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

PŘEVOD VĚDECKÝCH ČLÁNKŮ NA TEXT

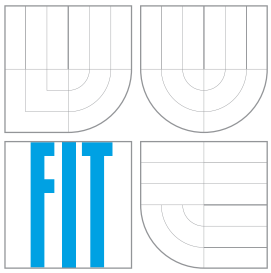
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

JIŘÍ MATIČKA

BRNO 2010

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

PŘEVOD VĚDECKÝCH ČLÁNKŮ NA TEXT

CONVERSION OF SCIENCE ARTICLES TO PLAIN TEXT

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

JIŘÍ MATIČKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. LUBOMÍR OTRUSINA

BRNO 2010

Abstrakt

Tato práce se zabývá převody vědeckých článků v elektronické podobě z různých formátů do prostého textu. Zaměřuje se hlavně na množinu problematických článků, u kterých je možné odhalit určité prvky způsobující neakceptovatelný výstup. Bylo proto zkoumáno mnoho převodních nástrojů a vybrán ten, jehož výstup nejvíce odpovídá požadované přesnosti převodů. Další část práce řeší problematiku automatizace převodu. Spadá sem vytvoření požadavku na převod, předání všech článků k převodu, vlastní převod, detekování ukončení převodu, kontrola výsledků převodu a předání převedených článků zpět. Toho je dosaženo na principu komunikace architektury klient/server, spoluprací skriptů napsaných v jazyce Python a dostupných potřebných knihoven. Z pohledu klienta je nutné vytvořit pouze seznam článků na převod a zavolat příslušnou funkci (vytvořit požadavek). O zbytek procesu je postaráno automaticky a výsledné textové soubory má klient k dispozici v předem zvolené složce.

Abstract

Purpose of this bachelor's work is a research in the area of converting scientific articles in electronic form to plain text. Main topic is the group of problematic articles with certain possible components causing non-acceptable output. Many conversion tools were investigated and the one with the required and most accurate conversion was chosen. Second part of this thesis examines the problematic of automated conversion, including creation of conversion request, forward of all articles to conversion, the conversion itself, detection of finished conversions and delivery of all converted articles. To achieve this objective, a communication principle based on client/server in conjuction with Python scripts and available needed libraries were created. From the client's point of view, it is required only to create a list of articles for conversion and then call the appropriate function (create a request). Rest of the process is taken care of automatically and the resulting text files are available for the client in a folder set beforehand.

Klíčová slova

ocr, převody článků, vědecké články, prostý text, pdf, konverze článků, elektronické články

Keywords

ocr, conversions of articles, science articles, plain text, pdf, electronic articles

Citace

Jiří Matička: Převod vědeckých článků na text, bakalářská práce, Brno, FIT VUT v Brně, 2010

Převod vědeckých článků na text

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Lubomíra Otrusiny.

.....
Jiří Matička
18. května 2010

Poděkování

Poděkování patří zejména mému vedoucímu Ing. Lubomíru Otrusinovi za jeho vstřícnost a ochotu pomoci při řešení problémů, které se během tvorby bakalářské práce vyskytly. Dále bych rád poděkoval i Ivu Dlouhému za cenné rady při tvorbě praktické části práce.

Obsah

Kapitola 1

Úvod

Cílem této práce je převádění článků, které většinou nelze převést některým běžně užívaným postupem. Veškeré uvažované články jsou dostupné v elektronické formě a v různých formátech. Cílem takového převodu je potom získat soubor pouze s prostým textem, tzn. bez obrázků, různých matematických vzorců a jiných speciálních znaků. Pojem "problematické články" bude upřesněn v samostatné kapitole 3.

Celá tato práce má pomoci s vývojem velkých projektů, na kterých se podílí Skupina zpracování přirozeného jazyka. Hlavní role skupiny v těchto projektech je výzkum a aplikace metod extrakce informací z textu, případně z multimediálních dat. To vše má napomoci studentům, vědcům či výzkumníkům hlavně k poskytování těch nejrelevantnějších a nejdůležitějších informací, které jsou v daném oboru nebo problematice k dispozici.

Obsah práce je rozdělen do několika kapitol, které zhruba odpovídají pořadí řešení jednotlivých podproblémů. Každá kapitola je pro přehlednost dále rozčleněna na několik podkapitol, jež důkladně popisují danou problematiku.

Kapitola 2

Převody článků na text

V dnešní době existuje nespočet různých článků, které jsou uloženy v elektronické formě. Existují také rozličné formáty, do nichž se takovéto články ukládají. První snahy a experimenty umožňovaly uložit dokument obsahující pouze text. S postupem času a rozvojem počítačových technologií vznikala potřeba obohatit samotný text i o obrázky nebo matematické vzorce pro vědecké články [8] a dokonce také o různé animace. To umožnilo vytvářet dokumenty bohatší na obsah, přehlednější, názornější a jistě i čitelnější.

Ne vždy je toto ale vyhovující stav a vyskytují se i potřeby opačné, tedy oproštění dokumentů od všech výše zmíněných "vylepšení". Tato práce se zabývá právě opačným postupem, kdy ze článků různých formátů s různým obsahem je zapotřebí vyextrahovat samotný text. Další nástroje a použité metody poté mohou takový text nějakým způsobem zpracovávat. Konkrétně například automaticky indexovat obsah dokumentu a převést jej do XML struktury.

Bylo vytvořeno již mnoho nástrojů, které z různých formátů článků dokaží vyextrahovat samotný text a ten uložit nejčastěji do textových souborů. Jak se ale ukázalo, tyto nástroje pracují korektně pouze s některými druhy článků. Situace je dále ještě složitější, protože různé nástroje mají pro stejné články různé výstupy. Výsledkem je fakt, že pro jistou množinu článků dosáhneme uspokojujících výsledků, s kterými poté můžeme dále pracovat. Pro ostatní je zapotřebí použití jiných postupů.

Kapitola 3

Problémy převodů článků na text

Na problémy při převodu má vliv více faktorů. Pokud bychom si prohlédli větší množství různých článků, došli bychom pravděpodobně k jistým závěrům, na jejichž základě by bylo možné takové dokumenty dělit do několika kategorií. Jiné dělení se potom zabývá přímo problematickými prvky v článcích.

3.1 Dělení článků podle různých kategorií

Veškeré uvažované dokumenty v elektronické formě se dají rozdělit podle různých kritérií. To mimo jiné umožňuje například pro různé typy článků užívat různé nástroje nebo i specializované nastavení parametrů převodu pro takovou skupinu. Správným rozdělením dokumentů je možné maximalizovat úspěšnost výstupů a tedy i úspěšnost výsledků dalších nástrojů užívaných v automatizovaném procesu.

U každého dělení bude diskutováno hlavně hledisko míry problematičnosti převodu daného typu článku.

3.1.1 Podle původu jejich vzniku

- Články vytvořené libovolným textovým editorem přímo na PC a uložené do souboru.
- Články získané ofocením, okopírováním nebo oskenováním předlohy v papírové podobě.

První skupina článků je ideální z hlediska pozdější nutnosti převodu na text. Text takovýchto dokumentů není deformován ani jinak znehodnocen a převádějící nástroje taktéž mohou využít vnitřní struktury dokumentu, což dále opět zlepšuje kvalitu výstupu.

Naproti tomu druhá uvedená skupina výchází v tomto směru o poznání hůře. Vstupuje tu do hry jeden důležitý faktor a tím je převod předlohy do elektronické podoby. Dochází zde do určité míry ke ztrátě kvality. Tato problematika je ale velice rozsáhlá a nebude tu podrobně rozebírána. Budou uvedeny pouze dva hlavní rysy. Jedním z nich je kvalita zařízení, které dokument převádí do elektronické formy. Ani sebedokonalejší zařízení ovšem nic nenadělá s dokumentem, který je sám o sobě ve špatném stavu. Drobné korekce jsou většinou možné, ale nelze si jednotlivé části dokumentu domýšlet. Druhým faktorem je člověk. Ten má vliv na upnutí nebo správné založení a pozici skenovaného či kopírovaného dokumentu. Oba faktory musí vykazovat k získání přijatelného výsledku uspokojivé vlastnosti. U takovýchto typů dokumentů také potom chybí informace o nějaké struktuře textu, protože soubor je uložen nejčastěji v obrazovém formátu.

3.1.2 Podle použitého písma (fontu)

- Články psané v minulosti na stroji nebo s nestandardním typem fontů.
- Články se současným ustáleným typem fontů.

Problém převodu článků psaných na stroji se úzce pojí s nutností takové články převést do digitální formy (viz výše). Písmo pak bývá často i nějakým způsobem znehodnocené a působí při převodu problémy. Některým nástrojům mohou dělat potíže i různé nestandardní použité fonty. Úplně naopak je tomu u druhého typu článků, kde použité písmo je ve většině případů bezpatkové (Arial), patkové (Times New Roman) nebo strojové (Courier). Všechny uvedené fonty se dnes běžně využívají při psaní dokumentů na počítači. Jejich užitím můžeme celý proces převodu jedině usnadnit.

3.1.3 Podle formátů, ve kterých jsou uloženy

- Formát PDF – Portable Document Format
- Formát PS – PostScript
- Formát DOC – formát textového editoru Microsoft Word

Uvedené formáty jsou nejběžnějšími pro uložení textových dokumentů, ale ne ojedinělé. Existují i mnohé další. Za všechny uvedu například rtf (Rich Text Format)[3].

Jedním z nejpoužívanějších a nejrozšířenějších je prvně jmenovaný pdf [2]. Jde o otevřený formát a má tedy všechny výhody těchto formátů. Jedná se hlavně o přenositelnost. K tomu, aby si uživatel mohl přečíst obsah souboru v pdf, postačí instalace některého volně dostupného programu například z internetu.

V případě fomátu ps [1] je situace obdobná. Není ovšem tolik používaným, nicméně je základem pro výše zmiňovaný pdf formát.

Poslední uvedený doc [6] je formát, který lze vytvořit pomocí textového editoru Word firmy Microsoft. Nejedná se o otevřený formát a k přečtení obsahu je nutné mít nainstalovaný Word (případně OpenOffice). To značně ovlivňuje přenositelnost takovýchto dokumentů.

Mezi jednotlivými fomáty lze provádět konverze. Na internetu je možné opět nalézt širokou škálu takových aplikací. O některých bude ještě pojednáno v kapitole 4.1.

3.1.4 Podle použitého nástroje

- Takové články, které daný nástroj převede korektně.
- Takové články, které po převodu obsahují neakceptovatelné množtví nekorektních znaků nebo je jejich výsledná forma zcela nerozpoznatelná od originálního článku.

Toto dělení není úplně typické, ale hraje významnou roli při využití některého nástroje v zautomatizovaném procesu. Je důležité vybrat ten, jehož první množina článků bude co možná největší. Další postupy potom musí pracovat s druhou množinou a úspěšně z ní převést co nejvyšší procento článků.

3.2 Příčiny neúspěchu při převodu článků

V této kapitole budou rozebrány nejčastější příčiny, které způsobují, že je převod nevyhovující či úplně nepoužitelný. Budou rozděleny do několika skupin, které vyplynuly z manuálního zkoušení různých nástrojů a následné kontroly správnosti výstupů.

