zložené textové jednotky nezodpovedajú štandardným slovníkovým záznamom, je vhodnejší spôsob reprezentovať všetky jednotlivé zložky zloženého výrazu osobitným tokenom. K operáciam spájania je možné sa dostať aj v neskorších fázach a to najčastejšie vo fáze morfologickej alebo syntaktickej analýzy. Všetky jednotlivé postupy a algoritmy sú však závislé od výberu použitého slovníka, jeho pravidlového systému a konkrétneho jazyka textu.

Zložené slovné spojenia

Problémy sa naskytujú aj pri zložených slovách a slovných spojeniach obsahujúcich spojovník. O tom, či slovné spojenia ako "Wi-Fi", "Marc-André" alebo "Poprad-Tatry" ponechať ako jeden token, alebo slovné spojenie rozdeliť neexistuje jednoznačné riešenie. Preto sa najčastejšie takéto javy uchovávajú ako jeden samostatný token. Jedným z ďalších riešení je rozdelenie slovného spojenia na dve samostatné jednotky a následné overenie toho, či sa nachádzajú v slovníku prípustných tvarov. V prípade, že pre obe samostatné jednotky je možné nájsť v slovníku prípustných tvarov ich prípustný slovníkový záznam, potom je možné zložený tvar rozdeliť na samostatné jednotky a tieto zložky ďalej reprezenovať ako dva samostatné tokeny.

Veľkosť písma

Dôležitú úlohu v procese tokenizácie zohráva aj veľkosť písma, teda malé a veľké písmená abecedy. V procese tokenizácie môže dochádzať ku konverzii veľkého písmena na malé. To však môže mať na výsledný efekt negatívny vplyv vzhľadom k tomu, že táto konverzia môže predstavovať stratu sémantickej informácie. V prípade, že slovo začína veľkým písmenom neznamená, že toto slovo označuje začiatok vety, a v dôsledku toho nemusia mať veľké písmená absolútnu sémantickú platnosť. Práve preto je vhodné zachovávať informáciu o veľkosti písma, a teda aj tvare tokenu.

Interpunkcia

Najčastejšie sa používajú interpunkčné znaky ako rôzne formy oddeľovačov, či už pre oddelenie slov alebo viet. Mnohokrát sú však interpretované iným spôsobom, a to predovšetkým v zložených slovách. Takými príkladmi sú zložené slová ako "miles/hour" alebo aj adresy elektronickej pošty v tvare "mail@domain.com" kedy interpunkčné znaky neplnia funkciu

oddeľovačov, ale sú súčasťou spájania zložených slov. V takýtoch prípadoch je potrebné prispôsobiť pravidlá systému tokenizácie takým spôsobom, aby bol systém schopný identifikácie niektorých typov zložených slov, ktoré obsahujú interpunkčné znaky.

Čísla a alfanumerické reťazce

Problémy vo fázi tokenizácie nastávajú aj pri identifikácii čísel a alfanumerických reťazcov a to vzhľadom k ich neurčitosti. V niektorých prípadoch je potrebné identifikovať číslo ako jednoznačný term. To nastáva napr. pri identifikácii telefónnych čísel, čísel bankových účtov, poznávacích čísel vozidiel. V ostatných prípadoch môžu byť čísla súčasťou slovných spojení kedy predstavujú rôzne typy označení (napr. technických ako "McLaren MP4 12"). Práve preto sú aj pri identifikácii čísel a alfanumerických reťazcov viaceré formy reprezentácie. Je možné zvoliť prístup identifikácie čísla ako tokenu s čo najširším textovým okolím a ak sa identifikované číslo nepodarí tokenizovať v určitom predpísanom formáte (napr. vo formáte dátumu alebo telefónneho čísla), dochádza k reprezentácii čísla ako samostatného tokenu.

3.3 Morfologická analýza tokenu

Vo fázi morfologickej analýzy tokenu dochádza k ohodnocovaniu tokenov morfologickými kategóriami pomocou *značiek*. Základná značka vyjadruje slovný druh tokenu. V závislosti od slovného druhu môžu ďalej značky udávať rôzne gramatické kategórie, ako napr. osobu, číslo, čas, rod alebo pád.

3.3.1 Český a slovenský jazyk

Zložitosť morfologickej analýzy je závislá od konkrétneho jazyka. Kým v jazykoch ako čeština alebo slovenčina je v dôsledku zložitosti morfologického systému pomerne obtiažna, tak v jazyku ako angličtina je morfologická analýza riešená na pomerne jednoduchšej úrovni.

3.3.2 Anglický jazyk

Proces morfologickej analýzy tokenu sa pri spracovaní textov v anglickom jazyku označuje ako identifikácia (značkovanie) slovných druhov (angl. *Part-of-Speech tagging*, skr. *POS* alebo *POS tagging*). Určovanie slovných druhov sa častokrát prevádza aj na základe vzájomného vzťahu sekvencie slov a ich poradia vo vetách. V závislosti od jazyka textu a metód využitých pri získavaní znalostí z textov niekedy nedochádza k detailnému značkovaniu tokenov, ale určuje sa len základná značka - slovný druh.

Pre proces POS existuje v praxi viacero prístupov, a to POS značkovanie založené na:

- slovníku
- pravidlách
- špeciálnych metódach (štatistických, pravdepodobnostných metódach, neurónových sieťach, skrytých markovovských modeloch atď.)
- hybridných metódach (kombinácia viacerých metód)

POS značkovanie založené na slovníkoch

Táto metóda patrí medzi jednu z najjednoduchších a v praxi najčastejšie využívaných metód. Medzi jej hlavné nevýhody patrí rozsiahlosť slovníka, ktorý v mnohých prípadoch nepokrýva celý rozsah slovnej zásoby daného jazyka a následne nie je možné pri neznámych slovách určiť značky.

