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

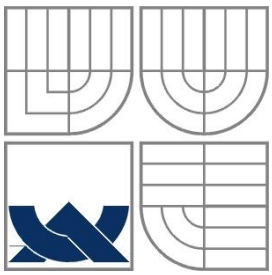
SYSTÉM PRO PŘEVOD SLOVNÍKŮ DO PODOBY LÉPE VYUŽITELNÉ PRO STROJOVÝ PŘEKLAD

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

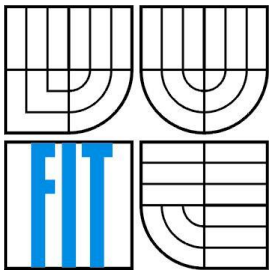
AUTOR PRÁCE
AUTHOR

MICHAL SCHOVAJSA

BRNO 2014



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

SYSTÉM PRO PŘEVOD SLOVNÍKŮ DO PODOBY LÉPE VYUŽITELNÉ PRO STROJOVÝ PŘEKLAD

DICTIONARY UP-TRANSLATION SYSTEM

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MICHAL SCHOVAJSA

VEDOUCÍ PRÁCE
SUPERVISOR

Doc. RNDr. PAVEL SMRŽ, Ph.D.

BRNO 2014

Abstrakt

Práce se zabývá zpracováním slovníků v elektronické podobě, jejich převodem do jednotné podoby a zejména problémy při tomto procesu vzniklými. Předmětem práce je vytvoření systému pro odstranění některých z těchto problémů s cílem usnadnit strojové zpracování slovníků. Nejprve jsou rozebrány jednotlivé problémy slovníků převedených do jednotné podoby. Poté se práce zabývá jejich řešením a tvorbou nástrojů k tomu určených. Závěrem jsou vyhodnoceny výsledky a úspešnost vytvořených nástrojů.

Abstract

The thesis concerns with the processing of dictionaries in electronic form, converting them into an unified form, and the problems arising in the process in particular. The subject of the work is to create a system for the elimination of some of these problems in order to facilitate machine processing of dictionaries. At first, different issues of dictionaries transferred into an unified form are concerned. Then, the thesis deals with the solution of these issues and the creation of tools for this purpose. Finally, the results and the efficiency of the instruments created are evaluated.

Klíčová slova

Slovník, strojově čitelný text, zpracování přirozeného jazyka, překlad podporovaný počítačem, rozšiřitelný značkovací jazyk, Lexical Markup Framework, řetězec, regulární výraz, morfologický analyzátor

Keywords

Dictionary, machine-readable text, natural language processing, computer-assisted translation, Extensible Markup Language, Lexical Markup Framework, string, regular expression, morphological analyzer

Citace

Schovajsa Michal: Systém pro převod slovníků do podoby lépe využitelné pro strojový překlad, bakalářská práce, Brno, FIT VUT v Brně, 2014

Systém pro převod slovníků do podoby lépe využitelné pro strojový překlad

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.

……
Michal Schovajsa
28. dubna 2014

Poděkování

Rád bych tímto poděkoval vedoucímu práce, panu doc. RNDr. Pavlu Smržovi, Ph.D., za odborné vedení a přínosné rady k realizaci této práce. Dále bych chtěl poděkovat své rodině a spolubydlícím za morální podporu.

Obsah

1 Úvod

Slovníky vznikly jako prostředek pro popis lidského jazyka, jeho slovní zásoby, vysvětlení významu jednotlivých slov i jejich spojení a v neposlední řadě pro popsání souvislostí mezi různými jazyky – vytvoření překladů. V průběhu historie vznikaly slovníky různého obsahu a zaměření a postupně se vyvíjely. Až do nedávna se ale vyskytovaly pouze v psané nebo tištěné podobě. Současný vývoj lidstva ale směřuje k čím dál širšímu využívání elektroniky a výpočetní techniky a tomu se přizpůsobují i slovníky. S moderní technikou také vznikají zcela nové možnosti, jak pracovat s jazyky, a tím pádem i využívat slovníková data. Jednou z takových možností je zpracování lidského, tzv. přirozeného jazyka počítačem – automatické rozpoznání významu, vytváření překladů apod. To ovšem vyžaduje data čitelná počítačem, ideálně v jednotné podobě. V dnešní době již existuje poměrně velké množství elektronických slovníků, které se ale navzájem liší, a proto byly vytvořeny prostředky pro převod slovníků do jednotné elektronické podoby. Při tomto procesu však vznikají další problémy. Řešením části těchto problémů se zabývá tato práce.