1. Obrázky
2. Matematické vzorce a výrazy
3. Vícesloupcové články
4. Nečitelná vnitřní struktura souboru článku
5. Asijské jazyky

3.2.1 Obrázky

Obrázky v textu umožňují doplnění textu o příklady, jindy jsou vhodné jako základ k pochopení nějakého složitějšího problému. Ovšem při snaze vyextrahovat z dokumentu pouze text jsou jakousi překážkou, kterou některé nástroje nedokáží překonat. Jiné si s nimi poradí jen částečně a potom vzniká ve výsledném textu množství nežádoucích znaků. Většina z prověřených a otestovaných nástrojů umožňuje obrázky z textu přeskočit. Takovéto chování je pro nás žádoucí a vyhovující.

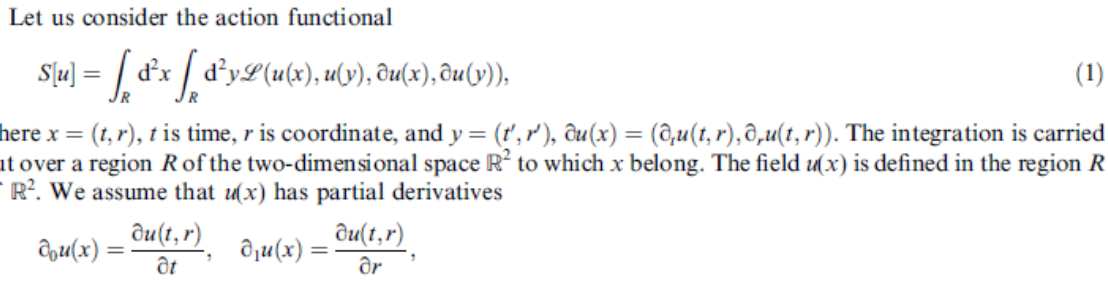
3.2.2 Matematické vzorce a výrazy

Vědecké texty, které jsou hlavní náplní této práce, obsahují povětšinou více či méně matematických vzorců, výrazů nebo různých (nejčastěji řeckých) symbolů. S nimi má problém značná část nástrojů, které se snaží o jejich převod. Výstup je často podobný a obsahuje velké množství nesmyslných symbolů. To je také jedna z příčin následného selhání jiných nástrojů provádějících například indexaci. Je tedy nutné maximalizovat snahu na eliminování těchto výsledků, a to ať již výběrem lepšího nástroje nebo následnými rutinami pro odstranění nekorektních znaků v textu.

Existují i takové nástroje, které (podobně jako u obrázků) dokáží vyfiltrovat některé matematické symboly nebo dokonce i jednoduché výrazy, ale s dalšími vzorci si opět neporadí.

V kapitole 4.1 o nástrojích pro převod bude pojednáno i o jednom speciálním programu, který zvládne zcela odfiltrovat i matematické vzorce. Bohužel jeho nevýhody jsou v jiných oblastech.

Na obrázcích 3.1 a 3.2 je ukázka textu před a po převodu článku obsahujícího vzorce a jiné matematické zápisy.



Let us consider the action functional

$$S[u] = \int_R \mathrm{d}^2x \int_R \mathrm{d}^2y \mathscr{L}(u(x), u(y), \partial u(x), \partial u(y)), \quad (1)$$

where $x = (t, r)$, t is time, r is coordinate, and $y = (t', r')$, $\partial u(x) = (\partial_t u(t, r), \partial_r u(t, r))$. The integration is carried out over a region R of the two-dimensional space $\mathbb{R}^2$ to which x belong. The field $u(x)$ is defined in the region R of $\mathbb{R}^2$. We assume that $u(x)$ has partial derivatives

$$\partial_0 u(x) = \frac{\partial u(t, r)}{\partial t}, \quad \partial_1 u(x) = \frac{\partial u(t, r)}{\partial r},$$

Obrázek 3.1: Původní text s matematickými zápisy před převodem ve formátu PDF

```
Let us consider the action functional Z Z S u Ľ d2 x
 d2 yLđuđxŢ; uđyŢ; ouđxŢ; ouđyŢŢ; R R

đ1Ţ

where x Ľ đt; rŢ, t is time, r is coordinate, and
y Ľ đt0 ; r0 Ţ, ouđxŢ Ľ đot uđt; rŢ; or uđt; rŢŢ.
The integration is carried out over a region  R
of the two−dimensional space R2 to which x belong.
The field u(x) is defined in the region R of R2 .
We assume that u(x) has partial derivatives o0 uđxŢ
 Ľ ouđt; rŢ ; ot o1 uđxŢ Ľ ouđt; rŢ ; or
```

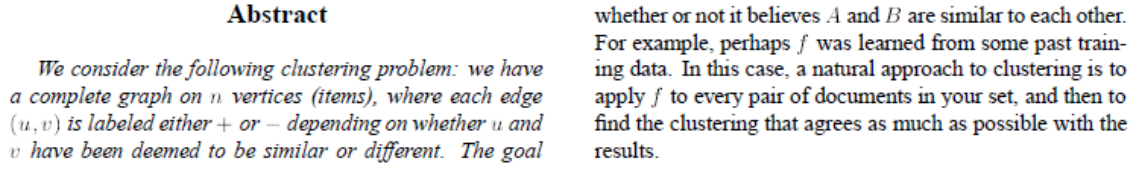
Obrázek 3.2: Prostý text s matematickými zápisy po převodu

3.2.3 Vícesloupcové články

I sebepropracovanější nástroje, které si poradí s předchozími úskalími, nic nenadělají v případě, že článek obsahuje více sloupců a tyto nástroje s takovým rozvržením nepočítají. Výsledný text je v každém případě nepoužitelný, byť může být správně převeden na úrovni symbolů a znaků.

V tomto případě se nástroje opět dělí na ty, které s více sloupci počítají a na ty, které ne.

Na obrázcích 3.3 a 3.4 je ukázka textu před a po převodu článku, který je napsán do dvou sloupců.



Abstract

We consider the following clustering problem: we have a complete graph on n vertices (items), where each edge (u, v) is labeled either $+$ or $-$ depending on whether u and v have been deemed to be similar or different. The goal whether or not it believes A and B are similar to each other. For example, perhaps f was learned from some past training data. In this case, a natural approach to clustering is to apply f to every pair of documents in your set, and then to find the clustering that agrees as much as possible with the results.

Obrázek 3.3: Původní text se dvěma sloupci před převodem ve formátu PDF

Abstract
whether or not it believes and are similar to each other.
For example, perhaps was learned from some past train−
We consider the following clustering problem: we have
ing data. In this case, a natural approach to clustering is to
a complete graph on vertices (items), where each edge
apply to every pair of documents in your set, and then to
is labeled either or depending on whether and
find the clustering that agrees as much as possible with the
have been deemed to be similar or different. The goal
results.

Obrázek 3.4: Prostý text původně obsahující dva sloupce po převodu

3.2.4 Nečitelná vnitřní struktura souboru článku

Vyskytují se i dokumenty, u nichž po zpracování nějakým nástrojem dojde k úplnému vypuštění části textu, případně jasně čitelný text se převede na změť nesmyslných znaků a symbolů. Prakticky všechny nástroje, které pracují nějakým způsobem se strukturou souboru a jeho metainformacemi, takový dokument nedokáží korektně převést. Jako nejlepší řešení se potom jeví nevyužívání těchto informací a struktury, ale pracování s dokumentem jako s obrázkem.

Na obrázcích 3.5 a 3.6 je ukázka textu před a po převodu článku, jehož vnitřní struktura je chybná, poškozená nebo pro daný nástroj jiným způsobem nečitelná.

DAM: a DoS Attack Mitigation Infrastructure *

Byung-Gon Chun, Rodrigo Fonseca and Puneet Mehra
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
{bgchun,rfonseca,pmehra}@eecs.berkeley.edu

ABSTRACT

Denial-of-Service (DoS) attacks represent a serious and growing threat to the firms utili ing the Internet. Defenses against DoS that rely upon identification and isolation of the attack stream are difficult due to the distributed nature of most attacks and source IP spoofing. Stopping an attacking host and a series of successful DoS attacks on such high-profile targets as eBay, Yahoo, CNN.com, and buy.com in February 2000 prompted attention from the Attorney eneral and an official FBI investigation into the source of the attacks [2]. These attacks have had a significant financial impact on companies, with the firms losing an estimated $1.2 bil-

Obrázek 3.5: Původní text s nečitelnou vnitřní strukturou před převodem ve formátu PDF

3.2.5 Asijské jazyky

Je otázkou, do jaké míry lze považovat převod asijských jazyků za problém. Je zřejmé, že pokud nástroj není na takový převod vybaven, nevznikne nikdy smysluplný výstup. Naštěstí je těchto článků poskrovnu a tudíž množství problémů s převodem není neúměrný počet. Nástroje, které dokáží takovéto jazyky převést, ale existují a v případě potřeby je

e a t o n o e o m R m o a n n a S m a n o n o e n o d n
y o e a o o o n o o m m u s u a n U n o t d o n T R a n
o t a n n m
o u INTRODUCTION

DAM: a DoS Attack Mitigation Infrastructure

Byung–Gon Chun, Rodrigo Fonseca and Puneet Mehra

Department of Electrical Engineering and Computer Sciences
University of California, Berkeley

bgchun,rfonseca,pmehraa @eecs.berkeley.edu

c d o a n m o ta a t m o o t a tn n a en a ta a t o e e n
o oe a a d o n o n d a n a pd n a a n e om e a a m aa n
a a So ea a t m o o t a e n n a o n n o oe n m o a a t m e

Obrázek 3.6: Prostý text po převodu pdf s nečitelnou vnitřní strukturou

lze využít. Zmínka o jednom takovém bude ještě v kapitole 4.1 o nástrojích pro převod. Samotná problematika převodů asijských jazyků je ovšem rozsáhlá. Už například jenom z důvodu, že některé texty mohou být psány vodorovně, jiné svisle.

Na obrázcích 3.7 a 3.8 je ukázka textu před a po převodu jednoho z asijského jazyka.