POS značkovanie založené na pravidlách

Tieto metódy využívajú rozsiahly zoznam pravidiel vytvorený ručne expertami pre daný jazyk alebo sa pre ich tvorbu využívajú princípy strojového učenia, ktoré automaticky odvodzujú pravidlá zo vstupného trénovacieho korpusu.

3.4 Lematizácia, izolácia koreňa slova

V dôsledku toho, že vo vstupnom texte sa slová vyskytujú v rôznych morfologických tvaroch (osoby, číslo, pád, spôsob atď.), je niekedy potrebná ich transformácia na základný tvar, tzv. *lemu*, ktorá je základným slovníkovým tvarom tokenu. Pri slovesách lema predstavuje neurčitok, pri podstatných a prídavných menách je to prvý pád jednotného čísla.

Lematizácia je teda proces, ktorý zo základného slova vo vstupnom texte určí základný tvar. Najčastejšie sa táto transformácia prevádza odstraňovaním pádových, slovotvorných či rôznych ďalších prípon (sufixov) a predpôn (prefixov).

3.4.1 Izolácia koreňa slova

Jednou zo špeciálnych foriem lematizácie je *izolácia koreňa slova* (angl. *stemming*). Pri tomto procese sa slovo konvertuje na základný tvar tak, že sa nahrádza svojim kmeňovým základom (koreňom slova), pričom sa predpokladá, že význam základného tvaru slova je totožný s každým pôvodným gramatickým tvarom.

Dôvod, prečo v procese predspracovania a transformácie textových kolekcií dát dochádza k izolácii koreňa slova je ten, že touto transformáciou je možná významná redukcia počtu termov. Zo slova sa postupne odstraňujú predpony a prípony (pádové, slovotvorné atď.) takým spôsobom, že zostáva iba základ slova, ktorý je identifikovaný ako term, tj. kľúčové slovo.

Proces izolácie koreňa slova je znázornený na nasledujúcom príklade:

$$\{\textit{work; work-s; work-ing; work-ed}\} \rightarrow \textit{work}$$

Izoláciu koreňa slova je možné uskutočňovať:

- odstraňovaním predpôn a prípon
- pomocou slovníka koreňov
- s využitím štatistickej metódy

Odstraňovanie predpôn a prípon

Metóda izolácie koreňa slova pomocou odstraňovania predpôn a prípon je založená na algoritme, ktorého úlohou je odstraňovanie predpôn a prípon na základe pravidiel, podľa ktorých sú konkrétne predpony a prípony generované alebo na základe vopred definovaného zoznamu predpôn a prípon. Táto metóda je silne závislá na použitom jazyku.

Slovník koreňov

Najväčšou prednosťou metódy izolácie koreňa slova pomocou využitia slovníka koreňov je minimálna chybovosť tejto metódy. Napriek tomu je použitie tejto metódy častokrát obmedzované rozsiahlosťou slovníka, jeho obmedzením len na špecifický odbor alebo faktom, že slovník je neúplny. V týchto slovníkoch sa nenachádzajú napr. špecifické podstatné mená ako vlastné mená, čo môže mať dopad na celkovú efektivitu metódy izolácie koreňa slova, keďže tieto slová sú dôležitými nositeľmi informácie vo vstupnom texte. Rovnako ako metóda izolácie koreňa slova pomocou odstraňovania predpôn a prípon, aj táto metóda je závislá na použitom jazyku.

Štatistická metóda

Metóda nezávislá na použitom jazyku. Jej podstatou je určovanie predpôn a prípon pomocou frekvencie po sebe nasledujúcich zhlukov znakov (grafém).

3.4.2 Český a slovenský jazyk

V jazykoch akými sú slovenčina alebo čeština sa lematizácia rieši metódami morfologickej analýzy a komplexným lingvistickým prístupom. Tieto metódy boli zvolené na základe komplikovaných pravidiel morfológie v takýchto jazykoch.

3.4.3 Anglický jazyk

Iný prístup k izolácii koreňa slova je využívaný v anglickom jazyku. V dôsledku jednoduchších morfologických pravidiel je izolácia koreňa slova pomerne jednoduchou záležitosťou. Najčastejšie táto transformácia prebieha na základe päťkrokového *Porterovho algoritmu.*

Porterov algoritmus

Porterov algoritmus je založený na metóde odstraňovania prípon (angl. *suffix stripping*) s využitím vopred definovaného slovníka prípon a niektorých pravidiel morfológie anglického jazyka. Pravidlá sú uvádzané v tvare:

(podmienka - logický výraz) slovo1 $\rightarrow$ *slovo2*

Podmienková časť operácií Porterovho algoritmu môže obsahovať nasledujúce premenné:

- *c* - spoluhlásky
- *v* - samohlásky
- *m* - miera slova (angl. *Measure of a Word*) - počet samohláskových skupín slova
- **S* - slovo končiace na spoluhlásku S (rovnako pre ostatné písmená)
- **v** - slovo obsahujúce samohlásku
- **d* - slovo končiace zdvojenou spoluhláskou (napr. -ss)
- **o* - slovo končiace `cvc`, teda spoluhláskou, samohláskou a spoluhláskou, pričom druhá spoluhláska nie je `w`, `x` alebo `y`

Porterov algoritmus je zložený z nasledujúcich krokov:

1. krok

   (a) Odstránenie prípon -s a -es:

sses → ss	caresses → caress
ies → i	ponies → poni
ss → ss	caress → caress
s →	cats → cat

   (b) Odstránenie prípon -d, -ed a -ing:

(m>0) eed → ee	feed → feed
	agreed → agree
(*v*) ed →	plastered → plaster
	bled → bled
(*v*) ing →	motoring → motor
	sing → sing

   Ak je úspešne aplikované druhé a tretie pravidlo kroku (b):

at → ate	conflat(ed) → conflate
bl → ble	troubl(ed) → trouble
iz → ize	siz(ed) → size
(*d ∧ ¬ (*L ∨ *S ∨ *Z)) → 1 písmeno	hopp(ing) → hop
	tann(ed) → tan
	fall(ing) → fall
	hiss(ing) → hiss
	fizz(ed) → fizz
(m=1 and *o) → e	fail(ing) → fail
	fil(ing) → file