V úvodní kapitole je krátce vysvětlena definice pojmu slovník, poté se popisuje problém elektronických slovníků a rozdílnost způsobu uložení dat různých slovníků. Je vysvětlena potřeba jednotného formátu elektronických slovníkových dat z důvodu strojové čitelnosti a následného zpracování těchto dat. Dále jsou představeny prostředky pro uložení dat v jednotném standardizovaném formátu a problém převodu slovníků do takového formátu.

Následující kapitola se zabývá problémy a chybami vzniklými při převodu slovníků do jednotné podoby, analýzou konkrétních problémů, které jsou předmětem této práce, a návrhem jejich řešení.

Další kapitola rozebírá realizaci nástrojů pro řešení zmíněných problémů. Nejprve je popsána volba vhodných prostředků pro tvorbu těchto nástrojů. Poté se zaměřuje na jednotlivé nástroje navržené pro konkrétní problémy. Popisuje jejich funkcionalitu a způsob provedení požadovaných úkonů. Pro každý nástroj analyzuje výsledky a úspěšnost splnění požadavků.

V závěru jsou shrnuty výsledky jednotlivých systémů, jejich přednosti a nedostatky, jsou navrženy možné úpravy a rozšíření a shrnut celkový přínos práce.

2 Slovníky a jejich elektronická podoba

2.1 Definice pojmu slovník

Slovník je obvykle abecedně uspořádaná sbírka slovní zásoby, vysvětlující význam slov z různých hledisek. Vytvářením slovníků se zabývá obor lingvistiky zvaný lexikografie.

Slovníky se obvykle vyskytují v knižní podobě. V poslední době se však objevují i digitální slovníky, dostupné na CD nebo na internetu. [1, 2]

2.1.1 Typy slovníků

- slovníky výkladové (jednojazyčné) – jsou napsány celé v jednom jazyce, u každého slova lze nalézt informace ve stejném jazyku
- slovníky překladové (vícejazyčné, polyglotické) – slouží pro překlad z jednoho jazyka do druhého, ke slovům jednoho jazyka přiřazují překlad v druhém jazyce, často i s výslovností, komentáři, frázemi a příklady, nebo jinými doprovodnými informacemi. Některé větší překladové slovníky obsahují i druhou část, ve kterém jsou slova pro zpětný překlad z druhého jazyka do prvního. Tyto slovníky mohou být i specializované, například se omezovat jen na odborné termíny z některé oblasti. [1, 2]

2.2 Slovníky v elektronické podobě

Slovníky v elektronické podobě se vyskytují v různých nekonzistentních podobách. Ať už jako specializovaný software, případně součást nějakého software, tak v podobě elektronických dokumentů cíleně vytvořených nebo vzniklých převodem slovníků z podoby tištěné do elektronické např. s využitím technologie optického rozpoznávání znaků. Všechny tyto postupy vedou k vytvoření rozdílných formátů a způsobů uchování dat. Pro strojové zpracování slovníků je vhodnější zvolit jednotný formát, který je jednoznačně definovaný, neměnný a maximum informací uvádí explicitně. Existují standardy, které toto umožňují. Jedním z nich je Lexical Markup Framework (zkr. LMF) založený na eXtensible Markup Language (zkr. XML).

2.3 eXtensible Markup Language

eXtensible Markup Language (zkráceně XML, česky *rozšiřitelný značkovací jazyk*) je obecný značkovací jazyk. Navazuje na starší jazyk Standard Generalized Markup Language (zkr. SGML) – je jeho zjednodušenou podobou. Slouží k vytváření konkrétních značkovacích jazyků (tzv. aplikací) pro různé účely jako jeho podmnožin. Používá se i pro serializaci dat. Zpracování XML podporují různé nástroje a programovací jazyky.

