VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

AUTOMATICKÁ TVORBA REJSTŘÍKU PUBLIKACE

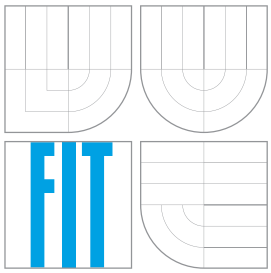
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

TOMÁŠ STRACHOTA

BRNO 2008

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

AUTOMATICKÁ TVORBA REJSTŘÍKU PUBLIKACE

AUTOMATIC CREATION OF THE PUBLICATION INDEX

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — TOMÁŠ STRACHOTA
AUTHOR

VEDOUCÍ PRÁCE — doc. RNDr. PAVEL SMRŽ, Ph.D.
SUPERVISOR

BRNO 2008

Abstrakt

Tato práce si klade za cíl prozkoumat možnosti běžných metod automatického zpracování jazyka a vytvořit prototyp systému, který bude schopen automaticky generovat rejstříky. Systém bude vyzkoušen na testovacích datech a na základě výsledků bude stanoven hlavní směr dalšího vývoje.

Klíčová slova

rejstřík, klíčová slova, vyhledávání klíčových slov, morfologická analýza, získávání informací

Abstract

The goal of this thesis is to survey potential of common language processing methods for text indexing. A prototype of automatic index-building system will be made and tested on gathered data. A direction for the next developement will be set based on the results of the tests.

Keywords

index, keywords, keyword search, morpholigical analysis, text mining

Citace

Tomáš Strachota: Automatická tvorba rejstříku publikace, bakalářská práce, Brno, FIT VUT v Brně, 2008

Automatická tvorba rejstříku publikace

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana doc. RNDr. Pavla Smrže, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Tomáš Strachota
12. května 2008

Poděkování

Za odborné vedení, poskytnutí cenných informací a pomoc při zpracování testovacích dat děkuji doc. RNDr. Pavlu Smržovi, Ph.D.

Obsah

1 Úvod **2**

2 Metody vyhledávání klíčových slov **4**
2.1 Četnosti 4
2.2 Backgroundový model 4
2.3 Směrodatná odchylka 5
2.4 Testování hypotéz 5

3 Morfologické analyzátory **6**
3.1 Ajka 6
3.2 PDT 2.0 6

4 Návrh systému **8**
4.1 Požadavky na rejstřík 8
4.2 Testovací data a jejich analýza 9
4.3 Vyhledávání klíčových slov 10

5 Implementace systému **13**
5.1 Morfologická analýza 13
5.2 Generování klíčových slov 13
5.3 Úprava tvarů klíčových slov 14

6 Výsledky **16**
6.1 Vyhodnocení výsledků 16
6.2 Úspěšnost úpravy výrazů 18
6.3 Přínos jednotlivých částí systému 19
6.4 Praktické použití systému 22

7 Závěr **24**

A PDT značky **26**

B Rejstřík vygenerovaný pro publikaci IZU - studijní opora **27**

Kapitola 1

Úvod

Rejstřík je bezesporu součástí každé kvalitní odborné publikace. Spousta knih, které jsou dostupné pouze v elektronické podobě rejstřík nemá, protože v nich lze poměrně snadno vyhledávat pomocí software pro prohlížení. Autor si tak značně ulehčí situaci. Čas, který by tvorbou rejstříku strávil, je jistě nemalý. Po vytištění se však všechny výhody elektronického dokumentu ztrácí a pokud se jedná o publikaci většího rozsahu, je vyhledávání v ní bez rejstříku takřka nemožné.

V dnešní době, kdy je při tvorbě software kladen velký důraz na uživatelskou přívětivost, je nabízena spousta pomocných nástrojů pro textové editory, jako například automatická oprava chyb nebo dokončování textu. Na poli tvorby rejstříků se však v běžně dostupných programech setkáme pouze s jednoduchými nástroji, které vygenerují rejstřík z nadpisů v dokumentu. Jedná se tedy spíše o jakousi formu obsahu. Pro běžného uživatele, který vytváří pouze středně rozsáhlé texty, je toto dostačující. Lze si ovšem jen stěží představit, že by tímto způsobem mohl být vytvořen kvalitní rejstřík, pro velkou vědeckou publikaci.

Pro vytvoření dobrého rejstříku je zapotřebí analyzovat text důkladněji. V českém jazyce je tato analýza komplikována mnoha tvary, ve kterých se můžou slova vyskytnout. Pro člověka to není problém, strojové zpracování to však komplikuje. Je třeba použít pokročilejších nástrojů, které dokážou upravit slova do jejich základních tvarů.

Cílem práce je prozkoumat možnosti běžných metod automatického zpracování jazyka a vytvořit prototyp systému, který bude schopný automaticky generovat rejstříky a vyhledávat vhodná klíčová slova přímo z textu. Práce by měla posloužit jako základ pro budoucí rozvoj generátoru. Měly by být nalezeny vhodné metody pro vyhledávání klíčových slov a určeno, co jsou hlavní nedostatky systému, kde se vyplatí zapracovat na vylepšení a co jsou „slepé uličky“. Budou prozkoumány a podle potřeby využity podpůrné nástroje pro analýzu českých textů.

V této práci budou hlavní cílovou skupinou generátoru texty studijních opor Fakulty Informačních Technologií VUT v Brně. Nástroj by měl tvůrcům opor co nejvíce ulehčit práci a studentům umožnit přístup k rejstříku ve studijních materiálech. Základy by však měly být dostatečně kvalitní, aby je po mírné úpravě bylo v budoucnu možno využít také pro vyhledávání klíčových slov za účelem indexace textů v jiných projektech.

Výsledky práce budou porovnány s rejstříky již existujících textů, kterým by se měly co nejvíce přiblížit. Kromě míry shodnosti s původním rejstříkem bude potřeba vyhodnotit také povahu ostatních vybraných slov. Nemělo by se jednat o obecné výrazy, ale o slova, která by případně šla do rejstříku zařadit. Tato část se bude muset provádět ruční kontrolou. Snahou bude pokud možno objektivní hodnocení, ovšem vliv autora na výsledky se v tomto místě nedá popřít. V práci bude na místa, kde existuje možnost zkreslení výsledků,

upozorněno.

Na závěr bude proveden test generátoru na studijní opoře k předmětu Základy umělé inteligence (IZU) a zhodnocení výsledků. K dané studijí opoře neexistuje rejstřík a proto bude možno provést simulaci reálného nasazení nástroje. Výstup generátoru bude pro dosažení opravdu kvalitního rejstříku pravděpodobně nutné upravit ručně. Snahou však bude finální úpravy rejstříku minimalizovat a nabídnout seznam kvalitních kandidátů na rejstříková hesla.

Kapitola 2

Metody vyhledávání klíčových slov

2.1 Četnosti

V následujících kapitolách shrnu běžně používané metody pro vyhledávání klíčových slov podle [5] a pokusím se nastínit alespoň jejich základní výhody a nevýhody.

Nejjednodušším způsobem jak vyhledat klíčová slova je na základě jejich četností v textu. Tato metoda funguje velice dobře pro kolokace, protože ustálená slovní spojení jsou v textech používána opakovaně. K selhání bohužel dojde při výběru unigramů, protože nejčastěji vyskytující se slova v českých textech jsou předložky, spojky a zájmena. Výběr nejčetnějších slov tedy v tomto případě není správnou cestou.

Pro generování unigramů se metoda stává zajímavou až při použití morfologického analyzátoru, který je schopný slovům přiřadit jejich druhy. Pokud odfiltrujeme nežádoucí slovní druhy (v případě rejstříku zřejmě vše, kromě podstatných a přídavných jmen), dá se očekávat, že slova s vyšší frekvencí výskytu budou vhodnými klíčovými slovy.

Tato metoda také není vhodná pro krátké texty, které nejsou dostatečně velkým vzorkem pro určení četností slov.

2.2 Backgroundový model

Backgroundový model je seznam slov a četností jejich výskytů v obecném textu. Principem použití pro výběr klíčových slov je výběr výrazů, které se v modelu vyskytují zřídka nebo vůbec. Takové slovo, nebo slovní spojení je tedy málo používané a tedy s vysokou pravděpodobností specifické pro zpracovávaný text.