   (c) Zmena prípony -y na -i:

(*v*) y → i	happy → happi
	sky → sky

2. krok

   Zmena prípon:

(m>0) ational → ate	relational → relate
(m>0) tional → tion	conditional → condition
	rational → rational
(m>0) enci → ence	valenci → valence
(m>0) anci → ance	hesitanci → hesitance
(m>0) izer → ize	digitizer → digitize
(m>0) abli → able	conformabli → conformable
(m>0) alli → al	radicalli → radical
(m>0) entli → ent	differentli → different
(m>0) eli → e	vileli → vile
(m>0) ousli → ous	analogousli → analogous
(m>0) ization → ize	vietnamization → vietnamize
(m>0) ation → ate	predication → predicate
(m>0) ator → ate	operator → operate
(m>0) alism → al	feudalism → feudal

(m>0) iveness → ive	decisiveness → decisive
(m>0) fullness → ful	hopefulness → hopeful
(m>0) ousness → ous	callousness → callous
(m>0) aliti → al	formaliti → formal
(m>0) iviti → ive	sensitiviti → sensitive
(m>0) biliti → ble	sensibiliti → sensible

3. krok
   Zmena alebo odstránenie prípon:

(m>0) icate → ic	triplicate → triplic
(m>0) ative →	formative → form
(m>0) alize → al	formalize → formal
(m>0) iciti → ic	electriciti → electric
(m>0) ical → ic	electrical → electric
(m>0) ful →	hopeful → hope
(m>0) ness →	goodness → good

4. krok
   Odstránenie prípon:

(m>1) al →	revival → reviv
(m>1) ance →	allowance → allow
(m>1) ence →	inference → infer
(m>1) er →	airliner → airlin
(m>1) is →	gyroscopic → gyroscop
(m>1) able →	adjustable → adjust
(m>1) ible →	defensible → defens
(m>1) ant →	irritant → irrit
(m>1) ement →	replacement → replac
(m>1) ment →	adjustment → adjust
(m>1) end →	dependent → depend
(m>1 ∧ (*S ∨ *T)) ion →	adoption → adopt
(m>1) ou →	homologou → homolog
(m>1) ism →	communism → commun
(m>1) ate →	activate → activ
(m>1) iti →	angulariti → angular
(m>1) ous →	homologous → homolog
(m>1) ive →	effective → effect
(m>1) ize →	bowdlerize → bowdler

5. krok

   (a) Úprava koreňov, odstránenie -e:

(m>1) e →	probate → probat
	rate → rate
(m=1 ∧ ¬ *o) e →	cease → ceas

   (b) Úprava koreňov, zmena -ll na -l:

(m > 1 ∧ *d ∧ *L) ll → l	controll → control
	roll → roll

3.5 Eliminácia stop-slov

Ústrednými nositeľmi informácie o obsahu vstupných textoch sú predovšetkým plnovýznamové slová. V dôsledku toho sú tokeny vzniknuté z takých slov transformované na termy, a sú priamymi nositeľmi informácie o obsahu textu. Na rozdiel od plnovýznamových slov majú neplnovýznamové slová len minimálny alebo nulový prínos k informácii o obsahu textu. Súhrnne sú tieto slová označované aj ako *stop-slová* (angl. *stop-words*). Hlavná úloha takýchto slov je v rovine syntaxe, kedy v textoch dotvárajú význam pre v okolí stojace plnovýznamové slová.

Vzhľadom k tomu, že v niektorých prípadoch je vhodné pre efektivitu systémov pre získavanie znalostí z textov aplikovať čo najvačšiu redukciu termov a aj k faktu, že stop-slová sú nositeľmi len minimálnej informácie, tak tokeny vzniknuté zo stop-slov sú vyraďované zo zoznamu termov. Preto je nevyhnutná čo najpresnejšia identifikácia stop-slova, pretože pri chybnej identifikácii (napr. označenia plnovýznamového slova ako stop-slova) môže prísť k strate dôležitej informácie.

Hlavným účelom odstraňovania stop-slov je teda redukcia počtu termov, ktoré reprezentujú dokumenty v textových kolekciách dát, a tým aj možné zvýšenie efektivity následných operácií získavania znalostí z textových dokumentov.

3.5.1 Spôsoby odstraňovania stop-slov

Odstraňovanie stop-slov môže prebiehať nasledujúcimi spôsobmi:

- použitím negatívneho slovníka
- automatické odstraňovanie slov s vysokou frekvenciou
- hybridná metóda

Eliminácia slovníkovou metódou

Negatívny slovník je manuálne vytvorený zoznam stop-slov a v dôsledku toho efektivita tejto metódy závisí na kvalite slovníka. Výhodou tohto spôsobu je jednoznačná identifikácia stop-slov vo vstupnom texte a ich následné odstránenie zo zoznamu termov. Táto metóda je však závislá na jazyku vstupného textu.
Nasledujúci zoznam obsahuje 30 najčastejšie používaných stop-slov v anglickom jazyku:

`I, a, about, an, are, as, at, be, by, for, from, how, in, is, it, of, on, or, that, the, there, this, to, was, what, when, where, who, will, with`

Automatické odstraňovanie slov s vysokou frekvenciou

Tento spôsob automaticky odstraňuje zo zoznamu termy, ktoré sa v textových kolekciách dát vyskytujú s vysokou, ale častokrát aj nízkou frekvenciou. Nie je však závislý na konkrétnom jazyku a je možné ho aplikovať pre väčšinu svetových jazykov. Napriek tomu efektivita tejto metódy nie je tak vysoká ako pri použití negatívneho slovníka.

Hybridná metóda

Táto metóda vznikla kombináciou predchádzajúcich dvoch metód. Jej snahou je dosiahnuť optimálne výsledky v takých prípadoch, keď použitie negatívneho slovníka alebo automatické odstraňovanie slov s vysokou frekvenciou nie je efektívne.