Jeho účelem je zejména vytvoření formátu pro výměnu dat mezi aplikacemi a pro publikování dokumentů. Popisuje strukturu dat z hlediska obsahu, nedefinuje vzhled – ten může být určen dalšími podpůrnými prostředky. Další možností zpracování je transformace do jiného typu dokumentu nebo do jiné aplikace XML. [3]

2.3.1 Vlastnosti XML

2.3.1.1 Standardní formát pro výměnu informací s mezinárodní podporou

Elektronické dokumenty existují v rozdílných podobách a při jejich sdílení je tedy nutné využívat na všech stranách specializovaný software určený ke zpracování daného formátu, který je často zpoplatněn nebo nějak vázán licencemi, neexistuje na všech používaných platformách apod. Tyto komplikace vedly k potřebě vytvoření jednoduchého otevřeného formátu, který není svázán s konkrétní platformou nebo technologií. Takovým prostředkem může být právě XML, který je založen na prostém textu, je zpracovatelný libovolným textovým editorem a jeho specifikace je zdarma přístupná všem.

Již z počátku byl XML vyvíjen pro potřeby různých jazyků. Jako znaková sada se implicitně používá ISO 10646 (také Unicode), ale přípustné je libovolné kódování (pro češtinu např. Windows-1250 nebo iso-8859-2). To pak musí být v daném dokumentu konkrétně určeno. XML umožňuje vytvářet i vícejazyčné dokumenty. [3]

2.3.1.2 Vysoký informační obsah

Pomocí XML značek, tzv. tagů, je v dokumentu určován význam jednotlivých částí textu. Značkování je zaměřeno na informační efektivitu a jednoznačnost zapsaných informací, ne na prezentační efektnost. To lze s výhodou využít např. při prohledávání obsahu v závislosti na významu dat. Vzhled pro prezentaci obsažených dat je možno definovat např. pomocí kaskádových stylů. [3]

2.3.1.3 Definice struktury dokumentu a její kontrola

XML nedefinuje konkrétní značky, ty se určují v závislosti na aplikaci. Je možné je definovat v souboru DTD (Document Type Definition), který poté umožňuje automatickou kontrolu, zda vytvářený XML dokument definici odpovídá. DTD ovšem neumožňuje kontrolovat typy dat. Na vytvoření jednotného standardu, který tyto kontroly umožní, se v současné době se pracuje. Zatím se používají tzv. schémata vytvořená pro konkrétní aplikace. [3]

Program zpracovávající strukturu XML se nazývá *parser*. Existují dva nejčastější přístupy ke zpracování XML dokumentu:

- DOM parser (DOM = Document Object Model) vyrobí obraz XML dokumentu v paměti.
- SAX parser (SAX = Simple API for XML) postupně prochází XML dokument a vyvolává události, které je následně nutné zpracovat

2.3.1.4 Syntaxe XML

Aby byl dokument považován za správně strukturovaný (well-formed), musí splňovat následující předpoklady:

- Musí mít právě jeden kořenový (root) element.
- Neprázdné elementy musí být ohraničeny startovací a ukončovací značkou. Prázdné elementy mohou být označeny značkou „prázdný element“.
- Všechny hodnoty atributů musí být uzavřeny v uvozovkách – jednoduchých (') nebo dvojitých (") – vždy párové, na začátku i na konci shodné. Opačný pár uvozovek může být použit uvnitř hodnot.

- Elementy mohou být vnořeny, ale nemohou se překrývat; to znamená, že každý (ne kořenový) element musí být celý obsažen v jiném elementu.
- Jména elementů v XML rozlišují malá a velká písmena: např. „<Příklad>“ a „</Příklad>“ je pár, který vyhovuje správně strukturovanému dokumentu, pár „<Příklad>“ a „</příklad>“ je chybný. [3]

Jednoduchá ukázka XML:

```
<?xml version="1.0" encoding="UTF-8" ?>
<clovek jmeno="Michal" prijmeni="Schovajsa" vek="23">
  <adresa>
    <ulice>Bartošova</ulice>
    <cp>1199</cp>
    <mesto>Napajedla</mesto>
    <psc>76361</psc>
  </adresa>
</clovek>
```

