Posudek vedoucího diplomové práce

Radoslav ZÁPOTOCKÝ

Shlukování textových dokumentů a jejich částí

Cílem této práce bylo navrhnout a implementovat systém, který by dovolil – na rozdíl od běžných indexačních systémů – analyzovat nikoli kolekci samostatných dokumentů, ale jeden dokument po částech s různou granularitou (například na úrovni kapitol, odstavců, případně vět), vytvářet vektory pro tyto části a ty následně shlukovat, případně obdobně zpracovávat a výsledky dále analyzovat. Na rozdíl od jiných prací byla tato práce zaměřena více „experimentálně" a přesný obsah a zadání se upravovalo v průběhu práce.

Jak se ukazuje, program, přiložený na CD ve své standardní 32-bitové verzi u velmi dlouhých dokumentů s desetitisíci položek (kapitol/odstavců) naráží na 2GB omezení, dané paměťovým modelem. Pokud se však vyjde z přiloženého projektu pro MS Visual studio, je možné provést překlad pro 64-bitovou architekturu, který tato omezení nemá a program je potom omezen výhradně kapacitou dostupné paměti. Při stress testech, prováděných na počítači s 4GB paměti zvládala testovací dokumenty s ~ 20 000 kapitolami. S větší pamětí byla mez posunuta k ještě vyšším hodnotám při době zpracování matice podobností stále řádově v minutách. Pro překlad stačilo pouze nahradit knihovnu pro práci s SQLite databází za její 64bitový ekvivalent ze stránek projektu.

Program je řešen modulárně s tím, že si jednotlivé algoritmy doplňují příslušné položky do ovládacích panelů, dostupných v GUI aplikace. V současné době je naimplementováno několik filtrovacích modulů pro zpracování HTML textu počínaje segmentací textu na věty, odstavce a kapitoly, rozdělování na slova, jednoduchá „lemmatizace" (spíše „stemming" založený na odtrhávání běžných českých a anglických přípon), a vyčleňování stopslov. Vektorizace potom počítá vektory jednotlivých fragmentů pomocí běžného TF*IDF modelu.

Z dalších modulů je v aktuální verzi k dispozici zobrazování HTML textu obohaceného o možnost zobrazování slov obsažených v dané části a vektorů, spočtených pro daný fragment. Dále je možné počítat matice podobností objektů na dané úrovni granularity a tyto matice buďto zobrazovat v jejich číselné podobě, v podobě monochromatických obrázků, nebo je exportovat pro zpracování v programech třetích stran (CSV, PNG).

Z „pokročilejších" možností je k dispozici shlukování pomocí k-mean algoritmu a hierarchické shlukování pomocí postupné (binární) aglomerace nejpodobnějších vektorů.

Co se týká členění na kapitoly, je škoda, že program neumí rozlišovat kapitoly ve více úrovních, a všechny považuje za kapitoly na jediné úrovni místo samostatných kapitol uvozených značkou *h1*, obsahujících sekvence kapitol druhé úrovně atd. Vzhledem k experimentální povaze programu bych uvítal na jedné straně vyšší konfigurovatelnou implementovaných algoritmů (např. způsob počítání vektorů pomocí TF, NTF, TF*IDF, NTF*IDF, a řada dalších, které by dovolily snadno kombinovat a porovnávat výsledky bez nutnosti zasahovat do kódu nebo si psát vlastní moduly odvozené z těch již napsaných.

Program nabízí možnost „zkracování" textu pomocí hledání nejcharakterističtější věty nebo odstavce ke každému z k shluků získaných pomocí k-mean. Na podobném principu se snaží značkovat hierarchické shluky nejpodobnější větou. Opět bych vzhledem k povaze programu uvítal, kdyby byl naimplementován i nějaký standardní způsob značkování pro účely porovnání výsledků. V případě zkracování textu bych uvítal možnost výběru odstavců/vět i jiným způsobem než jen na základě 1 věta na 1 shluk. Pro daný účel by šlo použít i další algoritmy a opět porovnat výsledky mezi sebou, což by práci po stránce kvality pozvedlo.

Čekal bych také, že popis možných algoritmů v textové části bude nadmnožinou těch implementovaných a přehledně popíše důvody implementace daných algoritmů a další možné alternativy k těm implementovaným. Bohužel v tomto směru je text práce leckdy příliš stručný. Čekal bych také o dost obsáhlejší experimentální část, kde bude program a zatím dosažené výsledky ukázány na více různých typech publikací, bude zhodnoceno, nakolik výstupní data mohou přispět k vyššímu stupni orientace v daném typu publikace (sborník krátkých článků, rozsáhlé manuály, referenční příručky, ...) a případně se pokusí zhodnotit, proč daný algoritmus na daném souboru nebo typu souboru přinesl ty či ony výsledky, případně zda a jak by šlo program pro zpracování daného typu souboru dále vylepšit.

Právě pro tuto experimentální část by bylo vhodné mít více než jeden algoritmus v několika alternativách nebo s vyšší možností konfigurace stávajícího. Ucelenosti programu by prospělo to, aby program nereprezentoval v podstatě samostatné algoritmy, ale nějak je dokázal synergicky využít. Když už program pracuje s HTML, mohl by program zastřešit všechny/většinu ze stávajících částí a umožnit například do vstupního HTML souboru doplnit dodatečnou navigaci mezi kapitolami na základě jejich podobnosti (odkazy na nejpodobnější kapitoly, odkazy na nejbližší dostatečně podobné kapitoly), upozornit odkazem na části textu podobné tomu, co je uvedené v daném odstavci, ale nacházející se v jiné části dokumentu, doplnit rejstřík založený na shlucích a podobně. Uživatel by potom mohl takto rozšířený dokument běžně použít ve svém prohlížeči a mnohem lépe posoudit případné přínosy než na základě výsledků jednotlivých algoritmů odděleně.

Implementace mi v daném stavu přijde jako základ pro testování samostatných algoritmů. Ačkoli doporučuji proto práci k obhajobě, nejsem si jist, nakolik text práce v podobě, v jaké je odevzdán, svým rozsahem, formální úrovní a hloubkou odpovídá práci diplomové.

V Praze dne 23. 5. 2011