3.6 Váhovanie termov a reprezentácia dokumentov

Jednou z najdôležitejších častí predspracovania a transformácie textových kolekcií dát je aj vhodné *váhovanie termov* a následná *reprezentácia výsledkov*.

Všetky identifikované termy je nevyhnutné doplniť o dodatočnú vlastnosť, ktorá udáva dôležitosť daného termu v rámci konkrétneho dokumentu alebo v rámci všetkých dokumentov v textovej kolekcii dát. Váhovanie termov teda poskytuje dodatočnú vlastnosť, ktorá môže mať v konečom dôsledku dopad na celkovú efektivitu ďalších procesov získavania znalostí z textov.

Váhovanie termov je možné aplikovať dvoma spôsobmi:

- lokálne váhovanie termov
- globálne váhovanie termov

3.6.1 Lokálne váhovanie termov

Lokálna váha termu je špecifická pre konkrétny dokument. V praxi sa používa jednoduché binárne váhovanie, kedy je každému termu priradená vlastnosť, či sa v dokumente vyskytuje alebo nie (binárne označenie 0/1).

Častejšie sa však používa komplexnejšia *termová frekvencia*. Termová frekvencia (angl. *Term Frequency*, skr. *tf*) je váhovanie založené na počte výskytov termu v konkrétnom dokumente podľa vzťahu 3.1.

$$tf(w,d) = TermFreq(w,d) \tag{3.1}$$

kde `w` je term v dokumente a `d` predstavuje konkrétny dokument z textovej kolekcie dát.

3.6.2 Globálne váhovanie termov

Tento typ váhovania termov je rozšírením lokálneho váhovania, konkrétne termovej frekvencie. Globálne váhovanie vyjadruje dôležitosť daného termu v rámci celej textovej kolekcie dát.

Medzi spôsoby globálneho váhovania termov patria napr. metódy normovanie veľkosti vektora termu, entropia či v praxi najčastejšie využívaná termová frekvencia - inverzná dokumentová frekvencia.

Termová frekvencia - inverzná dokumentová frekvencia (angl. *Term Frequency – Inverse Document Frequency*, skr. *tf–idf*) je určená podľa nasledujúceho vzťahu 3.2.

$$tf\text{–}idf(w,d,N) = TermFreq(w,d) \times log\left(\frac{N}{DocFreq(w)}\right) \tag{3.2}$$

kde `TermFreq` je termová frekvencia (lokálne váhovanie termu v rámci dokumentu), `N` počet dokumentov v textovej kolekcii dát a `DocFreq(w)` je počet dokumentov, v ktorých sa vyskytuje term `w`.

Samotný počet výskytov termu v dokumente však nie je najvhodnejším reprezentantom váhy dôležitosti termu vzhľadom k obsahu dokumentu. Preto sa v praxi pri určovaní tf-idf nevyjadruje termová frekvencia (`TermFreq`) priamo, ale logaritmom tejto frekvencie podľa vzťahu 3.3.

$$TermFreq(w,d) = \begin{cases} 1 + log\big(tf(w,d)\big) & \text{ak } tf(w,d) > 0 \\ 0 & \text{ak } tf(w,d) = 0 \end{cases} \tag{3.3}$$

3.6.3 Reprezentácia dokumentov

Po fázi váhovania termov nasleduje finálna fáza predspracovania a transformácie textových kolekcií dát, ktorou je reprezentácia dokumentov.

V systémoch získavania znalostí z textových kolekcií dát sú dokumenty typicky reprezentované vo forme *vektora termov*. Ten reprezentuje dokumenty textovej kolekcie dát ako sekvenciu termov a ich váh.

Jednotlivé dokumenty sú reprezentované t-rozmerným vektorom, kde `t` zodpovedá počtu termov a zložky vektora sú tvorené váhami jednotlivých termov. Vektorová reprezentácia textovej kolekcie dát potom predstavuje t $\times$ n rozmernú maticu, kde `n` zodpovedá počtu dokumentov v textovej kolekcii dát.

Spôsoby reprezentácie dokumentov doplnené o príklady budú podrobnejšie uvedené v kapitole 4.11.

Kapitola 4

Implementácia

Hlavnou úlohou tejto kapitoly je podrobnejší popis samotnej implementácie aplikácie, kde bude rozobraný návrh jednotlivých modulov, ich komunikácia navzájom, a zároveň popis použitých knižníc. Dôležitou súčasťou je podkapitola 4.11, ktorá detailnejšie vysvetľuje spôsob reprezentácie výsledkov, teda výstup aplikácie.

4.1 Všeobecné ciele práce

Zadaním tejto práce je zoznámenie sa s textovými kolekciami dát a s problematikou dolovania textových dát so zameraním sa na ich predspracovanie a transformáciu.

Cieľom je návrh a vytvorenie aplikácie, ktorá bude implementovať jednotlivé metódy predspracovania a transformácie textových kolekcií dát pre webové dokumenty publikované v anglickom jazyku a bude poskytovať dosiahnuté výsledky.

V záverečnej časti je nevyhnutné pripraviť si vzorku dát (textové kolekcie) a následne na nej vytvorenú aplikáciu vhodne otestovať a overiť správnosť výsledkov.