Rozeznáváme dva druhy backgroundového modelu:

- korpusový (obecný)
- doménový (specializovaný)

Doménový model není založen na obecném textu, ale bere v úvahu oblast, o které zkoumaný text pojednává a snaží se vyhnout výběru slov, která jsou pro danou problematiku příliš obecná. Například v knize o savcích by se v rejstříku mělo objevit slovo pes, kdežto v knize o psích plemenech bude jako klíčové slovo jistě nezajímavé.

Backgroundový model se stává kvalitním nástrojem pro určování klíčových slov jen pokud je vytvořen z dostatečně velkého textu (řádově miliony slov). Je-li rozsah zdrojového textu nízký, nelze zaručit, že vybrané heslo není pouze zřídka se vyskytující obecné slovo.

2.3 Směrodatná odchylka

Kromě jednoduchého vybírání sousedních slov, jako kandidátů na slovní spojení, existují i jiné techniky, které umožňují určit relevantní spojení slov, mezi nimiž jsou „díry“. Nejjednodušší z těchto metod je určování směrodatné odchylky.

Spočítá se následujícím způsobem. Nejprve je nutné určit průměrnou vzdálenost slov ve větách, kterou získáme ze vztahu:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Samotná směrodatná odchylka se pak určí, jako:

$$\sigma^2 = \frac{\sum_{i=1}^{n}(d_i - \mu)^2}{n-1}$$

kde n je počet společných výskytů slov ve větách, d_i je vzájemná vzdálenost i-tého výskytu slov a μ je průměrná vzdálenost výskytu.

Blíží-li se hodnota směrodatné odchylky σ nule, slova se ve většině případů vyskytla blízko sebe, jedná se pravděpodobně o kolokaci.

2.4 Testování hypotéz

Přestože se slova v textu vyskytnou blízko sebe s vysokou frekvencí, může se jednat o pouhou náhodu a nemusí tvořit kolokaci. Abychom tuto možnost vyvrátili, používá se obvykle statistické testování hypotéz.

Stanovíme nulovou hypotézu H_0, která říká, že neexistuje spojení mezi slovy a vyskytují se spolu pouze náhodou. Následně spočítáme pravděpodobnost p s jakou událost nastane, pokud je H_0 pravdivá.

Pro slova, která jsou na sobě naprosto nezávislá, platí:

$$P(s_1 s_2) = P(s_1)P(s_2)$$

kde s_1 a s_2 jsou jednotlivá slova a P je pravděpodobnost. Nulovou hypotézu zamítneme, pokud je p příliš nízké.

Pravděpodobnosti se určují pomocí statistických testů, jako je například t test. Jinou možností je Pearsonův χ^2 test, který je považován za vhodnější, protože se narozdíl od t testu nezakládá na normálním rozložení, které ne zcela odpovídá povaze textových korpusů. Dokonalý přehled o statistickém testování poskytuje publikace [4].

Kapitola 3

Morfologické analyzátory

3.1 Ajka

Morfologický analyzátor Ajka vznikl v rámci diplomové práce Mgr. Radka Sedláčka [9] „Morfologický analyzátor češtiny“ v roce 1999 na Masarykově Univerzitě. Byl implementován na základě algoritmického popisu české formální morfologie. Ajka používá slovníkový přístup, takže veškerá data potřebná pro správnou funkci morfologického analyzátoru jsou uložena ve strojovém slovníku češtiny a v definičním souboru koncovkových množin a vzorů.

Formát strojového slovníku byl důmyslně navržen tak, aby byl uživateslky co možná nejjednodušší a při tom bylo možné data z něj dále používat pro jiné lingvistické experimenty. Jedná se o textový soubor, takže jeho editace je bezproblémová za pomoci běžných textových editorů.

Nejdůležitější součástí tohoto morfologického analyzátoru je definiční soubor koncovkových množin a vzorů. Obsahuje nejpodstatnější informace týkající se morfologie českých slov. Jedná se opět o textový soubor a jeho formát byl částečně převzat z [8].

Před spuštěním Ajky jsou data převedena k tomu určeným nástrojem do binárního tvaru, který je rychlejší k načtení. To umožňuje rychlejší start Ajky a efektivnější použití v cyklech.

Ajka disponuje poměrně širokou škálou nastavení ze strany uživatele.

3.2 PDT 2.0

PDT (Prague Dependency Treebank) [2] je projekt MFF Univerzity Karlovy, jehož cílem je ruční anotace českých textů bohatou lingvistickou informací. Obsahuje velké množství českých textů (2 milióny slov) s provázanými anotacemi na úrovni morfologie (2 milióny slov), povrchové syntaxe (1,5 miliónu slov) a hloubkové syntaxe a sémantiky (0,8 miliónu slov).

Součástí projektu jsou podpůrné softwarové nástroje pro prohledávání korpusu, anotaci dat a jazykovou analýzu. K dispozici je také rozsáhlá dokumentace.

Data v PDT 2.0 jsou anotována na třech rovinách: na morfologické rovině, analytické rovině a tektogramatické rovině. Ve skutečnosti existuje ještě jedna, neanotační rovina, reprezentující „surový text“. Na této rovině, zvané slovní rovina, je text rozdělen do dokumentů a odstavců. Jsou tu rozlišeny slovní jednotky (slova, čísla, interpunkce) a jsou opatřeny jednoznačnými identifikátory.

Hlavním formátem dat v PDT 2.0 je PML („Prague Markup Language“). Je to formát

založený na XML navržený pro reprezentaci bohaté lingvistické anotace textu, jako jsou morfologické značkování, závislotní stromy apod. Umožňuje mezi sebou propojit jednotlivé oddělené roviny anotace.

PML nahrazuje formát CSTS („Czech sentence tree structure“), který je založený na SGML a byl hlavním formátem dat v PDT 1.0. CSTS může reprezentovat jen morfologickou a analytickou anotaci (jeho definice obsahuje i několik elementů vztahujících se k tektogramatické anotaci, ale není schopen plného popisu této roviny). Ačkoliv autoři doporučují používat PML, některé starší nástroje stále výhradně používají CSTS.

Kapitola 4

Návrh systému

4.1 Požadavky na rejstřík

O tvorbě rejstříků pojednává Čsn norma ISO 999 [7]. Zabývá se jak volbou vhodných hesel pro rejstřík, tak jejich úpravou do vhodných tvarů a typografickým zpracováním. Norma definuje mnoho pojmů, z nichž pro další práci se nám budou hodit tři.

Definice 4.1. *Rejstřík* je abecedně, nebo jiným způsobem uspořádaná posloupnost hesel, jejíž způsob řazení se liší od zpracovávaného dokumentu nebo souboru, navržená tak, aby umožnila uživatelům najít informaci dokumentu nabo specifických dokumentů v souboru.

Definice 4.2. *Rejstříkové heslo* je jednotlivá položka v rejstříku. Skládá se ze záhlaví, z kvalifikátoru nebo poznámky o aplikaci, je-li potřeba; jednoho nebo více podzáhlaví, jsou-li potřeba; a buď lokátoru (tj. údaje o místě v dokumentu nebo fondu), nebo odkazu, případně obojího.

Definice 4.3. *Rejstříkové záhlaví* je termín zastupující v rejstříku prvek nebo pojem z dokumentu.

Podle normy je funkcí rejstříku sloužit jako efektivní prostředek k vyhledávání informací. Zpracovatel rejstříku tudíž musí:

- identifikovat a najít ve zpracovávaném dokumentu relevantní informaci
- odlišit podstatné informace o daném předmětu od zběžných zmínek
- analyzovat v dokumentu užívané pojmy a vytvořit z nich posloupnost záhlaví
- zajistit, aby uživatel byl schopen rychle zjistit, zda se v neznámém díle nachází nebo nenachází informace o určitém předmětu
- umožnit uživateli rychle vyhledat informace, které si zapamatoval po přečtení díla

Z těchto bodů vyplývá, že je potřeba rejstříková hesla volit střídmě a vyhnout se obecným pojmům. Rejstřík musí zároveň podat ucelený přehled o obsahu publikace. Dále norma promlouvá o kvalitě rejstříků, a to následujícím způsobem:

> Kvalitní rejstřík musí umožnit vyhledání informací obsažených ve zpracovávaných dokumentech.

> Hesla musí být pro uživatele přístupná všemi způsoby připadajícími v úvahu (např. román nebo hra, o nichž se pojednává v dokumentu, musí být zpracovány jak pod autorem, tak i pod názvem).