超光谱遥感成像是一种极为重要的技术,被广泛应用于环境监测、地质、农业、医学和军事等领域.它通常由上百个非常窄的连续的光谱波段组成,通过分析图像中近乎连续的光谱曲线,可以分辨出不同物体光谱特征的微小差别,有利于识别更多的目标.但是,在获得较高的谱间分辨率的同时,也产生了巨大的数据量.为了能够快速传输和处理这些数据并减少存储空间,必须对数据进行有效的压缩编码,但目前国内外还没有一种标准的超光谱(多光谱)遥感图像的压缩标准.与普通二维静止图像不同,超光谱图像数据可以被看作由 2D 的空间域和 1D 的频谱域组成 3D 体数据,图像中不仅存在空间相关,还存在谱间相关.目前普遍采用变换的方法去除相关性,比如利用三维 DCT(discrete cosine transform)变换[1],或者先利用 Karhunen-Loeve 变换(Karhunen-Loeve,简称 KLT)去除谱间相关,再利用 DCT 去除空间相关[2].由于 DCT 变换编码会产生方块效应,因此越来越多的变换方法都采用小波变换(wavelet transform,简称 WT)[3-11],因为小波变换能够产生高度集中的能量,且没有 DCT 变换的方块效应,同时,它的复杂度适中.在文献[4-6]提出的方法中,在利用 KLT 去谱间相关的同时,还利用 WT 去除

Obrázek 3.7: Původní text psaný asijským jazykem před převodem ve formátu PDF

Ę ő ,± » ą ă ,ĺ ¨ ą ý .µ « Ę Ç .Ó ë Ć Ő Í ¨ ¶ ţ Î ¬ Ľ ¸ Ö ą Í Ľ Ď ń ¸ » Í ¬ ,Ô Ú » ń µ Ä ˝ Đ ¸ ß µ Ä ,± Ř Đ ë ¶ Ô Ę ý

·şÓ¦ Ä ·Ö ÎöÍĽĎńÖĐ ĽÝ˝ř

ÓÚ»·Ľ˛ ˝üşő Ć×Ľä·Ö±ĆĂĘ Đ ÓĐ ,» ą ´ ć Ô Ú Ć × Ľ ä Ď ŕ ą Ř
[1]

Ľŕ ˛â˘˘ Á¬ĐřµÄąâ µÄ Đ§µÄŃąËő ,˛ ¬ ą â Ć × Í Ľ Ď ń Ę ý Ľ Ý ż É Ń Ô ± » ż ´ × ÷ Ó É

Obrázek 3.8: Prostý text po převodu asijského jazyka

Kapitola 4

Možnosti převodu problematických článků

V této kapitole budou probrány jednotlivé nástroje pro převod článků a u každého zdůrazněny jak jeho přednosti, tak naopak i jeho nevýhody. Další část kapitoly se potom zabývá postupy a různými kombinacemi vstupů a výstupů jednotlivých nástrojů, jež by měly vést k dosažení požadovaných výsledků.

Jak již bylo lehce nastíněno, existují dva různé přístupy, jak se postavit k řešení problémů s převody článků.

1. Vyhledání a použití vhodných nástrojů na převod.
2. Zrealizování postupu, pomocí něhož dosáhneme požadovaných výstupů převodu.

Je nutné říci, že hranice mezi těmito dvěma postupy je tenká a v mnohých případech dosáhneme požadovaného výsledku kombinací obou, tzn. zapojením vhodných nástrojů do nějakého postupu práce. Vše je podrobněji vysvětleno v podkapitole 4.2.

4.1 Nástroje pro převod problematických článků

Při hledání vhodného nástroje, který by vyhovoval potřebám převodu takových článků bylo nalezeno velké množství různých aplikací. Některé svými výsledky vyhovovaly více, jiné méně a další byly zcela nepoužitelné. Pokud je řeč o nepoužitelnosti, jde stále o tu skutečnost, že se pracuje s články problematickými (kapitola 3). Naopak s velkým množstvím ostatních dokumentů si dokáží poradit velice dobře nebo alespoň uspokojivě.

Veškeré nástroje použitelné pro tuto práci se v zásadě dají rozdělit podle dvou různých kritérií.

Dělení podle způsobu převodu:

- Aplikace pro extrakci textu z pdf
- Aplikace na principu optického rozpoznávání znaků (OCR) [7]

Dělení podle licence:

- Komerční aplikace

- Volně dostupné aplikace

Za komerční aplikace je potřeba uhradit nějaký poplatek úměrný často její robustnosti a propracovanosti. Naproti tomu jsou volně dostupné aplikace distribuované zdarma a postačí je pouze stáhnout z příslušné adresy na internetu. Velice často bývají aplikace pro Windows zároveň i komerční a aplikace pro Linux volně dostupné. Neplatí to ovšem pro všechny a existují výjimky jak na jedné, tak i na druhé straně.

4.1.1 Aplikace OCR pro operační systém Linux

Při hledání nástrojů pro operační systém Linux bylo zjištěno, že takových aplikací je poskrovnu. Nicméně nějaké existují a zmíněny budou dva, které prošly odzkoušením na jistém vzorku článků a srovnáním jejich výstupů s původním textem.

Tento druh aplikací má jednu nespornou výhodu – většina z nich lze provozovat pomocí příkazového řádku. S tím by nejspíše nesouhlasili běžní uživatelé, kteří jsou zvyklí na různé podoby grafického uživatelského rozhraní. Avšak pro zapojení takových aplikací do automatizovaného procesu je tato vlastnost naopak téměř nepostradatelná (to na druhou stranu neznamená, že by to jiným způsobem nebylo možné).

GOCR

Tento software [22] je sice zařazen do Linuxových aplikací, ale dnes již funguje i na dalších platformách (např. Windows, OS/2). Jako vstup implicitně přijímá soubory ve formátu pnm, pbm, pgm, ppm, pcx a tga. Další běžné formáty jako jsou png, jpg, tiff, gif a bmp jsou automaticky konvertovány na jeden z předešlých.

Po otestování na různých množinách vzorků problematických článků vykazoval tento software nepřijatelné výstupy.

Výhody:

- Možnost ovládání z příkazového řádku.
- Dostupný zdarma, GNU Public Licence [11].
- Podpora formátů jako jsou (jpg, bmp, gif, tiff, png).

Nevýhody:

- Problémy se samotným rozpoznáváním některých znaků.
- Problémy s převodem matematických vzorců a výrazů.
- Problémy s více sloupci textu.

Tesseract

I tento software [21] je použitelný na více platformách, ale pro jeho možnost ovládání pouze z příkazového řádku byl zařazen do této kapitoly. Nutno podotknout, že příkazový řádek je jediná možnost ovládání. Vstupem nástroje je soubor ve formátu tiff a jako výstup dostaneme textový soubor.

Po otestování na stejných množinách vzorků článků nebylo dosaženo přijatelných výstupů.

Výhody:

- Možnost ovládání z příkazového řádku.
- Dostupný zdarma, Apache Licence.
- Jeden z nejlepších open source OCR nástrojů.

Nevýhody:

- Neporadí si s obrázky v textu.
- Neporadí si s matematickými vzorci a výrazy.
- Chybí analýza rozvržení dokumentu (nepoužitelné pro vícesloupcový text).

4.1.2 Aplikace extrahující text z pdf pro Windows

Aplikací umožňujících převody článků pro Windows je celá řada. Zpravidla se jedná o komerční nástroje dostupné nebo plně funkční až za nějaký poplatek. Většina z nich jsou jednoduché aplikace (řádově jednotky, maximálně desítky MB), které dávají podobné výstupy. Objevují se ale i robustní a velmi propracované nástroje, jejichž preciznost a přesnost převodů různorodých článků je na velice vysoké úrovni.

Budou zde uvedeni zástupci obou těchto skupin a někteří budou popsáni podrobněji. Každý z uvedených zástupců má své grafické rozhraní a není možné jej ovládat z příkazového řádku.

PDF Converter (AnyBizSoft)

Velice účinný nástroj [19], který pracuje s formáty pdf. Podporuje převod do různých výstupních formátů. Pro naše účely je nejzajímavější právě prostý text. Ve většině případů spolehlivě odfiltruje z dokumentů obrázky a také velice problematické matematické vzorce, výrazy a symboly. Nedokáže převést chráněná pdf. Celkově je převod relativně rychlý a spolehlivý.

Výhody:

- Rychlost převodu.
- Filtrování matematických výraziv (vzorce, výrazy, symboly).
- Filtrování obrázků z dokumentu.

Nevýhody:

- Nepřevede chráněná pdf.
- Nemožnost ovládání pomocí příkazového řádku.
- Zdarma pouze zkušební verze.

ABC Amber PDF Converter

Tento nástroj [13] umí vyfiltrovat matematické výrazivo i obrázky. S některými dokumenty má ale problém. Zvládá i vícesloupcové texty a jeho největší výhodou je veliká rychlost převodu.

Výhody:

- Filtrování velké části matematických výraziv.
- Filtrování obrázků.
- Velice rychlý převod.

Nevýhody:

- S některými dokumenty se nevypořádá.
- Nemožnost ovládání pomocí příkazového řádku.
- Zdarma pouze zkušební verze.

Solid Converter

Uspokojivě odstraňuje obrázky a matematické výrazivo z dokumentu. Nepřevede chráněné pdf soubory. Více informací o produktu je možné nalézt zde [20].

Výhody:

- Filtrování velké části matematických výraziv.
- Filtrování obrázků.

Nevýhody:

- Nepřevádí chráněná pdf.
- Nemožnost ovládání pomocí příkazového řádku.
- Zdarma pouze zkušební verze.

4.1.3 Aplikace OCR pro Windows

Nitro PDF Professional

Jedná se o nástroj [15] s mnoha funkcemi. Pro naše účely nejzajímavější je právě převod do prostého textu. Umožňuje odfiltrovat z textu obrázky. Zvládne i filtrování matematických vzorců, ovšem ne vždy. Problémy této aplikaci působí některá pdf ze skupiny problematických (kapitola 3) s poškozenou vnitřní strukturou, která vůbec nepřevede. Neporadí si ani s chráněnými pdf soubory. Současná verze umožňuje již i OCR rozpoznávání.

Výhody:

- Filtrování velké části matematických výraziv.
- Filtrování obrázků.
- Další funkcionalita s pdf.

Nevýhody:

- Nepřevádí chráněná pdf.
- Neporadí si s některými pdf a nepřevede je.
- Nemožnost ovládání pomocí příkazového řádku.
- Zdarma pouze zkušební verze.

ABBYY FineReader

Jedná se o velice precizní OCR software [12]. Stejně jako předchozí nástroje uvedené v této podkapitole tedy pracuje se soubory ve formě obrázku. Poradí si s obrázky, vícesloupcovým textem, tabulkami a dalšími. Problémem jsou ovšem matematické vzorce a výrazy.

Výhody:

- Dovede vyfiltrovat obrázky.
- Korektně převádí i soubory ve zhoršené kvalitě.
- Korektně převádí vícesloupcové texty.

Nevýhody:

- Neporadí si s matematickým výrazivem.
- Nemožnost ovládání pomocí příkazového řádku.
- Zdarma pouze zkušební verze.

InftyReader

Velice propracovaný OCR program [14] zaměřený hlavně na převod a zpracování vědeckých článků. Vstupem mohou být soubory ve formátech tiff, bmp, gif, png a také pdf. Výstup je umožněn do formátů iml (formát pro InftyEditor), xhtml, Microsoft Word 2007 (xml) a tex. Uživatelské rozhraní je velice jednoduché a intuitivní. Snad jedinou — zato velkou — nevýhodou je dlouhá doba převodu. U asi 20-ti stránkového pdf souboru cca 15–20min.

Výhody:

- Dokáže vyfiltrovat obrázky.
- Korektně převádí i soubory ve zhoršené kvalitě.

- Korektně převádí vícesloupcové texty.
- Dokáže označit matematické výrazivo a korektně ho převést.
- Jednoduché uživatelské rozhraní.

Nevýhody:

- Dlouhé trvání doby převodu jednoho článku.
- Při výskytu chyby při převodu u jednoho článku skončí celý převod (dávkové zpracování).
- Nemožnost ovládání pomocí příkazového řádku.
- Zdarma pouze zkušební verze.