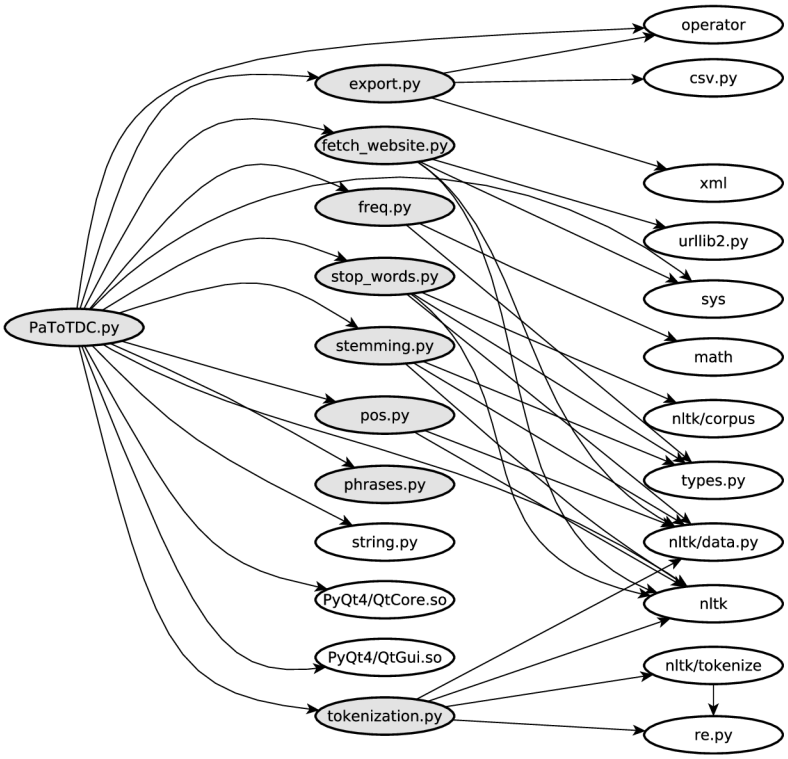
4.2 Návrh

Prvou fázou samotnej implementácie aplikácie bolo vytvorenie návrhu, t.j. *logickej štruktúry aplikácie*.

Logická štruktúra aplikácie predstavuje vhodné rozdelenie jednotlivých fáz predspracovania a transformácie textových kolekcií dát do modulov tak, aby mohli byť tieto moduly jednoducho aktualizované, príp. v budúcnosti dopĺňané o novú funkcionalitu. Zároveň bolo dôležitou úlohou vytvoriť internú štruktúru dát, pomocou ktorej budú spracovávané dáta predávané medzi jednotlivými modulmi aplikácie.

Zoznam modulov implementovaných v aplikácii doplnený o schému komunikácie je zobrazený na obrázku 4.1.

Súčasťou celkového návrhu aplikácie sa stal aj spôsob transformácie dosiahnutých výsledkov do serializačných formátov (kapitola 4.11).



Obrázok 4.1: Štruktúra aplikácie - schéma komunikácie modulov aplikácie.

4.3 Výber implementačného jazyka aplikácie

Ako implementačný jazyk aplikácie som zvolil interpretovaný, interaktívny programovací jazyk *Python*[10], konkrétne jeho verziu *Python 2*.

Medzi dôvody výberu tohto jazyka by som zaradil prácu s vysoko-úrovňovými údajovými typmi, rozšíriteľnosť a štruktúrovateľnosť jazyka. Zároveň umožňuje pri písaní programov používať nielen procedurálne, ale aj objektovo orientované paradigma a je dostupný pre väčšinu bežných platforiem (Unix, Windows, Mac OS).

Programovací jazyk Python je v oblasti získavania znalostí z textov pomerne často využívaný aj v praxi. Tento jazyk používajú pre implementáciu svojich nástrojov spoločnosti ako Google, Amazon, NASA a mnohé iné.

4.4 Modul Natural Language Toolkit

V rámci implementácie aplikácie bol využitý modul *Natural Language Toolkit* [5][9] (skr. *NLTK*). Modul NLTK vytvorený Stevenom Birdom, Edwardom Loperom, Ewanom Kleinom v roku 2001 predstavuje nástroj umožňujúci aplikovať základné operácie týkajúce sa spracovania reči. Tento nástroj je primárne zameraný na anglický jazyk, avšak mnohé operácie sú aplikovateľné aj na iné svetové jazyky, medzi ktoré môžme zaradiť slovenčinu a češtinu. Kedže celý modul je vedený ako projekt s otvoreným obsahom, v budúcnosti je možné očakávať zvýšenie podpory aj pre spomenuté stredoeurópske jazyky.

Pre výber tohto modulu som sa rozhodol na základe vyššie uvedených faktov. Modul NLTK implementuje efektívne operácie pre spracovanie textov v anglickom jazyku a zároveň nesie predpoklady pre budúce rozšírenie aplikácie pre ďalšie svetové jazyky.

Funkcie z modulu NLTK využité v rámci mojej aplikácie budú bližšie popísané v nasledujúcich podkapitolách.

4.5 Načítanie a konverzia webových elektronických dokumentov na čistý textový formát

Aplikácia umožňuje užívateľovi pracovať s elektronickými textovými dokumentami uloženými v prostredí webu, príp. webovými stránkami uloženými lokálne na počítači. Načítanie webových stránok zabezpečuje modul `fetch_website.py`.

Dôležitou fázou v procese predspracovania dát je konverzia načítanej webovej stránky na čistý textový formát. Práve pre tieto potreby bola využitá funkcia `nltk.clean_html()` z modulu NLTK. Tá odstraňuje formátovacie značky jazyka HTML, CSS a v neposlednom rade aj skripty, ktoré sú súčasťou načítanej webovej stránky (skripty v jazykoch JavaScript, Dart a pod.). Výstupom tejto funkcie je webová stránka transformovaná na čistý textový formát.

4.6 Tokenizácia

Implementáciu procesu tokenizácie tvorí modul `tokenization.py`, ktorý zo svojho vstupu, čistého textového formátu, vytvorí zoznam tokenov.

V tejto fázi je vstupný text najprv rozdelený na postupnosť viet a následne je každá veta rozdelená na menšie jednotky - tokeny. Takýto postup bol navrhnutý z toho dôvodu, že pri rozdelení vstupného textu na tokeny bez predchádzajúceho členenia na vety by mohlo dôjsť k strate dôležitej informácie, napr. informácie o poradí slov vo vetách. Táto strata by mala dopad na niektoré ďalšie procesy predspracovania a transformácie textových kolekcií dát, čím by bol negatívne ovplyvnený aj celkový proces a výsledky.