Je nutné se těchto doporučení držet a zohlednit je při návrhu systému a pochopitelně také při vyhodnocování výstupů.

Záhlaví reprezentují pojmy, které se nacházejí v dokumentu. Obecně se záhlaví skládají ze jmen, která jsou případně rozvita pomocí adjektiv nebo jiných slov ve funkci přívlastku. Předložky se v rejstříku užívají pouze tam, kde by jejich nepřítomnost mohla způsobit nejednoznačnost.

Objevuje-li se termín vybraný jako záhlaví v dokumentu v jednotném i množném čísle, je vybrána pouze jedna z těchto forem. V celém rejstříku se pak udržuje jednotný styl. Vyjímkou jsou případy, kdy každá z forem má svůj vlastní význam.

Rejstřík, aby vyhověl předpokládaným požadavkům jeho uživatelů, musí být dostatečně podrobný a musí postihnout všechna témata v dokumentu. Na druhou stranu uživatel systému pro automatickou tvorbu rejstříku nemůže být zahlcen přílišným množstvím hesel, které má projít a odstranit z nich ty nerelevantní. Pro dodržení požadavků normy a zároveň pro co největší uživatelskou přívětivost navrhovaného systému je nutné stanovit optimální rozsah generovaného rejstříku.

S ohledem na normu bylo rozhodnuto, že do rejstříku budou hesla ukládána jako základní tvary v jednotném čísle. Víceslovné výrazy budou přeskupeny tak, aby první bylo vždy podstatné jméno. Základní tvary bude vytvářet, popřípadě odhadovat morfologický analyzátor.

4.2 Testovací data a jejich analýza

Pro testování metod použitých k vyhledávání klíčových slov bylo použito celkem 5 publikací s již existujícím rejstříkem (viz tabulka 4.1). Byly zvoleny texty tematicky spadající do oboru informačních technologií. Tímto se blíží studijním oporám, na které má být generátor zaměřen nejvíce.

název publikace	přibližný počet slov	počet stran
SUSE aplikace	34 000	210
SUSE referenční příručka	110 500	548
TEXbook naruby	163 000	467
Linux - Dokumentační projekt	356 500	1020
Učebnice fyziky HRW	698 000	1278

Tabulka 4.1: Přehled testovacích publikací a jejich rozsahů.

Většina publikací měla strukturovaný rejstřík, který je nevhodný pro analýzu a porovnávání s výstupem generátoru. Proto byly nejprve všechny rejstříky ručně převedeny do klasické podoby a byly odstraněny duplicitní pojmy, které touto úpravou vznikly.

Rejstříky byly označkovány morfologickým taggerem PDT a byl proveden rozbor jejich složení. Všemi unigramy v rejstřících byla, podle očekávání, podstatná jména. Valná většina z nich (94.33 %) byla taggerem označena správně. V několika případech byla slova označena chybně jako jiný slovní druh, přestože se jednalo o podstatné jméno (např. X11[1] označeno

[1] X Window System (často zkráceně označovaný jen jako X11 nebo X) je síťový a zobrazovací protokol poskytující GUI (Graphical User Interface), založené na konceptu okna pro bitmapové displeje.

jako číslovka, Kopete[2] jako sloveso).

V případě bigramů bylo 72.29 % slov označeno jako podstatná jména a 26.37 % jako přídavná jména. Zbytek byl rozptýlen mezi ostatní slovní druhy.

Trigramy obsahovaly 67.05 % podstatných jmen a 26.63 % přídavných jmen. V malém množství zde do hry vstupují také spojky.

slovní druh		unigramy	bigramy	trigramy
podstatná jména	N	94.33 %	72.29 %	67.05 %
přídavná jména	A	2.08 %	26.37 %	26.63 %
číslovky	C	1.67 %	0.36 %	1.10 %
spojky	R	0.27 %	0.11 %	2.89 %
příslovce	D	0.48 %	0.24 %	0.99 %
částice	I	0 %	0.04 %	0 %
předložky	P	0.20 %	0.04 %	0.05 %
slovesa	V	0.48 %	0.22 %	0.68 %
neznámé slovní druhy	X	0.41 %	0.29 %	0.36 %

Tabulka 4.2: Zastoupení slovních druhů v rejstřících testovacích publikací (podle PDT).

Pokud bychom jako kandidáty na klíčová slova do rejstříku vybírali pouze majoritně zastoupené spovní druhy (podstatná a přídavná jména), došlo by ke ztrátě zhruba 6 % unigramů, 2 % bigramů a 6 % trigramů, ale množina vhodných kandidátů by se tak výrazně zúžila.

4.3 Vyhledávání klíčových slov

Po prostudování dostupné literatury a provedení mnoha pokusů s cvičnými texty jsem se rozhodl, že hesla do rejstříku budou generována pomocí kombinace několika technik zmíněných v kapitole 2 Metody vyhledávání klíčových slov.

Prvním krokem bude morfologické označkování celého textu. Pro tuto část systému byly vybrány nástroje z projektu PDT 2.0. Na projektu se stále pracuje a tak lze očekávat podporu i do budoucna. V současném návrhu se počítá pouze s využitím první, morfologické vrstvy PDT 2.0. Pokud by se to ukázalo jako vhodné, je do budoucna možné generátor rozšířit ještě o další vrstvy.

Z analýzy testovacích dat vyplývá, že je vhodné do rejstříku zařadit pouze podstatná a přídavná jména. Pokusem jsem ověřil, že výběr víceslovných hesel s „dírami“ zvětší množinu kandidátů asi o 50 %, avšak přírustek vhodných hesel je minimální. Stejně tak bylo zjištěno v [6]. Proto budou z řad víceslovných výrazů vybírána jenom slova, která spolu těsně sousedí v rámci jedné věty.

Ze vzorů (podle slovních druhů) vyskytujících se v textu budou akceptovány pouze ty v tabulce 4.3, kde N je podstatné jméno, A je přídavné jméno a X je neznámý slovní druh (viz příloha PDT značky).

Vybírání vzorů, kde je podstatné jméno na prvním místě, je podle mých pokusů možné ignorovat, protože se v českých textech vyskytují zřídka.

Dalším kritériem pro výběr hesla pro rejstřík bude jeho četnost v původním (zkoumaném) textu. Přestože se na první pohled zdá vhodné vybírat výrazy s nejvyšší četností,

[2]Kopete je multiprotokolový GNU-GPL software pro rychlou výměnu zpráv.

vzor	
N	X
AN	AX
NN	XX
AAN	AAX
ANN	AXX
NNN	XXX

Tabulka 4.3: Vzory pro výběr klíčových slov podle slovních druhů.

ukázalo se výhodnější stanovit hranici maximální četnosti a výběr určitým způsobem omezit. Vyhneme se tak výběru výrazů, které se v textu vyskytují často a jsou pro oblast (doménu), kterou se zkoumaný text zaobírá, příliš obecné.

Systém bude využívat také backgroundového modelu. Použijí se soubory četností slov, které byly vytvořeny z dostatečně objemného korpusu (cca 4GB čistého textu). Podmínky výběru jsou zde celkem jasné. Lépe ohodnoceny budou ty výrazy, které se v souborech s referenčními četnostmi budou vyskytovat s velmi nízkou hodnotou nebo se nebudou vyskytovat vůbec.

Posledním, pravděpodobně o něco méně důležitým, ukazatelem kvality klíčového slova, jsou shluky, ve kterých se v textu nacházelo. Toto kritérium vychází z úvahy, že v odborném textu budou jednotlivé kapitoly tematicky zaměřeny a hesla se v nich budou vyskytovat s větší hustotou. Shluky je případně také možné použít pro určení nejvhodnějšího místa v publikaci, kam se má heslo z rejstříku odkazovat.

Klíčová slova nelze z textu vybírat pouze na základě tohoto kritéria, avšak dokáže zvednout hodnocení těm relevantnějším ze seznamu slov, které byly vybrány jinými metodami. Rovněž pomáhá odstranit výrazy obecné pro danou doménu, protože takové výrazy se zřejmě budou vyskytovat rozptýleně po celém textu.