Ukázka 2.1: příklad XML

2.4 Lexical Markup Framework

Lexical Markup Framework (zkráceně LMF) je standard pro *zpracování přirozeného jazyka* (natural language processing, zkr. NLP) a *strojově čitelné slovníky* (machine-readable dictionary, zkr. MRD). Patří do rodiny ISO (International Organization for Standardization) standardů ISO/TC37, které se zabývají přirozeným jazykem a jeho strojovým zpracováním. Účelem LMF je standardizace zásad a metod týkajících se jazykových prostředků v souvislostech vícejazyčné komunikace a kulturní rozmanitosti. Historie LMF sahá do roku 2003, kdy začal jeho vývoj. Současná specifikace byla oficiálně zveřejněna jako mezinárodní standard dne 17. listopadu 2008. Struktura LMF vychází z XML. [4]

2.4.1 Cíle LMF

Cílem LMF je poskytnout společný model pro vytváření a používání lexikálních prostředků, který řídí výměnu dat mezi těmito prostředky, a umožnit sloučení velkého počtu jednotlivých elektronických zdrojů k vytvoření rozsáhlých globálních elektronických zdrojů.

Typy jednotlivých konkretizací LMF mohou zahrnovat výkladové, dvojjazyčné nebo vícejazyčné lexikální prostředky. Stejné specifikace jsou určeny pro malé i velké slovníky, jednoduché i složité slovníky, pro písemnou i mluvenou lexikální reprezentaci. Rozsah výrazových prostředků se pohybuje od popisu morfologie, syntaxe nebo sémantiky až po překlad podporovaný počítačem. Pokrytí jazyků není omezeno jen na evropské jazyky, zahrnuje všechny přirozené jazyky. Rozsah podporovaných aplikací pro podporu NLP není omezen. LMF je schopen reprezentovat většinu slovníků, včetně WordNet, EDR a PAROLE slovníků. [4]

2.4.2 Specifikace LMF

Jak již bylo řečeno v úvodu, LMF je jeden z členů rodiny ISO/TC37 standardů. Normy ISO/TC37 jsou v současné době zpracovány jako specifikace na vysoké úrovni a zabývají se segmentací slov, anotacemi, strukturou vlastností, multimediálními kontejnery a slovníky. Tyto standardy jsou založeny na specifikacích nízké úrovně definujících kategorie dat, kódování písma a jazyků, kódy zemí, a Unicode. [4]

Struktura LMF vychází z XML a je možné ji specifikovat pomocí diagramů tříd UML, skládá se z následujících komponent:

- Jádrem je kostra struktury popisující základní hierarchii informací v lexikálním záznamu.
- Rozšíření jádra, která popisují opětovné použití jádrových položek ve spojení s dalšími komponenty potřebnými pro konkrétní lexikální zdroje. Specificky se zaměřuje na morfologii, MRD ,syntaxi a sémantiku NLP, vícejazyčné zápisy NLP, morfologické vzory NLP, vzory pro víceslovné výrazy a vzory pro omezení obsahu výrazu. [4]

2.4.2.1 Jednoduchý příklad

V následujícím příkladu je lexikální záznam spojen s lemmatem „slovo“ a dvěma jeho formami – jednotné a množné číslo. Kódování jazyka a jazyk jsou nastaveny pro celý dokument:

```
<LexicalResource dtdVersion="15">
  <GlobalInformation>
    <feat att="languageCoding" val="UTF-8"/>
    <feat att="language" val="cz"/>
  </GlobalInformation>
  <Lexicon>
    <LexicalEntry>
      <feat att="partOfSpeech" val="commonNoun"/>
      <Lemma>
        <feat att="writtenForm" val="slovo"/>
      </Lemma>
      <WordForm>
         <feat att="writtenForm" val="slovo"/>
         <feat att="grammaticalNumber" val="singular"/>
      </WordForm>
      <WordForm>
        <feat att="writtenForm" val="slova"/>
        <feat att="grammaticalNumber" val="plural"/>
      </WordForm>
    </LexicalEntry>
  </Lexicon>
</LexicalResource>
```

Ukázka 2.2: příklad LMF

Prvky (značky) Lexical Resource, Global Information, Lexicon, Lexical Entry, Lemma, and Word Form definují strukturu slovníku. Prvky languageCoding, language, partOfSpeech, commonNoun, writtenForm, grammaticalNumber, singular, plural určují vlastnosti hesla. [4]

2.5 Převod elektronických slovníků do jednotné podoby

Lexical Markup Framework tedy specifikuje jednotný formát XML pro uložení slovníkových dat. Slovníky je ale nejdříve nutné do tohoto formátu převést. Při převodu je třeba vzít v úvahu, z jakého zdroje pochází slovník, který se bude převádět.