4.7 Part-Of-Speech značkovanie, identifikácia fráz

Hlavnou úlohou procesu Part-Of-Speech je značkovanie každého tokenu príslušnou značkou reprezentujúcou slovný druh tokenu. V rámci mojej implementácie je značkovanie postavené na štatistickom modeli založenom na klasifikácii podľa *maximálnej entropie* (skr. *MaxEnt*).

Model maximálnej entropie patrí medzi veľmi populárne modely, najmä v oblasti spracovania prirodzenej reči a jazyka. Vznikol trénovacím procesom s využitím anotovaného korpusu, pričom štandardným anotovaným korpusom pre trénovacie procesy sa stal dátový set projektu *Penn Treebank* Pennsylvánskej univerzity.

Jeho hlavnými zdrojmi sa stali predovšetkým články medzinárodného denníka Wall Street Journal (WSJ korpus), Brownov korpus či prepisy nahrávok telefonickej konverzácie (Switchboard korpus).

Podľa viacerých výskumov dosahuje model maximálnej entropie presnosť približne 97%, čo ho v porovnaní s ostanými modelmi radí v presnosti na prvé miesto.

Zoznam všetkých značiek POS[6] v projekte Penn Treebank doplnenými o popis a príklad obsahuje tabuľka 4.1.

Značkovanie Part-Of-Speech v aplikácii prevádza modul `pos.py` funkciou `nltk.pos_tag()`. Po aplikovaní značkovacej funkcie vzniká štruktúra reprezentovaná ako zoznam n-tíc (angl. *tuple*), teda dvojica prvkov (*token*, *značka*).

Značka	Slovný druh	Príklad
CC	spojka	and
CD	základná číslovka	1, third
DT	determiner	the
EX	there (existenciálne)	there (is)
FW	cudzie slovo	d'hoevre
IN	predložka alebo podraďovacia spojka	in, of, like
JJ	prídavné meno	green
JJR	prídavné meno, komparatív	greener
JJS	prídavné meno, superlatív	greenest
LS	značka položky zoznamu	1)
MD	modálne sloveso	could, will
NN	podstatné meno, singulár	table
NNS	podstatné meno, plurál	tables
NP	vlastné podstatné meno, singulár	John
NPS	vlastné podstatné meno, plurál	Vikings
PDT	predeterminer	both (the boys)
POS	privlastňovacia koncovka	(friend)'s
PP	osobné zámeno	I, he, it
PP$	privlastňovacie zámeno	my, his
RB	príslovka	however, usually, naturally, here, good
RBR	príslovka, komparatív	better
RBS	príslovka, superlatív	best
RP	častica	(give) up
SYM	symbol	%
TO	to	to (go), to (him)
UH	citoslovce	uhhuhhuhh
VB	sloveso	take
VBD	sloveso, minulý čas	took
VBG	sloveso, slovesné podstané meno	taking
VBN	sloveso, príčastie minulé	taken
VBP	sloveso, prítomý čas, nie 3. osoba, singulár	take
VBZ	sloveso, prítomý čas, 3. osoba, singulár	takes
WDT	Wh-determinant	which
WP	Wh-zámeno	who, what
WP$	privlastňovacie wh-zámeno	whose
WRB	Wh-príslovka	where, when

Tabuľka 4.1: Značky POS doplnené o príklad.

4.7.1 Identifikácia fráz

V neskoršej fázi predspracovania a transformácie textových kolekcií dát umožňuje značkovanie Part-Of-Speech identifikáciu fráz (angl. *phrase chunking*). Deje sa tak na základe aplikácie vzorky na zoznam n-tíc v module `phrases.py`. Vzorku tvoria pravidlá, ktoré si je možné predstaviť ako určitú podobu regulárneho výrazu.

Vzorku v mojej implementácii tvoria nasledovné pravidlá:

1. {<DT|PP$>?<JJ>*<NN>+} - determiner alebo privlastňovacie zámeno nasledované prídavnými menami a podstatným menom , napr. "the little dog".
2. {<NNP>+} - sekvencia vlastných podstatných mien, napr. "Barack Hussein Obama".
3. {<NN>+} - následnosť podstatných mien, napr. "money market fund".

4.8 Izolácia koreňa slova

Izolácia koreňa slova prebieha v module `stemming.py` a to na základe Porterovho algoritmu popísaného v kapitole 3.4.3.

4.9 Eliminácia stop-slov

Eliminácia stop-slov je realizovaná v module `stop-words.py` pomocou negatívneho slovníka dostupného z korpusu modulu NLTK (`nltk.corpus.stopwords.words('english')`).

4.10 Grafické užívateľské rozhranie

Grafické užívateľské rozhranie bolo implementované s využitím modulu *PyQt4*.