Byla stanovena hodnotící funkce, která určuje míru vhodnosti kandidáta pro rejstřík.

$$f = 0.5 * f_1 + 0.4 * f_2 + 0.1 * f_3$$

kde f_1 až f_3 jsou dílčí hodnotící funkce pro:

- f_1 ... četnost slov ve zkoumaném textu
- f_2 ... referenční četnost slov
- f_3 ... průměrnou sílu shluků

Obor hodnot dílčích hodnotících funkcí je $< 0, 1 >$. Každé z nich byla nastavena váha, kterou se podílí na výsledné hodnotící funkci. Váhy dílčích funkcí byly voleny tak, aby byl zachován obor hodnot $< 0, 1 >$ i pro výslednou funkci. Čím vyšší je hodnota funkce, tím je kandidát vhodnější pro zařazení do rejstříku.

Dílčí funkce byly zvoleny následovně

$$f_1 = e^{\frac{(x_1-\mu_1)^2}{2{\sigma_1}^2}}$$

kde x_1 je četnost slova ve zkoumaném dokumentu, μ_1 je střední hodnota, ke které by se svou četností měla nejlépe hodnocená slova blížit a σ_1 je rozptyl.

$$f_2 = e^{\frac{(x_2-\mu_2)^2}{2{\sigma_2}^2}}$$

kde x_2 je referenční četnost slova, μ_2 je střední hodnota, ke které by se svou četností měla nejlépe hodnocená slova blížit a σ_2 je rozptyl.

$$f_3 = e^{\frac{(x_3-\mu_3)^2}{2{\sigma_3}^2}}$$

kde x_3 je průměrný početshluků slova, μ_3 je střední hodnota, ke které by se měla nejlépe hodnocená slova blížit a σ_3 je rozptyl.

Pro μ a σ ve funkcích byly po sadě testů vybrány hodnoty z tabulky 4.4.

hodnotící funkce pro	μ_1	σ_1	μ_2	σ_2	μ_3	σ_3
unigramy	30	30	20	160	10	5
bigramy	100	100	0	1	10	5
trigramy	100	100	0	1	10	5

Tabulka 4.4: Hodnoty μ a σ pro jednotlivé hodnotící funkce.

Kapitola 5

Implementace systému

5.1 Morfologická analýza

Po spuštění generátoru musí nejdříve vstupní soubor projít morfologickou analýzou. O tu se starají podpůrné nástroje pro první vrstvu systému PDT 2.0. Jejich výstupem jsou morfologicky anotovaná data rozčleněná do vět a odstavců. Takto označená data jsou uložena do textového souboru ve formátu CSTS.

Pro zpracovávání CSTS souborů byl napsán parser, který umí rozpoznat začátky odstavců, vět a jednotlivá slova. U slov pak dále umí vybrat první nabízený základní tvar, poznámky k němu a ostatní určené morfologické kategorie podle [3].

Omezením nástrojů pro PDT 2.0 je pouze jedinné možné kódování vstupních souborů, a to ISO 8859-2. Pokud se v průběhu morfologické analýzy na vstupu vyskytne znak v jiném kódování (např. v UTF8), který není interpretovatelný, dojde k chybě a analýza je ukončena. Protože k takové situaci může v praxi dojít celkem snadno, byla implementována její detekce. V případě chyby není výstupní CSTS soubor ukončen. Morfologický analyzátor používá buffer, který se bohužel při chybě nevyprázdní a tak lze pouze přibližně určit oblast ve zdrojovém textu, která chybu způsobila. Uživatel je tak upozorněn na důvod, proč nemohlo být generování rejstříku dokončeno a je vypsána část textu, která předchází neinterpretovatelnému znaku.

5.2 Generování klíčových slov

Samotný generátor klíčových slov je rozdělen na generátor unigramů, bigramů a trigramů. To umožňuje jednoduše zasáhnout do mechanismu generování jednotlivé skupiny, aniž by byl ovlivněn zbytek systému. Do hlavní části programu je pak předán seznam ohodnocených kandidátů. Seznamy jsou spojeny, setřízeny a je vybráno N nejlepších kandidátů. Pokud není uživatelem zadáno jinak, automaticky se N vypočítá jako 0.7 % z počtu slov v textu. Tato hodnota byla určena na základě pokusů s hodnotami precision a recall na testovacích datech, při porovnávání nagenerovaných rejstříků s původními.

Kvůli zrychlení programu a snížení paměťových nároků bylo nutné upravit objemné soubory četností. Soubory byly označkovány morfologickým taggerem, výrazy převedeny do základních tvarů a odfiltrovány slovní druhy, které se při hledání kandidátů ignorují. Po tomto kroku došlo k výraznému zlepšení.

V testovacích výstupech se velice často objevovala krátká, jedno až dvoupísmenná, slova, která ve většině případů pocházela z příkladů v textech. Jednalo se například o označení

proměnných či neznámých v rovnicích. Pro jednoduchou eliminaci tohoto problému bylo implementováno použití stoplistu. V našem případě se jedná o soubor regulárních výrazů. Slova, nebo sousloví, která vyhovují alespoň jednomu z výrazů v souboru jsou z dalšího zpracování vypouštěna.

Dalším z problémů, které se v rejstřících objevily, byl výskyt podvýrazů. Například heslo „typ icmp datagramu“ a „typ icmp“ (Linux - Dokumentační projekt). V uvedeném případě bychom jednoznačně upřednostnili delší variantu, protože druhá nemá žádnou vypovídací hodnotu. Nabízelo by se výsledný rejstřík projít a odstranit výrazy, které jsou součástí jiných, delších výrazů. Vyskytují se však i hesla, kde by jednoznačně mělo dostat přednost kratší sousloví. Například „index lomu“ a „index lomu různý“ (Fyzika HRW). V některých případech by dokonce měly být do rejstříku zařazeny obě formy. Po uvážení jsem se proto rozhodl, že se žádné vyřazování podvýrazů provádět nebude a rozhodnutí, které z hesel se vyřadí, se ponechá na uživateli.

5.3 Úprava tvarů klíčových slov

Součástí systému pro generování rejstříků je také modul pro úpravu vyhledaných výrazů do výsledného tvaru. Původní předpoklad byl, že jako výsledný tvar se použije lemma slova, které vytvoří analyzátor PDT. V praxi se však ukázalo, že v případě cizích slov dochází k vyšší chybovosti (viz tabulka 5.1). Naneštěstí lze očekávat, že právě v rejstřících se budou cizí slova vyskytovat ve větší míře.

tvar v textu	systémem určený základní tvar	správný základní tvar
beagle	bígl	beagle
digikamu	digikaem	digikam
impress	impressa	impress
live	liv	live
suse	sus	suse

Tabulka 5.1: Příklady nesprávně určených základních tvarů slov.

Proti tomuto typu chyby lze poměrně jednoduše bojovat tím, že pokud jsou všechny původní tvary slova v textu stejné, použije se původní tvar. Pro cizí výrazy tato metoda funguje téměř bezchybně. Problém nastává v případě, když se čistě náhodou objeví v textu slovo několikrát pouze v jednom tvaru, a to v jiném, než základním. Ač by za normálních okolností byl určen správný základní tvar, použije se nesprávný tvar z textu. Tato metoda se v systému aplikuje pouze na úpravu podstatných jmen.

Přídavná jména lemmatizátor převádí do prvního pádu jednotného čísla rodu mužského. Pokud je přídavné jméno ženského nebo středního rodu, je koncovnka adekvátně upravena. Zde dochází k chybám, pokud je morfologickým analyzátorem špatně určen rod.

Všechna slova ve víceslovných výrazech jsou přeskupena tak, aby vždy první stálo podstatné jméno a následně za ním jména přídavná, jak je doporučováno v [7]. Podstatná jména jsou ponechána v původním pořadí. Pro přiblížení je postup naznačen v následujícím sche-

matu:

$$
\begin{aligned}
AN &\rightarrow NA \\
N_1N_2 &\rightarrow N_1N_2 \\
A_1A_2N &\rightarrow NA_2A_1 \\
AN_1N_2 &\rightarrow N_1N_2A \\
N_1N_2N_3 &\rightarrow N_1N_2N_3
\end{aligned}
$$

Kapitola 6

Výsledky

6.1 Vyhodnocení výsledků

Vygenerované rejstříky byly vyhodnocovány ze dvou pohledů. Prvním z nich bylo porovnání na shodnost s původním rejstříkem publikace. Tato část se prováděla automaticky porovnáním lemmat hesel z původního a referenčního rejstříku. Další částí vyhodnocení bylo ruční procházení rejstříků, kdy u každého hesla bylo subjektivně rozhodnuto, jestli by byl výraz pro rejstřík přípustný.