Buďto se může jednat o digitální slovník, který už má data uložena v elektronické podobě dané nějakým obecným standardem, případně vlastním proprietárním formátem. Zde je nutné vyřešit problém struktury obsahu daného formátu a poté je možné vytvořit odpovídající strukturu LMF.

Dále může jít o slovník tištěný, který je nejprve nutné převést do elektronické podoby. Výhodou je možnost zvolit vlastní formát výsledné elektronické podoby tak, aby byl převod co nejsnazší. Převod do elektronické podoby je obvykle realizován skenováním do bitmapového formátu a následným provedením OCR, tedy optického rozpoznávání znaků, jehož výstupem může být např. dokument ve formátu *.docx (Open XML), kdy data již jsou uložena v XML formátu a je tedy možné aplikovat XML transformace. [5]

Tato práce se dále bude zabývat slovníky převedenými z různých zdrojů do jednotného formátu podle specifikace LMF. Pokud bude zmiňován pojem slovník, bude tím myšlen právě slovník ve formátu LMF, pokud nebude uvedeno jinak.

3 Problémy strojové čitelnosti slovníků

I přes to, že existují postupy pro převod různých podob slovníků do jednotné formy, je stále nutné vypořádat se s dalšími problémy. Ty jsou způsobeny jednak chybami při samotném převodu a také konvencemi používanými ve slovnících historicky z dob, kdy strojové zpracování nepřipadalo v úvah. Řešení některých těchto problému je hlavním tématem této práce.

Z problémů daných konvencemi slovníků se konkrétně jedná např. o používání zkratek a zástupek ve výkladových či příkladových částech nebo vysvětlivkách, kdy je v textu místo heslového slova použita nějaká zkrácená zástupná podoba čitelná člověku, ale znesnadňující strojové rozpoznání významu. Dále jde o zkrácený zápis výslovnosti, kdy se uvádí pouze část slova, kde se výslovnost odchyluje od psané podoby, což opět znesnadňuje strojovou čitelnost.

Jedním z problémů způsobujících potíže jak při převodu tak při následném zpracování je uvádění alternativních významů, synonym apod. v textu způsobem rozlišitelným člověkem (např. jednotlivé významy na řádku oddělené čárkami nebo středníky), ale obtížně detekovatelným při strojovém zpracování. Pak mohou už při převodu do LMF vznikat chyby, kdy jsou např. všechny alternativní významy obsaženy pod jednou značkou a při dalším zpracování se potom se značkou pracuje jako s jediným významem.

Dalším problémem znesnadňujícím strojovou čitelnost jsou chyby vzniklé při procesu skenování a OCR. Jedná se o chyby způsobené např. nekvalitním tiskem fyzického slovníku, nedokonalým skenováním nebo nedokonalostmi OCR systému. Výsledná elektronická podoba potom obsahuje např. nežádoucí nebo chybné znaky nebo chyby v samotné struktuře dokumentu, kdy se poté obtížně určují souvislosti. Takto vzniklý chybný obsah je poté velmi náročné, a někdy zcela nemožné, opravit automatizovaně. Obvykle to vyžaduje lidský zásah.

Chyby do výsledné podoby LMF může zanést i nedokonalý převáděcí systém, který např. špatně zpracuje strukturu zdroje a výsledný obsah je pak umístěn ve špatných značkách.

Následující podkapitoly se zabývají analýzou výše zmíněných problémů v kontextu slovníků ve formátu LMF a návrhem řešení těchto problémů.

3.1 Sloučený obsah značek

Různé překlady, příklady atd. hromadně zapsané uvnitř jedné značky je třeba převést do podoby, kdy jednotlivé výrazy budou zapsané ve vlastních značkách. Tyto výrazy musí být oddělené jasně definovaným oddělovačem, případně množinou oddělovačů, např. čárkami, středníky, lomítky atd. Na základě těchto oddělovačů je pak možné ve slovníku vyhledat značky obsahující tyto oddělovače, rozdělit jejich obsah na jednotlivé výrazy a vytvořit kopie dané značky vždy s jedním výrazem.