Nuance OmniPage Professional

Další OCR nástroj [18], který se řadí na přední pozice ve svém oboru. Nabízí kompletní řešení pro práci s dokumenty – od skenování a ukládání souboru až po výstup v podobě mnoha formátů. Jako vstup je možné zvolit soubory běžných obrázových formátů i některé méně známé. Jako výstupní soubory je možné zvolit formáty pro prostý text v několika variantách (formátované, odřádkované, neformátovaný text), doc formát, html a další. OmniPage podporuje rozpoznávání mnoha jazyků a to včetně asijských.

Je možné provádět převody, při kterých je uživatel přítomen a může si tak korigovat případné nedostatky v převodu nebo upravovat rozvržení různých oblastí dokumentu ještě před samotným převodem. Lze také doupravovat nahrané soubory na úrovni obrázků a tím případně napomoci k přesnějším výsledkům.

Uživatel ovšem nemusí být u převodů přítomen vůbec. OmniPage nabízí velice sofistikovanou funkcionalitu – provádění všech výše popisovaných manuálních korektur zcela automaticky. S tím souvisí nastavení tzv. workflow, což je sada úkonů a jejich parametrů, které se mají provést v definovaném pořadí na jeden dokument nebo skupinu dokumentů. Celý proces je ale doveden ještě o něco výše, a tak lze vytvářet úlohy na hromadné zpracování. Jedná se o tzv. batch processing. Zde se uplatní nastavení workflow a přidá se ještě informace o časovém údaji, kdy daný převod provést a kdy skončit. Podrobnější informace lze nalézt v uživatelském návodu [16].

Mimo jiné OmniPage poskytuje také užitečnou funkcionalitu nazvanou watched folder. Jedná se o nastavení složky, kterou aplikace v určitých intervalech testuje na přítomnost nových dat. V případě dodání souboru či souborů s očekávanou příponou se na všechny aplikuje nastavený workflow a dojde k převodu a uložení výsledků.

Snad jediná nevýhoda tohoto nástroje spočívá ve špatném převodu matematických vzorců a výrazů, které nejsou rozpoznány a jsou tedy převáděny jako běžný text.

Výhody:

- Dokáže vyfiltrovat obrázky.
- Korektně převádí i soubory ve zhoršené kvalitě.
- Korektně převádí vícesloupcové texty.

- Umožňuje dávkový převod souborů.
- Umožňuje plánování převodů.
- Uplatnění funkcionality watched folder na příjem souborů v předem nedefinovaných časech.

Nevýhody:

- Nemožnost ovládání pomocí příkazového řádku.
- Nerozpoznání a nevyfiltrování matematických vzorců.
- Zdarma pouze zkušební verze.

4.2 Různé postupy pro převod článků

Často se potýkáme se situací, kdy neexistuje nástroj, který by byl schopen naplnit veškeré naše představy o výstupu. Řešením potom může být hledání různých postupů skládajících se z více elementárních operací, jež mohou vhodnou spoluprací kýženého výstupu dosáhnout. Následující podkapitola se touto problematikou zabývá podrobněji a obsahuje konkrétní realizaci vytváření postupů pro automatizovaný převod článků na text.

Bylo by nemožné zde uvést vyčerpávající výčet různých takových možností postupů a ani to není cílem této podkapitoly. Budou zmíněny pouze takové, které se nějakým způsobem týkají převodu problematických článků, přičemž snaha bude o zobecnění použití na co největší množinu těchto článků.

4.2.1 Převod souborů na obrazový formát

V kapitole 3 byla zmínka o souborech, jejichž vnitřní struktura je nějakým způsobem poškozená nebo pro většinu nástrojů nečitelná. Pokusy o převody těchto souborů často končí chaotickým výstupem se spustou všech možných znaků, nepřevedením daného souboru nebo v nejhorším případě dokonce pádem či zamrznutím celé aplikace. To je samozřejmě nežádoucí jev a při snaze o převod způsobuje veliké potíže.

Řešením může být mezikrok obsahující konverzi výše zmíněných souborů na obrazový formát. Není to nijak závažně omezující krok, protože programů, které konverze formátů zvládají, je celá řada. Dokument v obrazovém formátu poté můžeme pohodlně zpracovat nějakým OCR nástrojem, který se žádnou vnitřní strukturou nezabývá a analyzuje soubor tak, jak je. Pixel po pixelu.

4.2.2 Násobná konverze mezi formáty

Myšlenka spočívá ve využití vlastností nějakého vhodného formátu, jehož obsahem by se případně mohlo snadno procházet a odfiltrovat nepotřebné části dokumentu. Rovněž může existovat nástroj, který zvládá převod do jiného formátu lépe než do námi požadovaného. Proto může být někdy lepší využít takového kroku k získání kvalitnějšího výstupu. Nejlepší bude si pro názornost vše vysvětlit na příkladu.

Umí-li některý nástroj převod například do formátu html, je možnost využít uzavírání obrázků nebo jiných elementů do tagů. Není většinou problém vytvořit jednoduchý skript, který projde takovým souborem a odfiltruje příslušné tagy. Zbyde nám tedy html soubor bez nežádoucích elementů a na ten lze opět využít jiného nástroje zajišťujícího převod z formátu html na prostý text.

Kapitola 5

Proces převodu

V této kapitole bude podrobně rozebrána celá problematika procesu převodu. Budou zahrnuty takové oblasti jako je zvolení vhodného převodního nástroje a s tím související cílová platforma, na které by se převody měly odehrávat, problematika automatického realizování převodů na dané platformě, zvolení metodiky přenosu souborů k převodu a zpět a další nezbytné činnosti z toho vyplývající.

5.1 Snaha o zautomatizování převodu

Existují dvě základní možnosti, jak lze dané články převádět. Každá má své pro i proti a záleží hlavně na konkrétní potřebě, pro jakou mají být převody realizovány.

- Manuální převody
- Automatické převody

První typ převodů se vyznačuje jedním důležitým faktorem, a tím je přítomnost člověka. Ten je jako tvor s určitou inteligencí schopen sledovat průběh převodu, reagovat na vzniklé situace a řešit je. Jde vlastně o určitý dialog mezi danou osobou a počítačem, kdy na začátku dojde k jistému impulsu vedoucímu na potřebu realizování převodu určitého článku. To má za následek započetí komunikace s počítačem, sdělení mu svých požadavků, dohodnutí se na určitých parametrech operace a čekání na výsledek, přičemž počítač může v celém průběhu klást různé upřesňující dotazy nebo sdělovat nenadálé problémy při provádění daného úkonu. Na výstup je pak člověk upozorněn nejčastěji ve formě zobrazení nějaké zprávy nebo doprovodným zvukovým signálem.

Výhoda tohoto postupu spočívá v tom, že uživatel má celý proces převodu pod kontrolou a je schopen se v případě potřeby rozhodnout pro různé alternativní možnosti. Tyto převody jsou pak vhodné v případech, kdy je třeba pokaždé pracovat s různými parametry převodu nebo jde o to si pouze něco vyzkoušet, otestovat. Z toho plyne nevýhoda, kterou je hromadné zpracování podobných typů článků nebo potřeba použít na jistou množinu článků převod se stejnými parametry. Tady by musel člověk opakovaně vykonávat tu samou činnost, což vede k velké neefektivitě a ztrátě cenného času.

Druhý typ převodů je realizován pouze za pomoci počítače. Tím odpadá potřeba přítomnosti osoby po celou dobu vykonávání převodu. Je tedy vhodný na velké množství opakujících se operací. Na druhou stranu se tím ale vytrácí možnost upravovat parametry převodu "na míru" každému jednotlivému článku a všechny výše zmíněné interakce člověk-počítač z tohoto procesu odpadají.

Jelikož je tato práce zaměřena na jisté výpomoci automatizovaným systémům pro zpracování přirozeného jazyka, je použita právě výše zmíněná druhá možnost automatického převodu a dále budou uvedeny všechny problémy z toho plynoucí a popsáno jejich řešení.

5.2 Příprava před samotnou realizací

Při vytváření konkrétní realizace postupu automatizace celého procesu převodu je třeba vycházet z možností dostupných prostředků a také z poznatků, které byly doposud uvedeny. Při práci na tomto problému vyvstaly tři základní otázky, které budou dále popsány a zodpovězeny.

5.2.1 Výběr převodního nástroje

Tento první krok je velice důležitý, neboť na něm největší měrou závisí samotné výsledky převedených článků. Bylo proto otestováno velké množství různých nástrojů a zkoumáno, jaké výstupy produkují při obdržení určité množiny testovacích článků. Ty, které obstály pro daný účel nejlépe, byly již zmíněny v kapitole 4.1. Daným účelem se rozumí poměr mezi co nejpřesnějším převodem a časem, který uplynul od začátku do konce celého převodu. To je nejdůležitější kritérium, podle kterého byl nástroj vybírán. Brány v potaz jsou také kritéria, jako je cena produktu, dostupnost produktu a v neposlední řadě použitelnost pro konkrétní potřeby (rozhraní pro komunikaci s okolním světem).

Obrázek 5.1: Logo produktu Nuance

Nakonec byl po zvážení všech těchto pro a proti a po konzulataci s některými členy pracujícími na projektu ReReSearch vybrán jako vhodný nástroj Nuance OmniPage Professional. Parametry tohoto nástroje byly již uvedeny v kapitole 4.1. Zapojení OmniPage do zautomatizovaného procesu převodu a využití některých jeho vlastností bude popsáno dále.

5.2.2 Výběr cílové platformy pro běh nástroje

Právě výběr cílové platformy byl dalším neméně důležitým kritériem pro výběr vhodného převodního nástroje, přičemž bylo nutné rozhodnout mezi dvěma přístupy. Buď vybrat určitý nástroj na základě zvolené platformy nebo nejprve zvolit nástroj a podřídit tomu potom výběr platformy.

Snaha byla o provádění převodů na unixovém systému, což by přinášelo jisté výhody – hlavně ovládání a komunikaci s nástrojem za pomoci příkazového řádku a v neposlední řadě by se eliminovaly problémy vyvstávající při přenosu souborů mezi dvěma různými platformami. U nástroje OmniPage existují varianty jak pro Windows (klasická verze OmniPage Professional [18]), tak multiplatformní (tzv. SDK verze – software development kit [17]). Velkou nepříjemností byla u verze SDK hlavně pořizovací cena, jež se pohybuje řádově v desítkách tisíc Kč.

Jako řešení byla nakonec vybrána verze OmiPage pro Windows, která již byla na fakultě dostupná. Zvolený postup je tedy takový, kdy je nutné vybranému nástroji přizpůsobit chod na cílové platformě, kterou je v tomto případě Windows. To přineslo další komplikace, které budou uvedeny dále.

5.2.3 Problémy s přenosem souborů

Po výběru platformy bylo ještě nutné zvolit způsob přenosu souborů mezi strojem, kde probíhá samotný převod, a strojem, který o převod zažádá a očekává výsledky v podobě převedených souborů. Nabízelo se hned několik variant, avšak po zvážení a důkladném promyšlení celé situace bylo voleno mezi následujícími:

XML-RPC

Jedná se o komunikační protokol [5] umožňující komunikaci na principu volání vzdálených procedur. V jazyce Python je možné ji využívat po naimportování příslušné knihovny. Velkou výhodou je jednoduchá práce s rozhraním knihovny. Nevýhodou je ovšem nepřítomnost podpory pro přenos souborů, což je pro naše účely nezbytné. I když tedy není tento modul primárně stavěný na přenášení různých souborů, přesto to lze. Obsah souboru se celý zkopíruje do proměnné a tu už je možné korektně přenést. Bohužel z toho plyne několik nevýhod, například omezení velikosti přenášeného souboru nebo neefektivita takovéhoto způsobu přenosu.