Výsledky první části hodnocení je možno vidět v tabulce 6.1. Při testu nebylo určeno, jaké množství hesel se má vyhledat, a proto se použila implicitní hodnota 0.7 % z textu. Jako ukazatel kvality výsledku byly použity hodnoty precision a recall, což jsou jedny z nejrozšířenějších metod určování kvality na poli získávání informací [1]. Vypočítají se následujícím způsobem:

$$\text{Precision} = \frac{|K_T|}{|K_A|}$$

$$\text{Recall} = \frac{|K_T|}{|K_R|}$$

Kde K_T je množina správně nalezených hesel, K_A je množina všech vygenerovaných hesel a K_R je množina referenčních hesel.

název publikace	počet hesel v původním rejstříku	počet vybraných hesel	počet správně vybraných hesel	precision	recall
SUSE aplikace	271	225	99	44.00 %	36.53 %
SUSE referenční příručka	607	756	144	19.05 %	23.72 %
TEXbook naruby	275	1 122	61	5.43 %	22.18 %
Linux - Dokumentační projekt	1 303	2 409	195	8.09 %	14.96 %
Učebnice fyziky HRW	1 979	4 821	541	11.22 %	27.33 %

Tabulka 6.1: Přehled výsledků generátoru bez použití stoplistu.

Nejlepších výsledků bylo dosaženo u publikace SUSE aplikace. Většina hesel v původním rejstříku jsou dosti specifická jména programů, což umožnilo jejich snadnější vyhledání.

Podobný charakter má i text SUSE referenční příručky.

Naopak o něco horší výsledek podal generátor u příručky TEXbook naruby a to paradoxně ze stejného důvodu. Příručka je plná TEXových příkazů, které jsou považovány za klíčová slova, ale do obecného rejstříku se příliš nehodí.

Učebnice fyziky HRW měla v původním rejstříku i dosti obecné pojmy, o kterých bylo už ze začátku jasné, že nebudou mít vysoké hodnocení. Ostatní pojmy byly však nalezeny poměrně úspěšně. Ve větší míře se mezi vyhledanými rejstříkovými hesly objevovala označení proměnných a neznámých z řešených příkladů v učebnici. Takové výrazy do rejstříku rozhodně nepatří a považuji je za chybu vážnějšího charakteru, protože by uživateli systému zbytečně znepřehledňovaly práci s vygenerovaným seznamem.

Ne úplně uspokojivého výstupu bylo dosaženo u publikace Linux - Dokumentační projekt. Čelní místa zaujaly výrazy z ukázkových skriptů, které se v publikaci vyskytují. Ke zvýšení jejich ohodnocení napomohlo i to, že jsou zapsány bez diakritiky a tudíž se nevyskytovaly v referenčním souboru četností.

název publikace	počet hesel v původním rejstříku	počet vybraných hesel	počet subjektivně správných hesel	precision
SUSE aplikace	271	225	134	51.55 %
SUSE referenční příručka	607	756	260	34.39 %
TEXbook naruby	275	1 122	100	8.91 %
Linux - Dokumentační projekt	1 303	2 409	440	17.39 %
Učebnice fyziky HRW	1 979	4 821	820	17.01 %

Tabulka 6.2: Přehled výsledků generátoru bez použití stoplistu, zahrnující subjektivně vhodné výrazy.

Při kontrole výstupů byly označeny všechny výrazy, které se zdály vhodné pro rejstřík, ale do původního rejstříku nebyly, ať už z jakéhokoliv důvodu, zařazeny. Přehled úspěšnosti zahrnující tyto výrazy se nachází v tabulce 6.2. Tabulka neobsahuje sloupec recall, protože není k dispozici žádný seznam původních slov a proto nemá smysl tuto hodnotu určovat.

```
.
..
.\ s.*
.*\ s.\ s.*
.*\ s.
..\ s.*
.*\ s..\ s.*
.*\ s..
.*[0-9\ -']+.*
```

Tabulka 6.3: Základní stoplist použitý při vyhodnocování výstupů generátoru.

Jak už bylo zmíněno v kapitole 5, k eliminaci nevhodných slov byla implementována funkce stoplistu. Pro vyhodnocení výsledků byl sestaven základní stoplist 6.3, který zakazuje výběr sousloví obsahujících alespoň jedno dvou, nebo jednopísmenné slovo a sou-

sloví obsahující číslovky, pomlčky a apostrofy. Volba regulárních výrazů vychází z toho, že většina označení proměnných z příkladů jsou jedno až dvoupísmenná, přičemž v původních rejstřících se tak krátkých slov vyskytuje minimum. Rovněž výrazy obsahující číslovky jsou s velkou pravděpodobností nevhodné.

název publikace	počet hesel v původním rejstříku	počet vybraných hesel	počet správně vybraných hesel	precision	recall
SUSE aplikace	271	225	104	46.22 %	38.37 %
SUSE referenční příručka	607	756	144	19.05 %	23.72 %
TEXbook naruby	275	1 122	68	6.06 %	24.73 %
Linux - Dokumentační projekt	1 303	2 409	197	8.17 %	15.11 %
Učebnice fyziky HRW	1 979	4 821	654	13.56 %	33.05 %

Tabulka 6.4: Přehled výsledků generátoru při použití stoplistu.

Přehled výstupů z programu při užití stoplistu je v tabulce 6.4. Na první pohled nezaznamenáme valné zlepšení. Maximální nárust hodnoty recall je o 5.72 % u Učebnice fyziky HRW. U ostatních publikací je nárust nízký (cca 2 %), nebo žádný.

Použití stoplistu se projeví až při subjektivnním hodnocení 6.5. Místo vytěsněných výrazů totiž zabrala jiná, relevantní, hesla. Nárust vhodných kandidátů se pohybuje mezi 10 % - 15 %, pouze u TEXbook naruby je nižší (cca 8.3 %). Z těchto výsledků tedy vyplývá, že stoplist je jednoznačně přínosem.

název publikace	počet hesel v původním rejstříku	počet vybraných hesel	počet subjektivně správných hesel	precision
SUSE aplikace	271	225	143	56.08 %
SUSE referenční příručka	607	756	263	34.79 %
TEXbook naruby	275	1 122	160	14.26 %
Linux - Dokumentační projekt	1 303	2 409	559	23.20 %
Učebnice fyziky HRW	1 979	4 821	1241	25.74 %

Tabulka 6.5: Přehled výsledků generátoru při použití stoplistu, zahrnující subjektivně vhodné výrazy.

6.2 Úspěšnost úpravy výrazů

Při úpravě vybraných rejstříkových hesel do náležitého tvaru docházelo ke čtyřem druhům chyb.

- První typ chyby byl způsoben morfologickým analyzátorem, který špatně určil mluvnické kategorie.
- Druhý typ chyby způsobovalo špatné určení základního tvaru slova.

- Třetí typ chyby byl do systému zavlečen pokusem o opravu předchozího závažnějšího typu. Je jím výběr tvaru z původního textu, pokud se vyskytuje jenom v jediném tvaru. Jelikož se tímto způsobem druhá chyba omezí v dostatečné míře, zavlečení nového druhu chyby se vyplatí.
- Čtvrtý druh chyby byl způsoben ponecháním podstatných jmen v základním tvaru (prvním pádě). U sousloví typu $N_1N_2N_3$ nebo N_1N_2A toto není vždy vhodnou volbou. Z textu, kde se hesla vyskytují v různých pádech, se však velice špatně určují pády vhodné pro rejstřík.

Souhrnný přehled úspěšnosti úprav výrazů najdeme v tabulce 6.6. Nejnižší dosažený výsledek je zhruba 94 %, což se dá považovat za poměrně vysokou úspěšnost.

název publikace	počet hesel ve správném tvaru	počet hesel	procentuelní vyjádření
SUSE aplikace	140	143	97.90 %
SUSE referenční příručka	260	263	98.86 %
TEXbook naruby	152	160	95.00 %
Linux - Dokumentační projekt	543	559	97.14 %
Učebnice fyziky HRW	1 168	1 241	94.12 %

Tabulka 6.6: Úspěšnost úpravy výrazů do rejstříkového tvaru.