Zápis ve slovníku může vypadat například takto:

```
<WordForm>
      <feat att="writtenForm" val="a hele!; a hleďme!; a hrome!; a vůbec!" />
</WordForm>
```

Vzhledem k rozdílné struktuře různých slovníků je problém automaticky určit, jakým způsobem, případně s jakou hloubkou zanoření značek se má pracovat, a obsah kterých značek se má prohledávat.

Jednou z možností je následující rozdělení:

```
<WordForm>
	<feat att="writtenForm" val="a hele!" />
	<feat att="writtenForm" val="a hleďme!" />
	<feat att="writtenForm" val="a hrome!" />
	<feat att="writtenForm" val="a vůbec!" />
</WordForm>
```

Další možností je:

```
<WordForm>
	<feat att="writtenForm" val="a hele!" />
</WordForm>
<WordForm>
	<feat att="writtenForm" val="a hleďme!" />
</WordForm>
<WordForm>
	<feat att="writtenForm" val="a hrome!" />
</WordForm>
<WordForm>
	<feat att="writtenForm" val="a vůbec!" />
</WordForm>
```

Pro lepší funkčnost by bylo vhodné umožnit určení konkrétní značky nebo kontextu, kde se může nacházet obsah, který se bude rozdělovat a stejně tak určení hloubky zanoření značek – jestli se bude pracovat jenom s danou značkou nebo i s nadřazenými apod. Mohlo by také docházet k nechtěným změnám při zpracování oddělovačů jinak běžně používaných v textu – např. při zadání "," jako oddělovače se rozdělí i souvětí v příkladech apod.

3.2 Užívání zkratek

Zkratky a zástupky za heslové slovo uvedené ve výkladech, vysvětlivkách a dalších částech slovníku je vhodné nahradit za jejich plné znění. Zkratky jsou ve slovnících používány v ustálených tvarech:

- zkratky ve tvaru „x-y“ kde „x“ je první písmeno daného heslového slova a „y“ změněná koncovka
- zkratky ve tvaru „x.“ kde „x“ je první znak heslového slova
- předpony (např. heslové slovo je „kopat“, v textu je obsažena předpona „za-“ – význam je „zakopat“)
- přípony (např. heslové slovo je „aeroplán“, v textu je obsažena přípona „-plánový“ – význam je „aeroplánový“)
- vysvětlivky, slovní druhy, určení oboru atd.

Jednoduchý příklad zápisu ve slovníku:

```
<Lemma>
	<feat att="writtenForm" val="průtočný" />
	<feat att="language" val="cz" />
</Lemma>
<WordForm><feat att="writtenForm" val="průtočný" /></WordForm>
<WordForm><feat att="writtenForm" val="-á" /></WordForm>
<WordForm><feat att="writtenForm" val="ne-" /></WordForm>
<WordForm><feat att="writtenForm" val="p. profil" /></WordForm>
<WordForm><feat att="writtenForm" val="p-á síla vody" /></WordForm>
<WordForm><feat att="writtenForm" val="p-é množství" /></WordForm>
```

Tyto zkrácené tvary zhoršují srozumitelnost textu, zejména při strojovém zpracování, proto je nutné nahradit je za plné znění ve správném tvaru. Základem je určení heslového slova pro dané slovníkové heslo. Poté je v obsahu značek možné vyhledat zkratky ve výše zmíněných tvarech a nahradit je za plné znění.

Nahrazení prostých zkratek ve tvaru „x.“, které zastupují heslové slovo v základním tvaru, a předpon by mělo být triviální záležitostí, protože se nemění tvar heslového slova. Zkratku je tak možné pouze nahradit heslovým slovem a dále nic neměnit.

Problémem je nahrazení přípon a koncovek, kdy se často mění i část heslového slova (několik koncových znaků). V takovém případě nelze koncovku nebo příponu pouze připojit k heslovému slovu. Je nutné zjistit, jaká část heslového slova se mění a tuto část slova nahradit tou upravovanou v příponě nebo koncovce.