Samba

Samba [9] je svobodná implementace síťového protokolu SMB (Server Message Block) používaného především pro vzdálený přístup k souborům (sdílení) v systémech Microsoft Windows. Velká výhoda takového řešení je zajisté v odstínění uživatele od přenosů souborů a komplikací s tím se pojících. Z uživatelského pohledu se celá věc jeví jako pouhé nakopírování souborů odněkud někam jinam. Možnou nevýhodou by mohla být snad jen počáteční konfigurace, která nemusí být vždy zcela triviální.

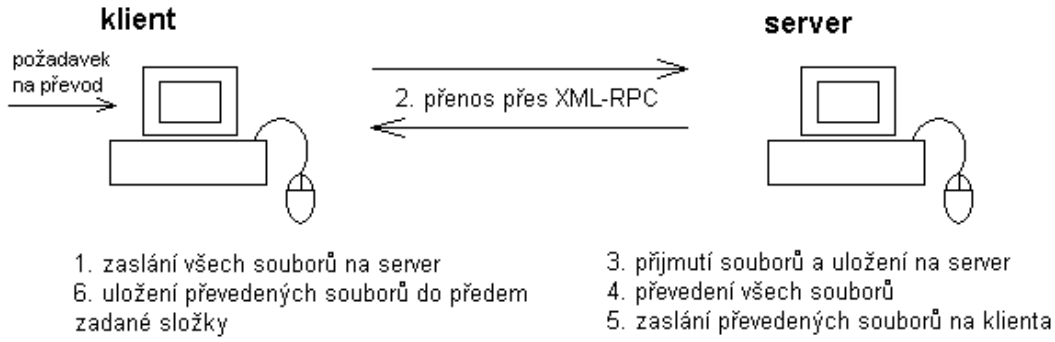
Zvolené řešení

Jako finální rozhodnutí byl nakonec vybrán přenos pomocí XML-RPC. To může být poněkud matoucí z pohledu nevýhod popsaných výše, avšak při zvážení, pro jaké účely je třeba a jaké soubory se budou přenášet, dostatečně vyhovovalo všem požadavkům. K výběru tohoto řešení lze přičíst také to, že nejprve byla vytvořena testovací verze právě na principu XML-RPC, aby se ověřilo, jestli všechna rozhodnutí (převodní software, platforma, přenos) mohou spolupracovat tak, jak se očekávalo. Protože otestování ukázalo, že zvolená cesta opravdu funguje a splňuje všechna očekávání, bylo toto řešení ponecháno, optimalizováno a uvedeno do provozuschopného stavu.

5.3 Realizace

V této sekci budou postupně probrány všechny části, které byly vybrány do procesu převodu, a u každé takové části bude nastíněno konkrétní řešení popřípadě její konečná implementace. Vše postupně v pořadí: převodní nástroj samotný a jeho komunikace s okolím,

některé komplikace cílové platformy v kombinaci s nástrojem a naposledy přenos souborů s články k převodu a zpět. Na obrázku 5.2 je naznačen celý postup převodu článků.



Obrázek 5.2: Celý proces převodu

5.3.1 Práce s OmniPage

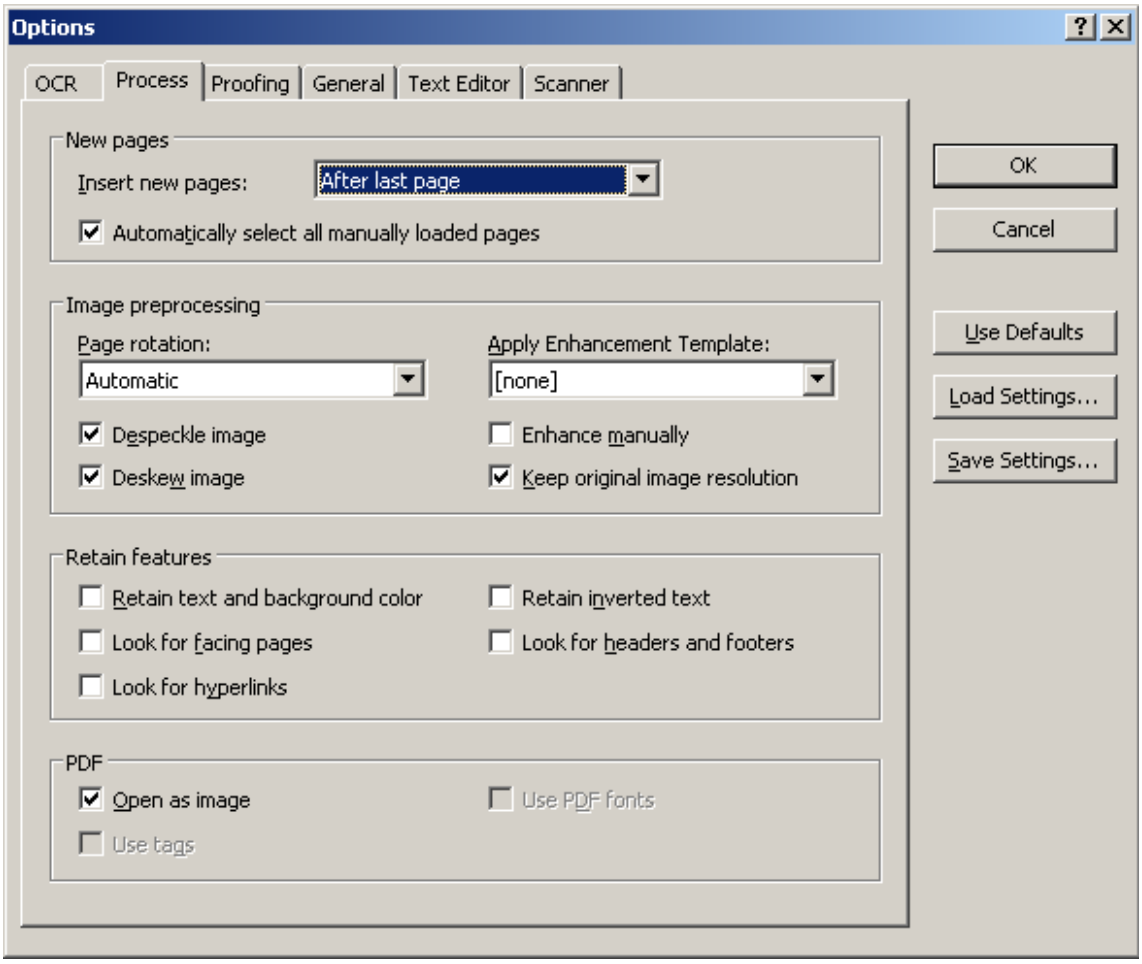
Jelikož je veškerá práce stále v souvislosti s OmniPage, není toto jediná kapitola, kde o něm nebo jeho rozhraní je zmínka. Tato kapitola je zaměřena na samotný převod, nastavení parametrů převodu a výstupů v podobě převedených článků.

Vstupní data

Jako vstup je možné zvolit soubory mnoha formátů. Nejdůležitějším typem, který je pro naše účely využit, je pdf soubor. Pracovat s ním lze dvěma odlišnými způsoby:

- Jako s klasickým pdf
- Jako s obrázkem

První volba může být sice užitečná a může výrazně napomoci samotnému převodu, ale jelikož se převádějí různé pdf, kde předem nemůžeme určit, zda není soubor nebo jeho vnitřní struktura například nějakým způsobem poškozená, je vždy zvolena volba práce s pdf jako s obrázkem. Tím je zajištěno, že korektní nebo jinak nepoškozené soubory půjdou převést bez problému a naopak ostatní budou upraveny tak, aby se dosáhlo co nejlepšího výstupu. Je nutné tuto volbu zatrhnout, jak je označeno na obrázku 5.3 dole.



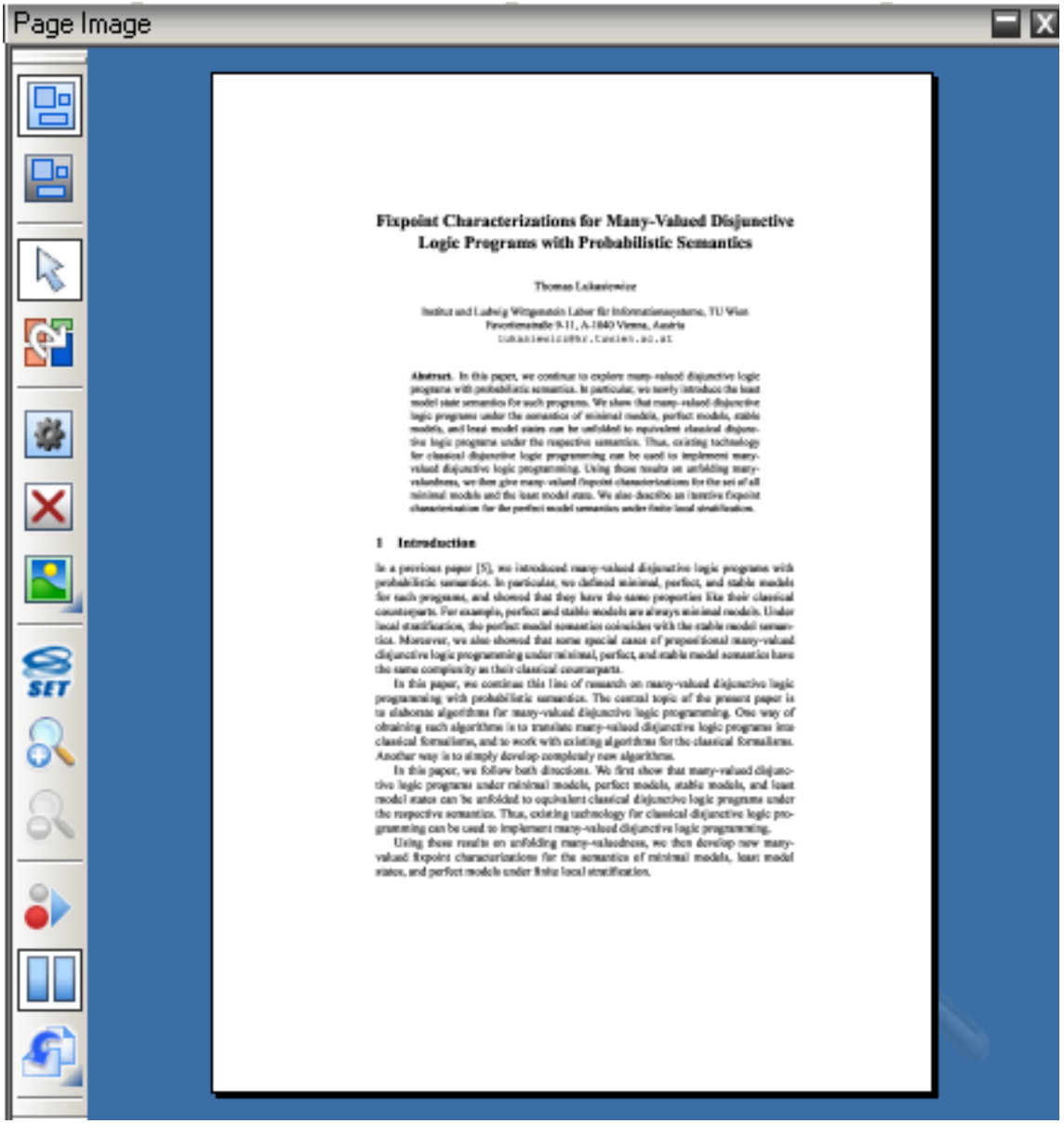
Obrázek 5.3: Otevření souboru jako obrázku

Převod

Zde je vhodné rozdělit převod na tři samostatné podfáze.