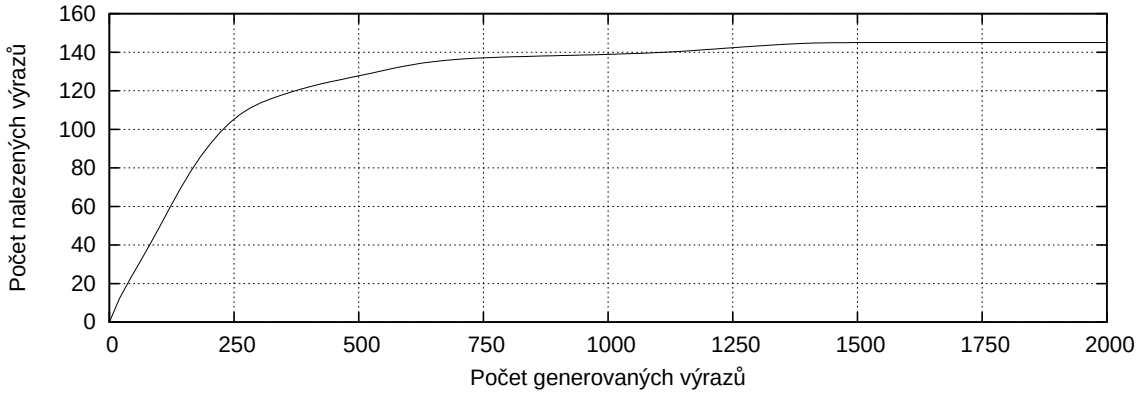
Z podrobného zastoupení druhů chyb 6.7 je vidět, že chybu 2 se opravdu podařilo minimalizovat. Největší podíl mají chyby 3 a 4. Jak už bylo řečeno výše, chyba 3 je méně závažnou. Největší nedostatek tedy spatřuji ve 4. druhu chyby, který by se v budoucnu vyplatilo omezit nejvíce.

typ chyby	procentuelní zastoupení chyby
1	14.43 %
2	10.31 %
3	35.05 %
4	40.21 %

Tabulka 6.7: Zastoupení jednotlivých druhů chyb úpravy výrazů.

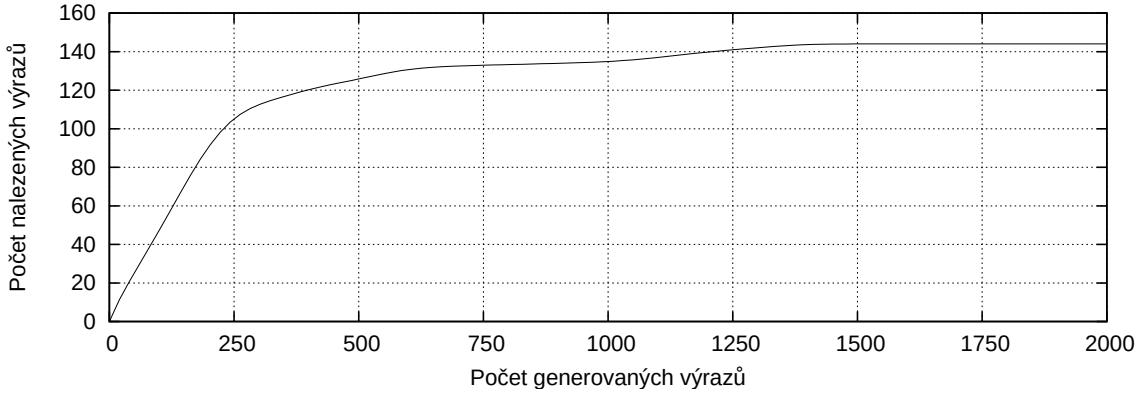
6.3 Přínos jednotlivých částí systému

Vytvořený systém je složen z několika částí, které se podílí na výběru kandidátních klíčových slov. Abych ověřil, že mají všechny v systému své místo oprávněně, provedl jsem pokusy, při nichž byly jednotlivé části postupně vypouštěny. Pro každou konfiguraci byl zaznamenán graf závislosti množství odhalených rejstříkových hesel v závislosti na počtu všech vybraných výrazů. Na základě výsledků těchto testů bude možno v budoucnu také určit, do kterých částí systému se vyplatí investovat úsilí a které naopak přinášejí minimální zisk.



Obrázek 6.1: Graf závislosti počtu správných nalezených výrazů na počtu generovaných výrazů.

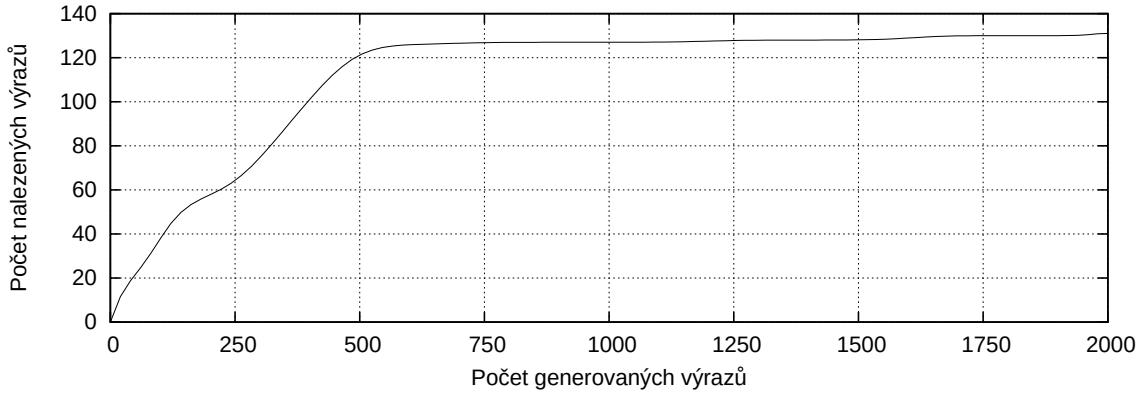
Testy byly provedeny s publikací SUSE aplikace, u které generátor dosáhl nejlepších výsledků. Původní rejstřík měl délku 271 hesel. Vždy bylo postupně generováno až 2 000 výrazů. Nejprve byl sestaven graf kompletního systému pro pozdější porovnání s výsledky nekompletních konfigurací. Z grafu 6.1 vidíme, že až do 250 vybíraných výrazů roste křivka poměrně strmě a na každých 100 vybraných výrazů získáme asi 40 správných. Pak dochází ke zlomu a přírustek je nižší až do 700. Dále už následuje jen mizivý nárust.



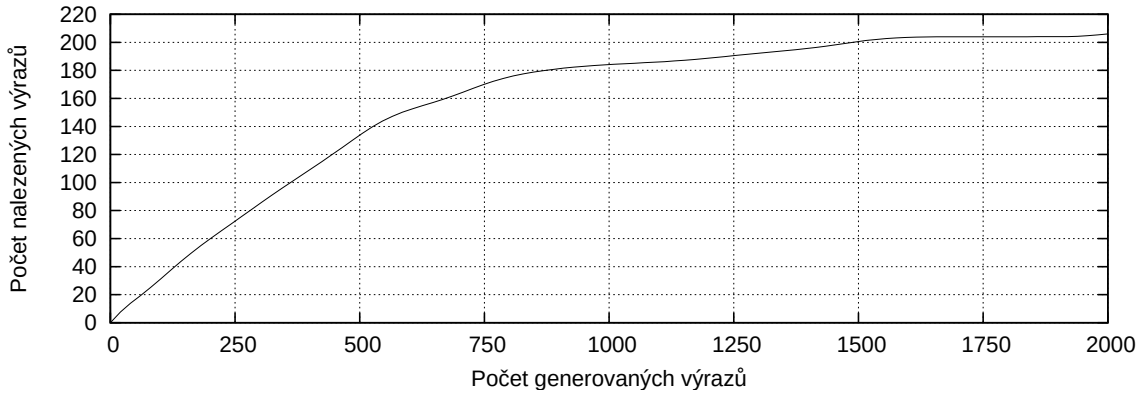
Obrázek 6.2: Graf závislosti počtu správných nalezených výrazů na počtu generovaných výrazů pro systém bez využití shluků výrazů.

Jako první bylo vypuštěno počítání shluků. Jak už bylo naznačeno dříve v této práci, nedá se očekávat, že by toto nějak drasticky zhoršilo výsledky. Graf 6.2 má téměř stejný průběh, jako původní graf. Změnu zaznamenáme až v druhé (hodnoty 750 - 1 200) méně strmé části, kde je bez použití shluků pokles asi o 5 správných výrazů. To je dáno charakterem funkce počítání shluků v systému, jehož úkolem není přímo vyhledávat vhodná slova, ale pouze je posouvat o něco výše v hodnocení.