4.10.1 Modul PyQt4

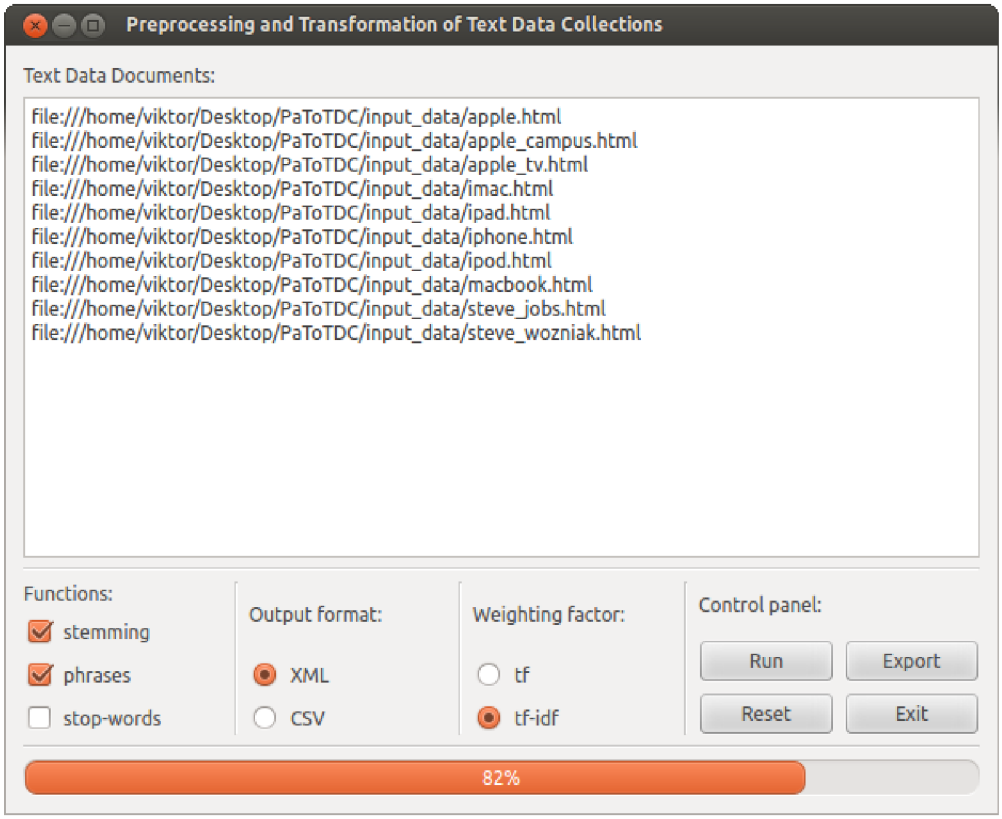
Modul PyQt4 je previazaním populárneho multiplatformného frameworku pre tvorbu grafických užívateľských rozhraní Qt a jazyka Python, vytvorený spoločnosťou Riverbank Computing Limited a licencovaný pod *GNU GPL*.

Pre modul PyQt4 sú v súčasnosti dostupné operačné systémy GNU/Linux, Windows a Mac OS X. Prednosťou tohto modulu v porovnaní s inými modulmi je jednotnosť vzhľadu na viacerých platformách a zároveň množstvo dostupných nástrojov, napr. pre prácu s XML.

4.10.2 Návrh grafického užívateľského rozhrania

Vzhľadom k tomu, že pri procesoch predspracovania a transformácie textových kolekcií dát je dôležitá interakcia s užívateľom, je k tomu prispôsobený aj návrh aplikácie, ktorý je zameraný predovšetkým na jednoduchosť, prehľadnosť a intuitívne ovládanie.

Implementácia grafického užívateľského rozhrania je dostupná module `PaToTDC.py`. Samotný návrh rozhrania, ktorý je možné upravovať v prostredí nástroja *Qt4 Designer* je uložený v súbore `gui.ui`. Náhľad hlavného okna aplikácie je zobrazený na obrázku 4.2.



Obrázok 4.2: Grafické užívateľské rozhranie aplikácie.

Hornú časť okna aplikácie tvorí textové pole, do ktorého užívateľ zadáva adresu zdrojovéhu súboru webového dokumentu. Pri zadaní adresy s protokolom *http* aplikácia umožňuje načítať webový dokument priamo z prostredia internetu, pri použití protokolu *file* z lokálneho počítača.

Spodná časť okna predstavuje ovládací panel aplikácie a zabezpečuje interakciu s užívateľom. Pomocou ovládacích prvkov je možné zvoliť, ktoré fázy predspracovania a transformácie textových kolekcií dát je potrebné previesť, je možné vybrať spôsob váhovania termov a zároveň je možné určiť formát v akom budú uložené výsledky. O priebehu procesu predspracovania informuje progress bar umiestnený v spodnej časti okna aplikácie.

4.11 Váhovanie termov a reprezentácia výsledkov

Aplikácia umožňuje dve techniky váhovania termov v textovej kolekcii dát, a to lokálne (tf) a globálne (tf-idf). Výpočty inverznej dokumentovej frekvencie a určovanie frekvencie termov v dokumentoch realizuje modul `freq.py`.

Export výsledkov prebieha v module `export.py`. Výsledky získané z váhovania termov je možné reprezentovať na základe výberu užívateľa vo formáte *XML* alebo *CSV*.

Spôsoby reprezentácie výsledkov ilustruje nasledujúci jednoduchý príklad (rovnaké výsledky (Tabuľka 4.2) zapísané vo formáte XML a CSV (Tabuľka 4.3)):

Dokument website1.html		Dokument website2.html	
term	frekvencia	term	frekvencia
world	30	car	11
travel	27	engine	10
holidays	15	world	8

Tabuľka 4.2: Výsledky získané z váhovania termov.

Formát XML

```
<?xml version="1.0" ?>
<data>
  <website url="./website1.html">
    <term freq="30">world</term>
    <term freq="27">travel</term>
    <term freq="15">holidays</term>
  </website>
  <website url="./website2.html">
    <term freq="11">car</term>
    <term freq="10">engine</term>
    <term freq="8">world</term>
  </website>
</data>
```

Formát CSV

```
world,travel,holidays,car,engine
30,27,15,0,0
8,0,0,11,10
```

Tabuľka 4.3: Reprezentácia výsledkov vo formáte XML a CSV.

4.12 Testovanie aplikácie a dosiahnuté výsledky

V priebehu vývoja bola aplikácia postupne testovaná na sade vlastných textových kolekciách dát, ktoré odhalili viaceré nepresnosti a chyby aplikácie. Na základe týchto testov sa podarilo všetky chyby úspešne odstrániť.

V záverečnom testovaní som vytvoril nové textové kolekcie dát založené na článkoch z *Wikipédie*. Na nich som pozoroval správanie aplikácie počas priebehu jednotlivých procesov predspracovania a transformácie textových kolekcií dát, pričom som porovnával textové kolekcie pred a po predspracovaní. Toto testovanie aplikácie mi pomohlo prakticky overiť interaktivitu procesu predspracovania a transformácie dát s užívateľom takým spôsobom, že som skúmal redukciu počtu termov pri použití jednotlivých metód predspracovania. Záverečné testovanie mi zároveň umožnilo uskutočniť niektoré optimalizácie vedúce k efektívnejšiemu dosiahnutiu výsledkov. Jednoduchý príklad textovej kolekcie dát určený na testovanie je uložený v priečinku `input_data`.