- Korekce před převodem
- Převod
- Korekce po převodu

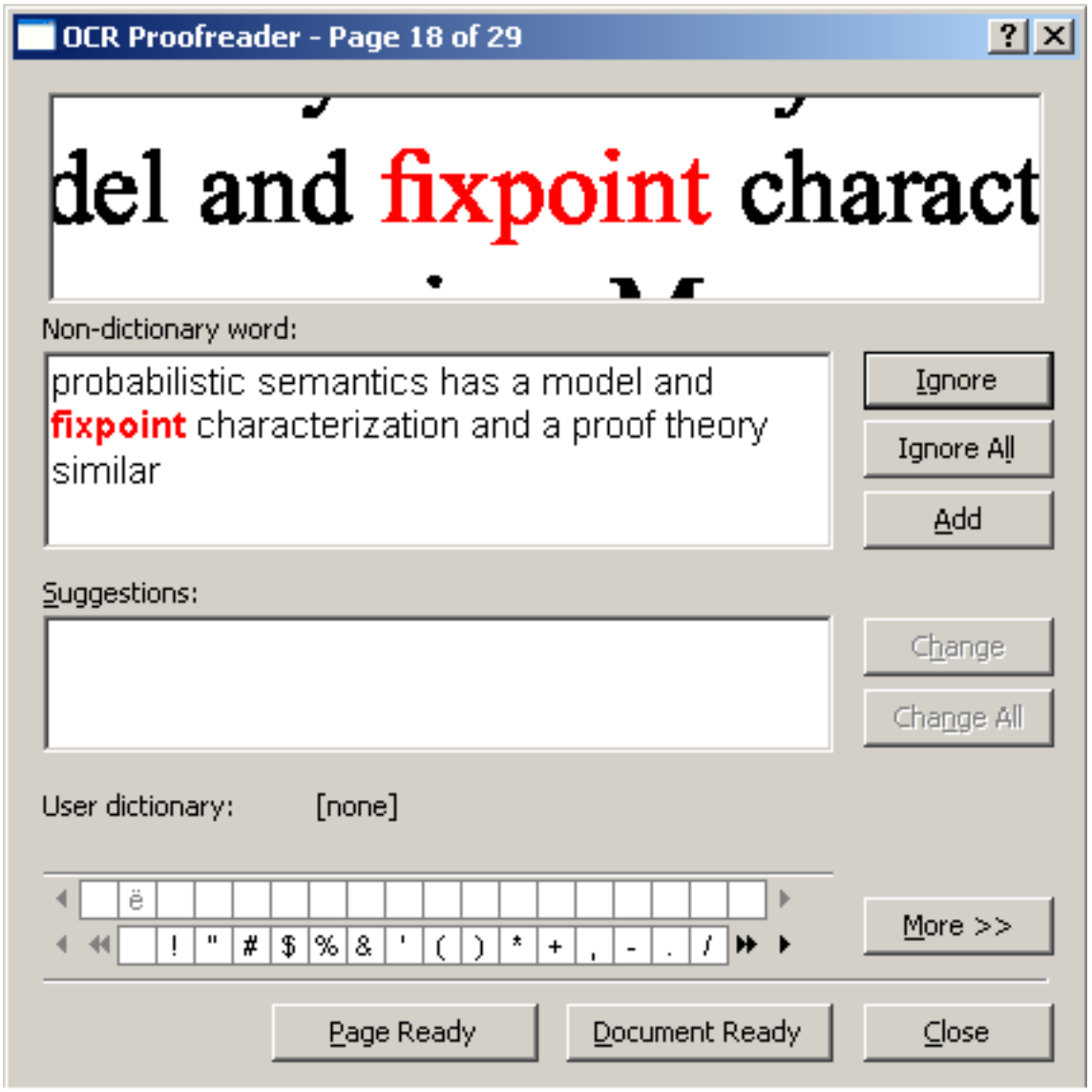
Ještě než dojde k samotnému převádění, je možné nahraný obrázek upravit. Jde především o změnu jasu, natočení, vyrovnání a další úpravy typické pro vylepšení obrazových dat. Samozřejmě čím lepších výsledků dosáhneme, tím je vyšší pravděpodobnost korektnosti výstupu po převodu. Zde obrázek 5.4 znázorňuje panel nástrojů (vlevo), pomocí kterého lze dokument upravovat.



Obrázek 5.4: Možná korekce dokumentu jako obrázku před převodem

Trvání samotného převodu je úměrné délce dokumentu a potom v neposlední řadě různým nastavením. Přitom se převádí všechny nahrané dokumenty.

Korekce po převodu je velice užitečný nástroj. Nejen že si uživatel může doupravit některé nepřesnosti ve výstupu, ale také dochází jistým způsobem k učení OmniPage. Proto když podruhé narazí na podobný problém, s velkou pravděpodobností použije nějakou uživatelovu korekturu, aniž by zobrazil kritické místo jako chybné. Možnost korektury výstupu po převodu lze provést v okně na obrázku 5.5.



Obrázek 5.5: Možná korekce výsledku převodu

Tyto vlastnosti mají však společné to, že je nutná přítomnost člověka a jeho zásahů do převodu. Protože ale celý proces má být automatický, byla využita možnost některé fáze provést automaticky a jiné přeskočit tak, aby nebyla nutná žádná uživatelská interakce. V konečném důsledku se jeví potom tyto tři fáze jako jedna jediná.

Výstupní data

I výstup je možné uložit do množství různých formátů, avšak pro nás je nejzajímavější uložení do souboru s prostým textem. Takových možností je více a liší se v použitém kódování nebo v uložení se řádky zalomenými či bez řádkových zlomů.

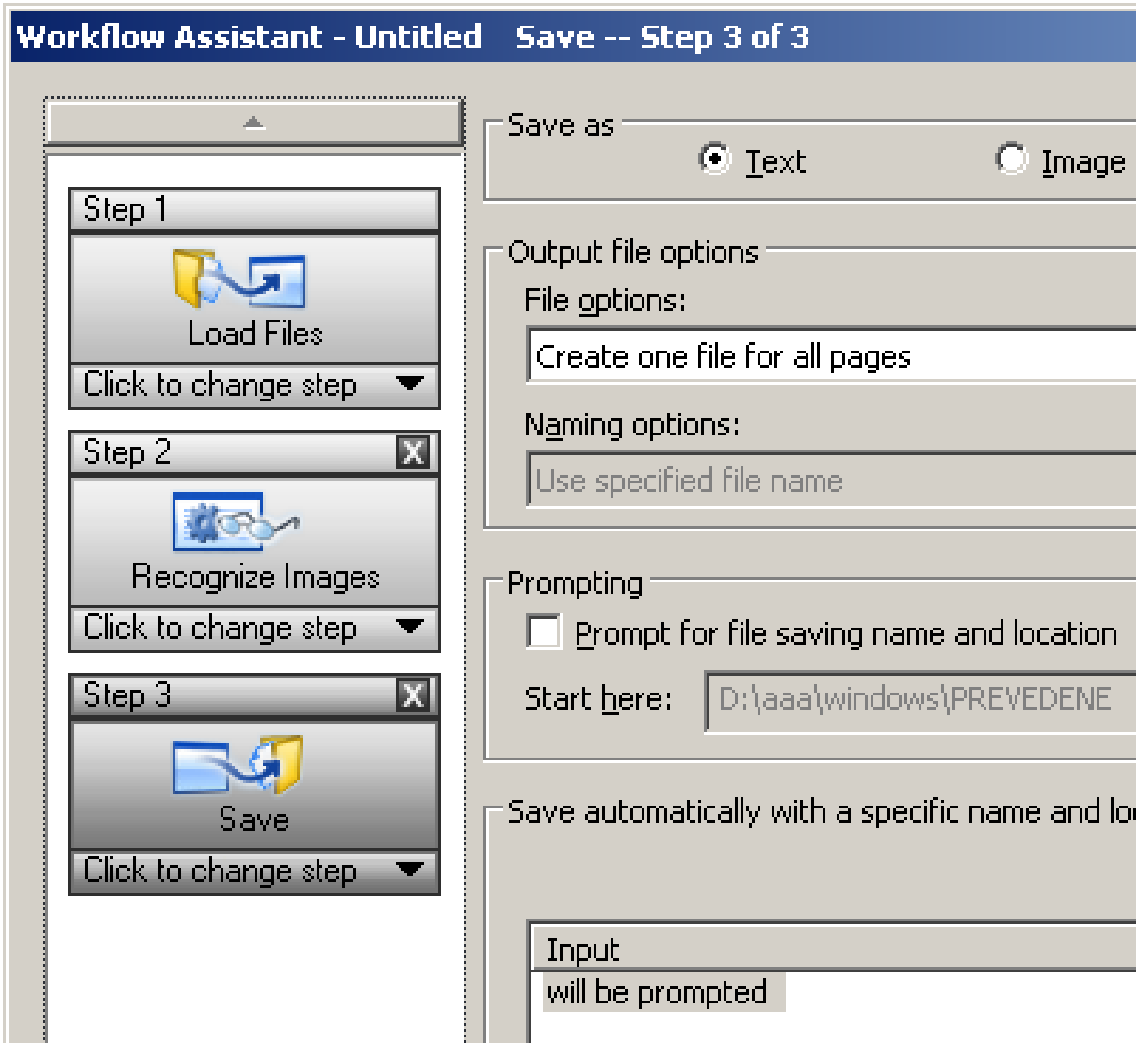
Pro nás zůstala jako nejlepší volba soubor s kódováním Unicode a zalamováním řádků. Ta byla také použita do výsledného řešení, protože je potřeba zajistit korektnost znaků nejen anglických textů, ale také například českých článků.

5.3.2 Rozhraní OmniPage na Windows

Po objasnění práce OmniPage je dále nutné určit, jakým způsobem budou v rámci automatického zpracování na vstup dodávány jednotlivé soubory s články a jakým způsobem bude poté možné detekovat dokončení převodu jednotlivých článků a vhodným způsobem na vzniklou událost reagovat. Nejprve je ale nutné nastavení OmniPage do režimu automatické práce.