Dalším v pořadí bylo vynechání počítání četností slov v publikaci. Zde už dochází ke znatelné změně. Při 500 vybíraných výrazech se sice dostaneme k množství cca 120 správně vybraných výrazů, což je velice blízko výsledkům kompletního systému, ale křivka grafu 6.3



Obrázek 6.3: Graf závislosti počtu správných nalezených výrazů na počtu generovaných výrazů pro systém bez četností v dokumentu.



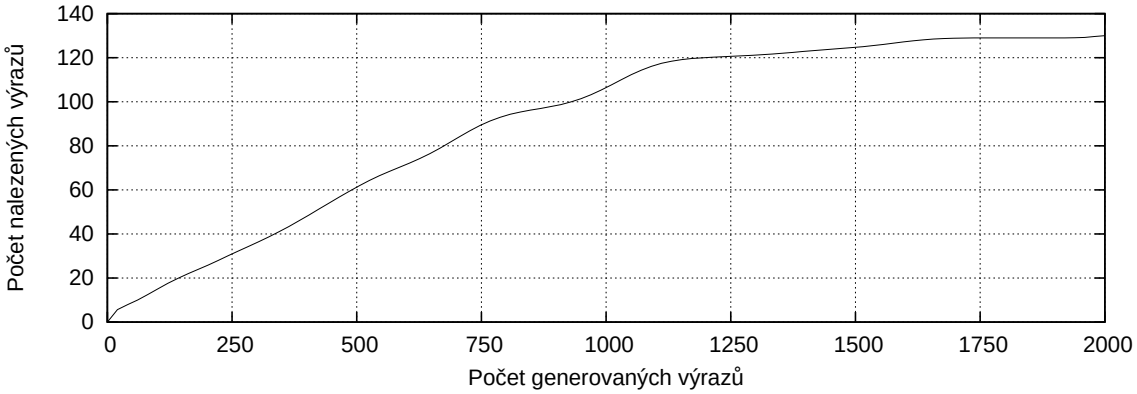
Obrázek 6.4: Graf závislosti počtu správných nalezených výrazů na počtu generovaných výrazů pro systém bez referenčních četností.

má jiný průběh. Z toho vyplývá, že bez četností z textu jsou sice vybrána vhodná slova, ale nejsou v hodnocení posunuta dostatečně dopředu oproti ostatním.

Jednou z posledních částí systému je backgroundový model s referenčními četnostmi. Z grafu 6.4 je vidět, že je to velice důležitá komponenta. Celkově bylo sice nalezeno víc správných rejstříkových hesel, protože hesla obecnějšího charakteru nebyla odsunuta na zadní pozice, nicméně množství nevhodných slov je při jejím vypuštění značné.

Graf systému bez využití morfologické analýzy 6.5 je uveden spíše pro ukázku, neboť je zřejmé, že se jedná o klíčovou součást systému. Množství generovaného balastu je v tomto případě již neúnosné. Kompletní systém najde 120 správných výrazů už při výběru 400 kandidátů. Bez morfologické analýzy se na tuto hodnotu dostáváme až při 1 200 kandidátech.

Z testu tedy vyplývá, že kromě morfologického analyzátoru je opravdu důležitou komponentou generátoru model s referenčními četnostmi. Má největší třídící sílu a systém by tedy měl disponovat kvalitním modelem. Naopak postradatelnou částí je vyhledávání shluků slov. Jeho vypuštěním se výsledky systému nijak výrazně nezhorší. Má ovšem potenciál k tomu, aby byl v budoucnu využit pro lokalizaci nejvhodnějšího místa v dokumentu, kam by se mělo rejstříkové heslo odkazovat.



Obrázek 6.5: Graf závislosti počtu správných nalezených výrazů na počtu generovaných výrazů pro systém bez využití morfologické analýzy.

6.4 Praktické použití systému

Pro důkladné zhodnocení výsledků na závěr jsem se rozhodl vyzkoušet generátor rejstříků v praxi. Byl sestaven rejstřík studijní opory k předmětu Základy umělé inteligence (IZU). Opora má cca 140 stran (zhruba 43 500 slov). Při sestavování jsem postupoval tak, jak by asi postupoval případný uživatel generátoru. Tak by měly dobře vyplynout základní nedostatnky a omezení systému. Předpokládá se, že člověk sestavující rejstřík má alespoň základní přehled o tématu, kterého se zpracovávaná publikace dotýká.

První věc, kterou uživatel pravděpodobně udělá, je testovací spuštění systému. Generátor byl tedy spuštěn bez jakýchkoliv zvláštních parametrů. Seznam kandidátů na výstupu obsahoval 330 hesel, což je množství automaticky určené generátorem. Na první pohled bylo patrné, že seznam obsahuje příliš mnoho krátkých slov pocházejících z příkladů a vzorců v opoře. Dalším logickým krokem bylo tedy zbavit se těchto pro rejstřík nevýznamných výrazů. Nejjednodušším způsobem je použití stoplistu. Následující generování bylo filtrováno již sestaveným stoplistem 6.3 z kapitoly Vyhodnocení výsledků.

Seznam návrhů jsme sice zbavili mnoha nevhodných slov, ale ukázalo se, že je příliš krátký. Proto bylo celé generování opakováno ještě jednou. Tentokrát byla explicitně zadána délka nabízeného seznamu. Jelikož implicitní hodnota je 0.7 %, zvolil jsem délku 2 %. Po této změně měl výstup délku 810 hesel, což se zdálo dostatečné a bylo možné přistoupit k jeho kontrole.

Při třídění nabídky klíčových slov jsem kromě hesel nabídnutých přímo sledoval také pojmy, které nelze do rejstříku zařadit v nabízené podobě, ale navádí člověka k zařazení jiných pojmů. I takovéto asociace mírně zjednodušují práci. Příklad takových hesel je možné vidět v tabulce 6.8. Do statistik byla taková slova zařazena pouze v případě, že asociované heslo nebylo do seznamu vloženo už generátorem.

Kontorla rejstříku ukázala, že 119 kandidátů je použitelných a 14 kandidátů [1] navádí na jiná hesla vhodná pro zařazení. Po zběžném protřídění rejstříku, které bylo hotovo asi za 15 minut, jsme tedy získali 133 kvalitních rejstříkových hesel.

Poměrně velkou část z nevhodně nabízených slov tvořila slova, která pocházela z různých příkladů v učebnici. Nalézaly se zde například příkazy jazyka LISP a PROLOG. Česká slova

[1] Jedná se o hodnotu do značné míry ovlivněnou subjektivním pohledem a mírou znalosti problematiky.

nabízené heslo	asociované heslo
climbing	hill climbing
constraint	constraint satisfaction problem
džbán metodou bfs	metoda dvou džbánů
metoda greedy	metoda greedy search

Tabulka 6.8: Příklad nevhodných hesel, které ovšem navádějí uživatele k jiným heslům.

zapsaná v útržcích zdrojových kódů bez diakritiky získala taktéž vyšší hodnocení, než by si zasluhovala a tak třeba i takové pojmy jako „jiri“ nebo „pocitat“ byly vybrány, jako vhodné pro rejstřík.

hesla	počet výskytů	%
vhodná pro rejstřík	119	14.69
vedoucí k vhodným pro rejstřík	14	1.73
nevhodná z příkladů	189	23.33
ostatní nevhodná	488	60.25

Tabulka 6.9: Počty nalezených a vyřazených hesel z rejstříku studijí opory k předmětu IZU (celkový počet hesel je 810).

Kompletní přehled 6.9 ukazuje, že slova tohoto charakteru zabrala asi 23 % z celého nabízeného seznamu. Proto byl proveden celý experiment ještě jednou, tentokrát s textem, ze kterého byly příklady odstraněny.

Text opory bez příkladů se výrazně zkrátil (cca 100 stran, 30 500 slov). Aby bylo dosaženo alespoň základní podrobnosti rejstříku, byl generátor spuštěn s parametrem 2 %, což vedlo k výstupu o délce 606 hesel. Seznam je tedy zhruba o 200 pojmů kratší. Přesto došlo ke zlepšení. 128 pojmů se zdálo být pro rejstřík vhodných, dalších 21 upozornilo na jiné, chybějící. V souhrnu tedy o 12 více, než při předchozím testu. Nevhodná slova pocházející z příkladů se nepodařilo odstranit úplně, protože na příklady a jejich výsledky je často odkazováno přímo z textu. Tyto zmínky není možné odstranit, protože by byla poškozena smysluplnost textu. Proto v nabízeném rejstříku přetrvalo 21 nedůležitých slov z příkladů. Tuto hodnotu považuji za výrazné zlepšení.