Výstup by pak měl být následující:

```
<Lemma>
	<feat att="writtenForm" val="průtočný" />
	<feat att="language" val="cz" />
</Lemma>
<WordForm><feat att="writtenForm" val="průtočný" /></WordForm>
<WordForm><feat att="writtenForm" val="průtočná" /></WordForm>
<WordForm><feat att="writtenForm" val="neprůtočný" /></WordForm>
<WordForm><feat att="writtenForm" val="průtočný profil" /></WordForm>
<WordForm><feat att="writtenForm" val="průtočná síla vody;" /></WordForm>
<WordForm><feat att="writtenForm" val="průtočné množství" /></WordForm>
```

Zkratky zastupující slovní druhy, určení oboru apod. nelze nahradit odvozením z textu, ale tyto bývají používány v ustálených tvarech, takže je možné pro ně vytvořit tabulku přepisů, kterou se převod bude řídit.

3.3 Zkrácený zápis výslovnosti

Výslovnost zapsaná zkrácenou formou je těžce čitelná. Pro strojové zpracování je vhodnější uvedení úplné podoby. Ve slovnících se výslovnost často vyskytuje ve zkráceném tvaru, kdy je uvedena pouze část slova, kde se výslovnost odchyluje od psané podoby.

Ve slovníku je uvedeno např. heslo „abilympiáda“ a výslovnost „[-pijá-]“:

```
<WordForm>
	<feat att="writtenForm" val="abilympiáda" />
	<feat att="phoneticForm" val="-pijá-" />
	<feat att="grammaticalGender" val="feminine" />
</WordForm>
```

Pro strojové zpracování je vhodnější uvést výslovnost v úplné podobě, tedy „[abilympijáda]“:

```
<WordForm>
	<feat att="writtenForm" val="abilympiáda" />
	<feat att="phoneticForm" val="abilympijáda" />
	<feat att="grammaticalGender" val="feminine" />
</WordForm>
```

Zkrácený zápis výslovnosti je nutné mapovat na odpovídající psanou formu, tzn., že se zjistí, kterou část psané formy popisuje úsek výslovnosti. Tato část se poté nahradí zápisem výslovnosti a výsledný tvar se použije jako úplný tvar výslovnosti.

To je možné řešit jednak mapováním znaků úseku výslovnosti na psanou formu, které je řešeno hledáním krajních znaků úseku výslovnosti v psané formě. Při nalezení shody se úsek výslovnosti vloží do psané formy namísto původních znaků.

Další možností je využití tabulky přepisů jednotlivých znaků nebo skupin znaků, kdy se podle tabulky zápis výslovnosti převede tak, aby odpovídal psané formě, ve které se pak tento úsek vyhledá, a na jeho místo je vložen původní úsek výslovnosti.

Obě tyto metody je možné kombinovat. Pokud mapování selže, úsek výslovnosti se převede pomocí tabulky přepisů a mapování se zopakuje.

Při přepisu výslovnosti na začátku nebo konci slova může hledání shody probíhat postupným „ořezáváním“ počátečních, resp. koncových znaků, dokud není nalezena shoda úseku výslovnosti s částí psané formy.

3.4 Chybný obsah

Hledání chybného obsahu je vzhledem k rozsahu slovníků náročný úkol. Pro jeho realizaci je vhodné nejprve vytvořit podklady v podobě statistické analýzy slovníků. Jednou z možností je zjistit četnost výrazů v překladových, příkladových a dalších částech slovníků a ověřit, zda jsou tyto výrazy obsaženy jako hesla v daném slovníku, v případě překladových slovníků i ve slovníku obrácené.

Když je chybný obsah odhalen, je možné přistoupit k řešení, jakým způsobem tento obsah opravit. Chyby zanesené hromadně ve větších částech nebo i v celém slovníku je možné opravit nástrojem, který umožní nalezení a editaci obsahu značek s asistencí člověka. Uživatel takového nástroje zadá hledaný výraz, a výraz, kterým bude hledaný nahrazen. Poté se vyhledají všechny výskyty hledaného výrazu v obsahu slovníku a nahradí se požadovaným výrazem.

4 Převod slovníků do podoby lépe použitelné pro strojové zpracování

Tato kapitola se zabývá řešením problémů strojové čitelnosti slovníků popsaných výše. Nejprve rozebírá volbu vhodných prostředků použitých k tvorbě nástrojů pro řešení těchto problémů, poté se zaměřuje na jednotlivé nástroje určené k řešení konkrétních problémů.