Nastavení Workflow a Job

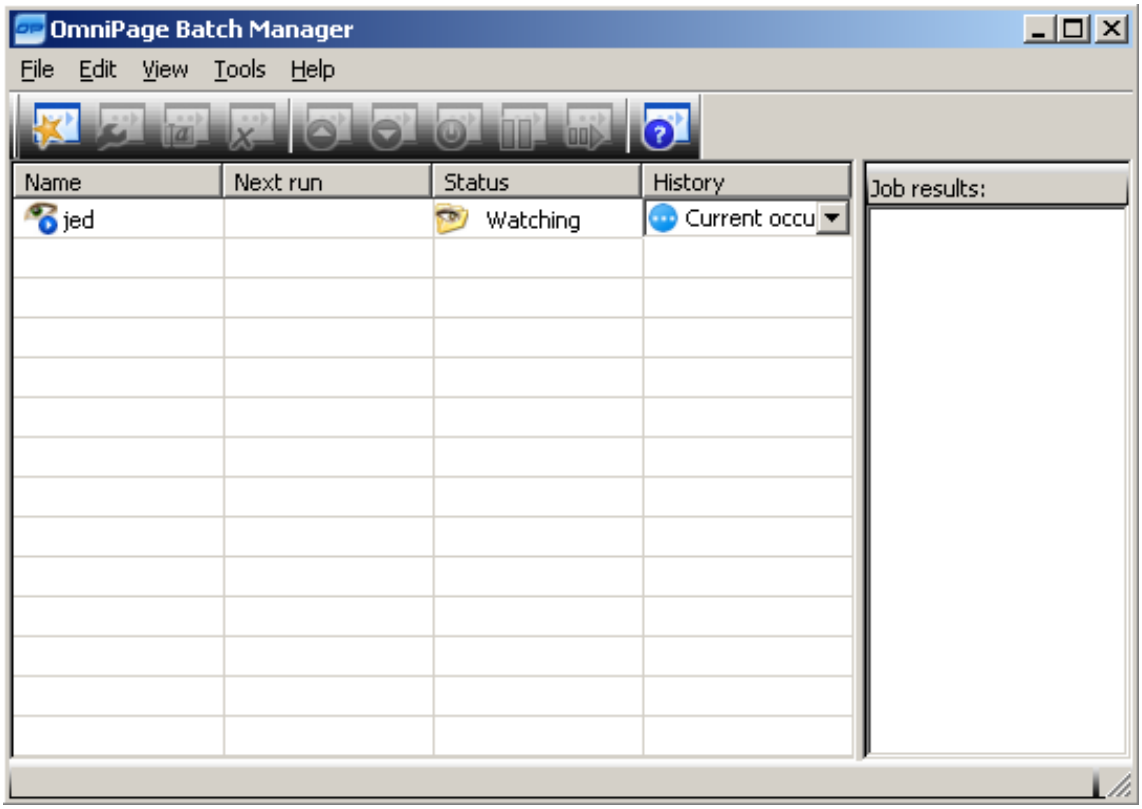
Ze všeho nejdříve je nezbytné nastavit takzvaný workflow. Nejedná se o nic jiného, než provedení výše zmíněných kroků převodu tak, jak mají jít za sebou. V rámci této činnosti je nutné také nastavení všech voleb, které vyžadují jakoukoli interakci s uživatelem, do stavu, kdy se úloha vykoná automaticky. Workflow není vlastně nic jiného než zabalení všech jednotlivých úkonů dohromady. Podrobnější popis je součástí uživatelského návodu [16]. Ukázka jednotlivých přednastavených kroků je na obrázku 5.6 vlevo.



Obrázek 5.6: Workflow: Nalevo jsou jednotlivé kroky tak, jak budou vykonány za sebou

Jako druhý krok je nutné nakonfigurovat takzvaný job. To je možné při spuštění OmniPage Agenta. Jedná se o správce všech jobů, které slouží k nastavení převodů, jež se mají spustit nějakým způsobem automaticky bez přítomnosti uživatele.

Zde využívám vlastnost OmniPage nazvanou watched folder (v překladu něco jako hlídaná složka). Dopředu totiž není jasné, kdy má převod proběhnout. Využitím možnosti hlídané složky lze dosáhnout asynchronní činnosti, kdy do této složky přijde soubor s článkem, OmniPage to sám zjistí a spustí převod. Není tedy nutné nastavovat jakékoli časy začátků a konců převodů a elegantním způsobem se také zbavíme nutnosti ovládání nástroje za pomocí skriptů. Příklad OmniPage Agenta a v něm aktivovaného jobu na principu watch folder znázorňuje obrázek 5.7.



Obrázek 5.7: OmniPage Agent: Ukázka nastaveného jobu jed, který je založen na Watched Folder

Detekce převedených souborů

Bohužel OmniPage nedisponuje žádnou vlastností, která by upozornila na dokončení převodu jednotlivých souborů. Z pohledu jeho funkce to není ani potřeba. Pro naše účely bylo nutné vytvořit vlastní postup, který by jistou detekci umožňoval. Jako řešení byly zvoleny moduly naprogramované v jazyku Python přímo pro Windows. Ty obsahují funkce, kterým je možné pomocí parametrů sdělit, jaká složka má být sledována a jaké události v ní nastávající mají být zachyceny. Poté není problém vhodně nastavit parametry tak, aby se detekovalo vždy vytvoření nějakého souboru, což je to, co potřebujeme, když dojde k dokončení převodu takového souboru.

5.3.3 Přenos souborů klient/server

Celý postup přenosu souborů je založený na architektuře klient/server, kde klientem je program nebo skript žádající o převod a jako server zde vystupuje počítač, na němž je nainstalovaný převodní nástroj. Byly tedy vytvořeny dva skripty, jeden klientský zajišťující vyvolání převodu a přenos souborů a jeden serverový, který má na starosti příjem souborů, detekování jejich převedených částí a zpětný přenos zpět na klienta.

Klientská část

Klientská část je vytvořena jako samostatný modul, který obsahuje dvě funkce realizující převod. Obě přijímají seznam článků určených k převodu ve formě textového souboru, kde na každém řádku je zapsaný právě jeden článek. Jedna z nich poté vrací xml soubor, v němž jsou vypsány jednotlivé soubory i s jejich absolutní cestou, kam byly umístěny. Druhá funguje úplně stejně až na to, že vrácený xml soubor obsahuje pouze články, které byly převedeny korektně. Rozhodnutí, zda je nebo není článek převeden korektně, bude popsáno ještě v samostatné kapitole 6.

Serverová část

Protože probíhá veškerá komunikace přes XML-RPC, jednotlivé funkce, jež server obsahuje, jsou vykonávány až po zavolání této funkce z klienta. Vytvoření serveru je díky jednoduchosti XML-RPC snadné a plně dostačující.

Serverová část skriptu má tedy na starost přijmutí souborů pro převod a jejich uložení do již zmíněné watched folder. Pokud není OmniPage spuštěn, provede se jeho spuštění. Dále je kontrolována složka, v níž by se postupně měly objevovat převedené soubory. Ty jsou evidovány a pokud se zjistí, že došlo k převodu všech článků, dojde k odeslání souborů zpět na klienta do předem zvolené složky. Pokud by OmniPage nefungoval správně nebo "zamrzl", vyprší nastavený timeout a na klienta se pošle informace o nemožnosti provedení převodu.

Určité problémy byly odhaleny při testováni převodu některých pdf článků chráněných heslem. V takovém případě OmniPage zobrazí okno a nechává uživatele vybrat ze tří možností. To je dosti nepříjemné, protože jakákoli interakce uživatele s počítačem neexistuje a pokud nedojde k odkliknutí tlačítka na ignorování této události, soubor se nepřevede a celý převod potom uvázne v situaci, kdy se čeká na dokončení převodu, které ale nikdy nenastane (tzv. uváznutí).

Na vyřešení této situace byl vytvořen skript za pomoci AutoIt [4], což je volně dostupný skriptovací jazyk pro automatické skriptování s Windows GUI. V tomto případě lze tedy jednoduchým skriptem detekovat zobrazení okna a odkliknout příslušné tlačítko. Na podobném principu byl vytvořen i skript, který hlídá, zda je OmniPage spuštěn.

Kód kostry skriptu pro server je uveden ve výpisu 5.1.

```
1
2 def main():
3   # vytvoreni serveru
4   s = SimpleXMLRPCServer((nazev_serveru, port))
5   print 'Server je pripraven na portu', port
6
7   # registrovani funkci, ktere lze z clienta volat
8   s.register_function(ulozeni_clientskeho_souboru, 'posli_soubor')
9   s.register_function(ziskej_prevedene_soubory, 'cekej_na_prevod')
10  s.register_function(posli_soubor_clientovi, 'prijmi_soubor')
11  s.register_function(spusteni_omnipage, 'spust_omnipage')
12  s.register_function(znovu_priprav_server, 'vycisti_server')
13
14  # server bezi v nekonecne smycce
15  s.serve_forever()
16
17 if __name__ == "__main__":
18   main()
```

Listing 5.1: Ukázka kostry skriptu pro server

Kapitola 6

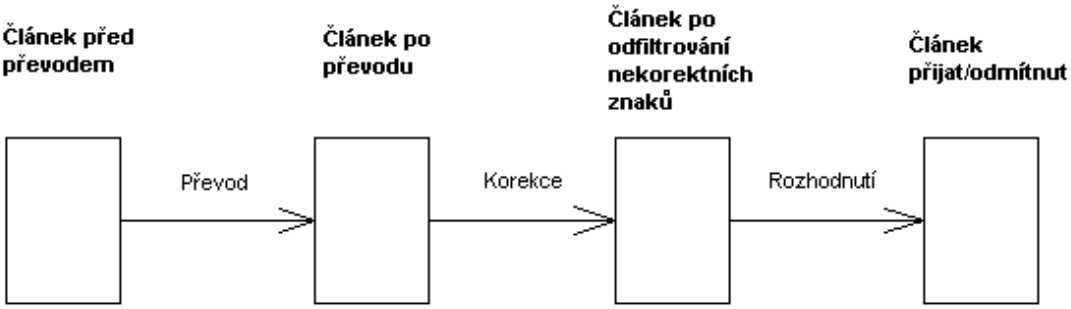
Detekce správnosti převedených souborů

Poslední fází v celé posloupnosti jednotlivých kroků při převodu problematických článků na prostý text je vykonání rozhodnutí, které nám sdělí, zda-li byl daný převod úspěšný nebo zda-li převedený článek obsahuje nezanedbatelné množství jistých nepřesností. Vše záleží na použitém převodním nástroji a jeho silných a slabých stránkách.

Nástroj OmniPage zvládá většinu převodů velice dobře a často splňuje neskromné požadavky na výstupní data. Jak již bylo řečeno v kapitole 4.1, která se zabývá vlastnostmi různých nástrojů, patří mezi jeho slabiny hlavně různorodé matematické zápisy, jakými jsou rovnice, výrazy, vzorce a další podobné konstrukce. Ve výstupním souboru se to projeví snahou o nahrazení některých řeckých znaků běžnými písmeny abecedy, záměnou za některé netypické znaky a celkově je celý pokus o převod těchto prvků neakceptovatelný.

Jako možné východisko se jeví kontrola výsledků převodu. Tady je ale nutné si vystačit pouze v rámci převedeného souboru a zkontrolovat, jestli takový výstup vyhovuje či ne. Zdůrazněna by měla být hlavně chybějící možnost použít například nějaké informace z převáděného pdf souboru, s kterými by bylo možné porovnat obsah výsledného textového souboru. Existuje sice nástroj, který umí z pdf souboru vyextrahovat nějaké statistické informace typu počet slov celkem, počet slov na stránku a podobně, ale jelikož zpracovávané soubory se řadí mezi problematické, není tuto možnost většinou možné využít.