Ačkoliv byl vygenerovaný seznam kratší, zabralo jeho zpracování delší dobu. Kontrola trvala asi 25 minut. Důvodem je větší množství relevantnějších hesel, nad nimiž je třeba se zamyslet a zvážit jejich případné vyřazení. Zdánlivé prodloužení je tedy přínosem.

hesla	počet výskytů	%
vhodná pro rejstřík	128	21.12
vedoucí k vhodným pro rejstřík	21	3.30
nevhodná z příkladů	36	5.94
ostatní nevhodná	421	69.64

Tabulka 6.10: Počty nalezených a vyřazených hesel z rejstříku studijí opory k předmětu IZU po odstranění příkladů (celkový počet hesel je 606).

Kapitola 7

Závěr

Cílem práce bylo prozkoumat možnosti běžných metod automatického zpracování jazyka a vytvořit generátor rejstříků publikací. Na základě vlastních pokusů a výsledků z [6] byly vybrány vhodné metody pro selekci klíčových slov a určeny poměry, jakými se budou podílet na výsledném ohodnocení kvality kandidátního slova. Pro prvotní rozdělení textu podle slovních druhů a určení základních tvarů slov byl použit systém PDT 2.0, který má potenciál pro budoucí rozšiřování generátoru. Výrazy z textu jsou dále vybírány na základě jejich četnosti, referenční četnosti z backgroundového modelu a průměrné síly shluků, které tvoří. Celý systém byl implementován v jazyce Python.

Testy systému ukázaly, že většina odborných nebo cizojazyčných pojmů je s velkou mírou úspěchu nalezena. Problematické zůstávají pojmy obecnějšího charakteru, které se běžně vyskytují v textech s vyšší četností a proto nejsou systémem vyhodnoceny jako vhodné. Tato slova by mohla získat lepší hodnocení při použití doménového modelu. Další zajímavou možností, kterou by v budoucnu bylo možné prozkoumat, je dohledávání korpusových textů na www stránkách podle skupiny klíčových slov, která by byla zadána před začátkem generování. Tyto texty by pak posloužily jako aktuální zdroje pro počítání referenčních četností slov.

Přestože norma [7] doporučuje vyhnout se v rejstříku jiným slovním druhům než přídavným a podstatným jménům, některé výrazy je nutno do rejstříku zařadit, i když tuto podmínku nesplňují. V současném stavu generátor tato sousloví ignoruje. Jedním z dalších rozšíření vyhledávacích metod by mohl být odhad zastoupení jednotlivých slovních druhů v N-gramech, podle kterého by se určil poměr, v jakém vůči sobě budou vybírány.

Nejpalčivějším problémem se v tuto chvíli však zdají být výrazy z příkladů či skriptů v publikacích. Mnoho z nich je zapsáno bez diakritiky, což zvyšuje jejich hodnocení, protože backgroundový model je nezná. Pokud by se systém omezil pouze na práci s LaTeXovými dokumenty, bylo by možné specifikovat, které sekce se mají vypouštět a které ponechat. Zároveň by bylo možné mírně zvýšit hodnocení slovům, která se vyskytují v nadpisech. Dále by se naskytla možnost do dokumentu přímo vkládat rejstříkové odkazy. Zde vidím největší prostor pro vylepšení a další rozvoj systému.

Kromě posledního jmenovaného problému by se další práce měly zaměřit také na úpravu rejstříkových hesel do vhodných základních tvarů a to především u víceslovných výrazů, kde systém v tuto chvíli spoléhá na určité pořadí slov. Pro toto vylepšení by bylo možné využít dalších vrstev PDT 2.0, které určují závislosti slov ve větách.

Literatura

[1] Wikipedia. [online], [cit. 2008-04-19].
URL <http://en.wikipedia.org/wiki/Precision_and_recall>

[2] HAJIČ, J.; HAJIČOVÁ, E.; aj.: *Průvodce PDT 2.0*. 2006.

[3] HANA, J.; ZEMAN, D.: *Manual for Morphological Annotation: Revision for the Prague Dependency Treebank 2.0*. 2005.

[4] HENDL, J.: *Přehled statistických metod zpracování dat: analýza a metaanalýza dat*. Portál, 2004, ISBN 80-7178-820-1.

[5] MANNING, C.; SCHÜTZE, H.: *Foundations of Statistical Natural Language Processing*. The MIT Press, 1999, ISBN 0-262-13360-1.

[6] MAŠLÁŇOVÁ, M.: *Automatická identifikace klíčových slov*. Diplomová práce, Vysoké učení technické v Brně, Fakulta informačních technologií, 2007.

[7] Čsn norma: *Čsn iso 999-1998 zásady zpracování uspořádání a grafické upravy rejstříků*. 1998.

[8] OSOLSOBĚ, K.: *Algoritmický popis české formální morfologie a strojový slovník češtiny*. 1996.

[9] SEDLÁČEK, R.: *Morfologický analyzátor češtiny*. Diplomová práce, Masarykova univerzita, Fakulta informatiky, 1999.

Dodatek A

PDT značky

značka	slovní druh/morfologická kategorie
N	podstatné jméno
A	přídavné jméno
P	zájmeno
C	číslovka
V	sloveso
R	předložka
J	spojka
I	částice
T	citoslovce
X	neznámý slovní druh

Tabulka A.1: Přehled PDT značek pro kategorie slov.

Dodatek B

Rejstřík vygenerovaný pro publikaci IZU - studijní opora

ADP
ADP learning
ADS
alfa řez
alfabeta
algoritmus BFS
algoritmus decision tree
algoritmus DFS
algoritmus DLS
algoritmus general to specific search
algoritmus genetický
algoritmus minimax
algoritmus prohledávací
algoritmus prohledávací základní
algoritmus UCS
backtracking
backtracking for CSP
best first search beta řez
BFS
branching factor
hill climbing
clustering
constraint satisfaction problems
CSP
depth
DFS
dichotonomie
DLS
dokazování nesplnitelnosti množiny
Eliza
expectimax
expectimin
expectminimax
faktor větvení
forma normální prenexní
formule atomická
formule prvotní
forvard checking
funkce DFS
funkce diskriminační
funkce heuristická
funkce ohodnocující
futility
goal
GPS
graf pojmový
greedy search
heuristika
hlavolam kryptoaritmetický
hodnota futility
hodnota pravdivostní
hraní her
hry jednoduché
hry složité
IDS
individuová proměnná
inteligence umělá
intelligence artificial
interpret Lisp
interpretace formule
jazyk Lisp
jazyk Prolog
JPL
klasifikátor

klauzule Hornova
kvantifikátor
kvantifikátor existenční
Lisp
literál
logika predikátová
K means clustering
metoda ADP
metoda ADP learning
metoda backtracking
metoda backtracking for CSP
metoda BFS
metoda DFS
metoda DLS
metoda forvard checking
metoda greedy search
metoda IDS
metoda rezoluční
metoda UCS
metody hraní her metody informované
metody neinformované
metody řešení úloh
metody prohledávací
minimax
množina trénovací
modifikaci znalostí
modus ponens
Mycin
navracení zpětné
PDL
pravidlo odvozovací
princip her
princip metod učení
procedura alfabeta
procedura forward checking
procedura minimax
procedura unify
prohledávání lokální
prohledávání obousměrné
prohledávání prostoru verzí
Prolog
proměnná vázaná
prostor obrazový
prostor stavový
prostor úlohy stavový
reinforcement learning
rezoluce základní
rezolventa
rozpoznávání statistické
rozpoznávání strukturální
rozpoznávání syntaktické
schéma procedurální
schéma reprezentace
schéma reprezentace logické
schéma reprezentace procedurální
schéma reprezentace síťové
schéma reprezentace znalostí
schéma strukturální
schéma síťové
best first search
Skolemova funkce
softcomputing
strategie lineární
Strips
strom rozhodovací
substituce procedurou
síť neuronová
síť sémantická
tautologie
template matching
traveling salesman problem
UCS
unifikátor
uzávěr formule
učení posilované
učení strojové
vektor vstupní
vektor výstupní
zanořování postupné
žíhání simulované
řešitelnost