Aplikácia poskytuje zaujímavé výsledky a je dostupná pre plnohodnotné využitie v praxi. Vzhľadom k tomu, že je ovládateľná cez grafické užívateľské rozhranie, je možné ju využiť vo viacerých oblastiach tak, aby užívateľ svojou interaktivitou získal požadované výsledky. Štruktúrované výsledky uchované v serializačných formátoch XML a CSV dovoľujú prenositeľnosť výsledkov a ich následné využitie v iných nástrojoch získavania znalostí.

Kapitola 5

Záver

Hlavným cieľom tejto práce bolo navrhnúť a implementovať aplikáciu, ktorá predspracuje a transformuje textové kolekcie dát vo forme webových stránok v anglickom jazyku a bude poskytovať získané údaje.

Pre dosiahnutie tohto cieľa bolo potrebné si naštudovať danú problematiku. V počiatočnej fázi riešenia práce som sa podrobnejšie zoznámil s problematikou textových kolekcií dát, získavania znalostí z textových dát, pričom som sa zameral na ich predspracovanie a transformáciu. Zoznámil som sa s konkrétnymi postupmi a algoritmami, ktoré som v neskoršej fázi implementoval vo svojej aplikácii. V záverečnej fázi riešenia práce som otestoval aplikáciu na textových kolekciách dát. Z testov som dospel k záveru, že aplikácia poskytuje výsledky správne a prezentuje ich vhodným spôsobom.

Z hľadiska rozšírenia aplikácie v budúcnosti by som ako najperspektívnejšie riešenie videl rozšírenie aplikácie o podporu ďalších jazykov. V dôsledku toho bola aplikácia navrhnutá a implementovaná tak, aby takýto spôsob rozšírenia aplikácie bol možný a v pomerne jednoduchej miere zrealizovateľný.

Ďalším zaujímavým rozšírením aplikácie by mohlo byť rozšírenie podpory vstupných dokumentov o viaceré formáty, napr. formát *PDF*, *DOC*, *XML* prípadne iné.

Aplikácia bola implementovaná na aplikovanie procesov predspracovania a transformácie textových kolekcií dát na kolekciu webových stránok. Práve tu vo fázi konverzie webových stránok na čistý textový formát boli odhalené niektoré nedostatky aplikácie. V súčasnej dobe je možné na webových stránkach pozorovať veľkú rozmanitosť, či už z hľadiska ich tvorby alebo výberu použitých technológií a práve táto rozmanitosť priniesla do tvorby aplikácie viaceré komplikácie.

V neposlednom rade považujem za zaujímavé vyskúšať experimentálne za účelom zvýšenia kvality aplikácie aj iné metódy jednotlivých krokov predspracovania, ktoré boli bližšie popísané v kapitole 3.

Ako hlavný prínos práce považujem rozšírenie svojich osobných znalostí o viaceré poznatky z oblasti získavania znalostí z dát a textových dokumentov. Ako bolo spomenuté v kapitole 2.6, oblasť získavania znalostí z dát a textových dokumentov je využitá vo viacerých sférach, kde v budúcnosti môžem nadobudnuté poznatky úspešne využiť.

Literatúra

[1] Banke, J.: Dashlink is Online Home for Collaborative Research [online]. http://www.nasa.gov/topics/aeronautics/features/dashlink.html, 2010-08-09 [cit. 2013-04-14].

[2] Feldman, R.; Sanger, J.: *The Text Mining Handbook*. New York, NY, USA: Cambridge University Press, 2007, ISBN 978-0-521-83657-9.

[3] Gantz, J.; Reinsel, D.: Extracting Value from Chaos [online]. http://www.emc.com/collateral/analyst-reports/idc-extracting-value-from-chaos-ar.pdf, 2011 [cit. 2013-03-29].

[4] Kostoff, R. N.; Losiewicz, P.; Oard, D. W.: Science and Technology Text Mining: Basic Concepts [online]. http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ada415886, 2003 [cit. 2013-04-10].

[5] Madnani, N.: Getting Started on Natural Language Processing with Python [online]. http://www.desilinguist.org/pdf/crossroads.pdf, 2007 [cit. 2013-04-24].

[6] Marcus, M.; Marcinkiewicz, M. A.; Santorini, B.: Building a large annotated corpus of English: the penn treebank [online]. http://dl.acm.org/citation.cfm?id=972470.972475, 1993 [cit. 2013-04-24].

[7] Paralič, J.: *Objavovanie znalostí v databázach*. Košice: Elfa, 2003, ISBN 80-89066-60-7.

[8] Paralič, J.; Furdík, K.; Tutoky, G.; aj.: *Dolovanie znalostí z textov*. Košice: Equilibria, s.r.o., 2010, ISBN 978-80-89284-62-7.

[9] Perkins, J.: *Python Text Processing with NLTK 2.0 Cookbook*. Birmingham, UK: Packt Publishing Ltd, 2010, ISBN 978-1-849513-60-9.

[10] Pilgrim, M.: *Dive Into Python 3*. Berkely, CA, USA: Apress, 2009, ISBN 978-1-430224-15-0.

[11] Wikipedia: PubMed - Wikipedia, the free encyclopedia [online]. http://en.wikipedia.org/wiki/PubMed, 2013 [cit. 2013-04-09].

Príloha A

Obsah CD

- zdrojové súbory aplikácie: adresár `/src/`
- manuál k aplikácii: súbor `/src/MANUAL`
- jednoduchá textová kolekcia dát: adresár `/src/input_data/`
- zdrojové súbory písomnej technickej správy: adresár `/thesis/src/`
- písomná technická správa vo formáte PDF: súbor `/thesis/bp-xmarun01.pdf`