Práce na výstupu z převodního nástroje byla rozdělena na dvě části. První se zabývá drobnou korekcí a druhá rozhoduje o použitelnosti takového výstupu. Převedený dokument musí projít oběma fázemi, jak ukazuje obrázek 6.1.



Obrázek 6.1: Jednotlivé etapy, kterými článek prochází

6.1 Korekce převedených souborů

Korekcí je tu myšleno hlavně nalezení a vymazání těch znaků, které by se v daném článku neměly vyskytovat. Jedná se především o symboly jiné znakové sady, než je například česká nebo anglická. Dále to může být také přemíra znaků typu copyright a dalších.

Na tuto činnost byl vytvořen skript v jazyku Python, který je na práci s textovými daty vhodný. Jako povolené byly vybrány na základě testů na určitém vzorku článků následující z výpisu 6.1.

```
1 import string
2
3
4 pripustne_znaky = ceske + string.ascii_letters + string.punctuation +
5 + string.whitespace + string.digits + '\x84\x97'
```

Listing 6.1: Povolené znaky v textu

kde,

- ceske – obsahuje velké a malé znaky české abecedy obsahující diakritiku
- string.ascii_letters – obsahuje písmena anglické abecedy
- string.punctuation – obsahuje základní znaky vyskytující se na klávesnici (například !,.* a další)
- string.whitespace – obsahuje všechny bílé znaky
- string.digits – obsahuje čísla 0 — 9
- \x84 – znak uvozovek dole
- \x97 – znak pomlčky (delší než znak mínus)

Výstup skriptu obsahuje původní převedený článek, ve kterém ale již nejsou nepovolené znaky a chybí zde také všechna slova, která takové znaky obsahovala.

6.2 Rozhodnutí o správnosti převedených souborů

Po předchozí filtraci je třeba učinit rozhodnutí, zda článek dále obsahuje jiná nekorektní slova nebo různým způsobem netypicky namíchané znaky, což je jeden z dalších rysů špatně převedeného textu. Skript na rozhodnutí o správnosti dokumentu je také napsán v jazyku Python a tvoří s předchozím jeden celý zdrojový kód.

6.2.1 Označení nekorektních slov

Není jednoduché odhalit některé nesrovnalosti textu, pokud neznáme kontext, ve kterém je dané slovo nebo i slovní spojení uvedeno. V tomto skriptu však není takový kontext brán v úvahu a analýza probíhá pouze na základě vyhodnocení konkrétního slova. Slovo je zde bráno jako cokoli oddělené bílým znakem, tedy i jediný znak. Slova považována za nekorektní jsou následující:

- slova obsahující velké písmeno jinde než na začátku a zároveň dlouhá (nemá vliv na zkratky)
- slova obsahující určité znaky jinde než na začátku nebo konci

Určité znaky reprezentuje množina znaků uvedená výše jako string.punctuation, ze které jsou ovšem vyjmuty symboly podtržítka a pomlčky. Bylo tak rozhodnuto opět na základě výsledků testů s omezeným množstvím článků.

6.2.2 Vyhodnocení statistik

Po projití celým textem a nasbírání dat o tom, která slova jsou korektní a která ne, je možné provést rozhodnutí o správnosti článku. Experimentálně byly vybrány určité oblasti, a to následující:

1. Poměr odstraněných slov k počtu slov celkem při filtraci textu
2. Poměr slov označených jako nekorektní k počtu slov celkem
3. Poměr jednopísmenných a dvoupísmenných slov k počtu všech slov
4. Průměrná délka slova
5. Poměr počtu znaků k počtu slov

U bodů 4 a 5 byly použity poznatky autora na interní wiki-stránce [10] zabývající se podobnou problematikou.

Pokud všechny oblasti budou spadat do nějakých pevně zvolených mezí, je takový článek označen za korektní a může být dále zpracován. V opačném případě byla v článku nejspíše přemíra různých matematických konstrukcí, které se ani po pokusu o jejich odfiltrování nepodařilo uspokojivě odstranit. Problém by také mohl nastat v případě převodu asijského jazyka nebo jiného, s kterým nebylo počítáno.

Kapitola 7

Závěr

Tato práce se zabývá převodem vědeckých článků do prostého textu. Úkolem bylo analyzovat, proč u některých článků takový převod selhává. Celá práce je přitom příspěvkem do projektu ReReSearch, jímž se zabývá Skupina zpracování přirozeného jazyka. Protože po převodu souborů s články dochází k jejich indexaci, bylo nutné zajistit takový proces, na jehož konci by bylo co nejvíce článků, jejichž obsah by co nejvíce odpovídal původnímu textu. Tím se zvýší pravděpodobnost, že se v následném procesu indexace nebudou indexovat nesmyslná slova. Čistou podobou je zde myšleno, že takový článek nebude obsahovat různé nežádoucí části textu, které mohou vzniknout v situacích, kdy konkrétní převodní nástroj není na některé vstupy dostatečně vybaven.

Nejprve bylo nutné ručně analyzovat množinu článků, jejichž převod dopadl neúspěchem, a vysledovat, z jakých důvodů se tak děje. Závěr z tohoto byl vyvozen v kapitole 3, jež obsahuje výčet několika problematických jevů. Dalším postupem bylo zkoušení různých dostupných nástrojů jednoduchých či robustnějších, volně dostupných či testovacích verzí, specializovanějších či obecněji zaměřených. Přitom bylo zkoumáno, jak si každý poradí právě s uvedenými problémy. Kandidáti s nejlepšími výstupy byli poté seřazeni podle priorit a vše bylo konzultováno s vedoucím bakalářské práce. Jako výsledné řešení se jevilo nejlepší použít nástroj OmniPage Professional od firmy Nuance. Jednak se řadí mezi nejlépe hodnocené nástroje ve svém oboru na trhu a jednak byl školou již zakoupen, a tak nic nebránilo testování při další práci.

Dále bylo nutné vyřešit komunikaci, která by zajistila převod článků na PC s nainstalovaným převodním nástrojem. K tomu se pojí potřeba pracovat s nástrojem, získat výsledky převodu a zaslat je zpět k dalšímu zpracování. Jako řešení se nabízelo vytvoření komunikace na principu klient/server. Byly tedy vytvořeny dva skripty zajišťující vše potřebné pro výše zmíněné kroky.

Jako posledním krokem byla nutnost rozhodnout, zda jsou převedené články korektní, a tedy vhodné k dalšímu zpracování, nebo zda převod nebyl úspěšný. O to se opět stará skript, který přijme výsledky převodu a na výstupu nabídne korektně převedené články.

Jako u většiny různých projektů i zde by se dalo pracovat na dalších možnostech, jak výsledky převodů dále zlepšit a pomoci tak maximalizovat úspěšnost systémů, které s převedenými články pracují. Jednou z možností je použít nástroj, jehož výstup by nebylo nutné dále upravovat. Není vyloučené, že někdy v budoucnu bude takový nástroj dostupný. Do té doby je potřeba si poradit jiným způsobem. Dalším vylepšením by mohl být propracovanější systém kontroly výstupů převodu. Zde by se možností a kombinací našla celá řada. Bylo by možné například pracovat s jednotlivými slovy v rámci kontextu, v jakém se vyskytují, nebo kontrolovat daná slova na přítomnost ve slovníku určitého jazyka. Pokud

by bylo nutné přenášet velké soubory (řádově desítky až stovky MB), muselo by dojít ke změně systému přenosu souborů. Implementované řešení pomocí protokolu XML-RPC by bylo v takovém případě neefektivní nebo zcela nepoužitelné.

Literatura

[1] Adobe Systems: *PostScript language reference*. Addison-Wesley, 1999, ISBN 0-201-37922-8.

[2] Adobe Systems: *PDF reference*. Adobe, 2005, ISBN 0-321-30474-8.

[3] Burke, S. M.: *RTF pocket guide*. O'Reilly & Associates, Inc., 2003, ISBN 0-596-00475-3.

[4] Flesner, A.: *AutoIt v3: your quick guide*. O'Reilly Media, Inc., 2007, ISBN 978-0-596-51512-6.

[5] Laurent, S. S.; Johnston, J.; Dumbill, E.: *Programming web services with XML-RPC*. O'Reilly & Associates, Inc., 2001, ISBN 0-596-00119-3.

[6] Microsoft: Word 97-2007 binary file format (.doc) specification [online]. `http://download.microsoft.com/download/0/B/E/0BE8BDD7-E5E8-422A-ABFD-4342ED7AD886/Word97-2007BinaryFileFormat(doc)Specification.pdf`, 2007 [cit. 2010-5-15].

[7] Rice, S. V.; Nagy, G.; Nartker, T. A.: *Optical character recognition*. Kluwer academic publishers, 1999, ISBN 0-7923-8492-X.

[8] Shamoo, A. E.; Resnik, D. B.: *Responsible conduct of research*. Oxford University Press, Inc., 2009, ISBN 978-0-19-536824-6.

[9] Ts, J.; Eckstein, R.; Collier-Brown, D.: *Using samba*. O'Reilly & Associates, Inc., 2003, ISBN 0-596-00256-4.

[10] Vácha, P.: Rrs article2txt [online]. `https://merlin.fit.vutbr.cz/nlp-wiki/index.php/GVP:Převod_publikací_do_textového_tvaru_a_jejich_indexování`, [cit. 2010-5-15], interní nlp-wiki stránka.

[11] WWW stránky: Gnu general public licence [online]. `http://www.gnu.org/licenses/gpl.html`, 2007-6-29 [cit. 2010-5-15].

[12] WWW stránky: Abbyy FineReader [online]. `http://finereader.abbyy.com`, 2010 [cit. 2010-5-15].

[13] WWW stránky: ABC amber pdf converter [online]. `http://www.processtext.com/abcpdf.html`, 2010 [cit. 2010-5-15].

[14] WWW stránky: Infty project [online].
http://www.inftyproject.org/en/index.html, 2010 [cit. 2010-5-15].

[15] WWW stránky: Nitro pdf professional [online].
http://www.nitropdf.com/index.asp, 2010 [cit. 2010-5-15].

[16] WWW stránky: OmniPage 16 user guide [online].
http://www.nuance.com/imaging/resources/userGuides/OPUserguide, 2010 [cit. 2010-5-15].

[17] WWW stránky: OmniPage capture software developers kit [online].
http://www.nuance.com/imaging/omnipage/omnipage-csdk.asp, 2010 [cit. 2010-5-15].

[18] WWW stránky: OmniPage [online].
http://www.nuance.com/imaging/products/omnipage.asp, 2010 [cit. 2010-5-15].

[19] WWW stránky: PDF converter [online].
http://www.anypdftools.com/pdf-converter.html, 2010 [cit. 2010-5-15].

[20] WWW stránky: Solid converter pdf [online].
http://www.soliddocuments.com/products.htm?product=SolidConverterPDF, 2010 [cit. 2010-5-15].

[21] WWW stránky: Tesseract-ocr [online]. http://code.google.com/p/tesseract-ocr, 2010 [cit. 2010-5-15].

[22] WWW stránky: Gocr [online]. http://jocr.sourceforge.net, [cit. 2010-5-15].