4.1 Použité technologie

Slovníky, se kterými se bude pracovat, jsou ve formátu LMF, tedy v textové podobě. Proto je důležité pro tvorbu nástrojů zvolit prostředí, které nabízí dobrou podporu zpracování textových souborů a textu. Vzhledem k tomu, že LMF vychází z XML, bylo by možné použít k transformaci obsahu prostředky pro XML určené – např. eXtensible Stylesheet Language Transformations (zkr. XSLT), který slouží ke změně struktury a obsahu XML dokumentů deklarativním způsobem. Úpravy a transformace dané řešenými problémy jsou ale poměrně specifické a zpravidla vyžadují specifické drobné úpravy konkrétního obsahu spíše než transformace celého dokumentu. Z toho důvodu jsem se rozhodl od XSLT upustit a vytvořit vlastní systém pro analýzu a rozbor vstupního textu, který budou využívat jednotlivé nástroje pro načtení obsahu slovníku do vlastních struktur, a každý nástroj bude poté s daty pracovat vlastním způsobem. [6]

Pro takový úkol bylo třeba zvolit vhodný programovací jazyk schopný dobře pracovat s textem. Poměrně dobrou podporu pro zpracování textových dat nabízí většina moderních jazyků, ale, na radu vedoucího práce, jsem zvolil jazyk Python ve verzi 3, který nabízí opravdu široké možnosti jak ve zpracování samotného textu, tak v další práci s daty.

4.1.1 Python

Python je interpretovaný dynamický objektově orientovaný programovací jazyk. Je to hybridní jazyk (tzv. víceparadigmatický), to znamená, že umožňuje při psaní programů používat jak objektově orientované paradigma, tak procedurální a v omezené míře i funkcionální, podle toho, komu co vyhovuje nebo se pro danou úlohu hodí nejlépe. Python má díky tomu vynikající vyjadřovací schopnosti.

Kód programu je ve srovnání s jinými jazyky krátký a dobře čitelný. Jeho zápis je již z principu filosofie jazyka přehledný a jednoduchý.

Význačnou vlastností jazyka Python je produktivnost z hlediska rychlosti psaní programů. Týká se to jak nejjednodušších programů, tak aplikací velmi rozsáhlých.

Obsahuje velké množství standardních knihoven a interních funkcí, které dále usnadňují práci. Taková komplexnost s sebou ovšem nese také negativa. Je nutné vhodně volit použité postupy a metody. K řešení jednoho úkolu Python obvykle poskytuje několik rozdílných přístupů. Některé jsou uzpůsobeny pro lepší přehlednost kódu, ale nenabídnou maximální výkon. Pokud je na prvním místě efektivita využití výkonu počítače, je často třeba sáhnout k postupům jiným, než těm, které se nabízejí jako první volba a „nejschůdnější cesta“. [7, 8]

4.1.1.1 Řetězce a regulární výrazy

Python umožňuje pracovat s řetězci dvěma základními způsoby. Prvním z nich jsou standardní operace nad řetězci implementované třídou reprezentující řetězec. Jedná se o jednoduché operace jako vyhledání podřetězce, spojování a rozdělování řetězce, nahrazení podřetězce apod., které neumožňují používání zástupných znaků, vytváření vzorců atd. Na tyto pokročilé funkce je zde knihovna regulárních výrazů. V nástrojích budou využity obě možnosti – standardní operace tam, kde není nutné nebo se nevyplatí používat regulární výrazy z důvodu menší výpočetní náročnosti standardních operací, případně i jejich jednoduchosti oproti regulárním výrazům, které ale nabízejí výrazně větší možnosti při vyhledávání nebo nahrazování vzorců, možnost uložení pozice nalezené shody vzorce ve vstupním textu, vytvoření seznamu těchto nálezů apod. [7, 8, 9]

4.1.1.2 Uchování a zpracování dat

Data je možné uchovávat v různých datových strukturách a kolekcích, které umožnují sdružovat prakticky jakákoliv data a objekty a je možné je libovolně kombinovat, čehož nástroje často využívají. Například vyhledání všech výskytů určitého vzorce v textu pomocí regulárních výrazů umožňuje vytvořit seznam objektů obsahujících informace o umístění nálezu v textu, v případě rozdělení vzorce na skupiny i obsah jednotlivých skupin atd. Struktury je také možné mezi sebou snadno konvertovat s využitím integrovaných funkcí Pythonu. [7, 8, 10]

4.2 Realizace nástrojů pro převod

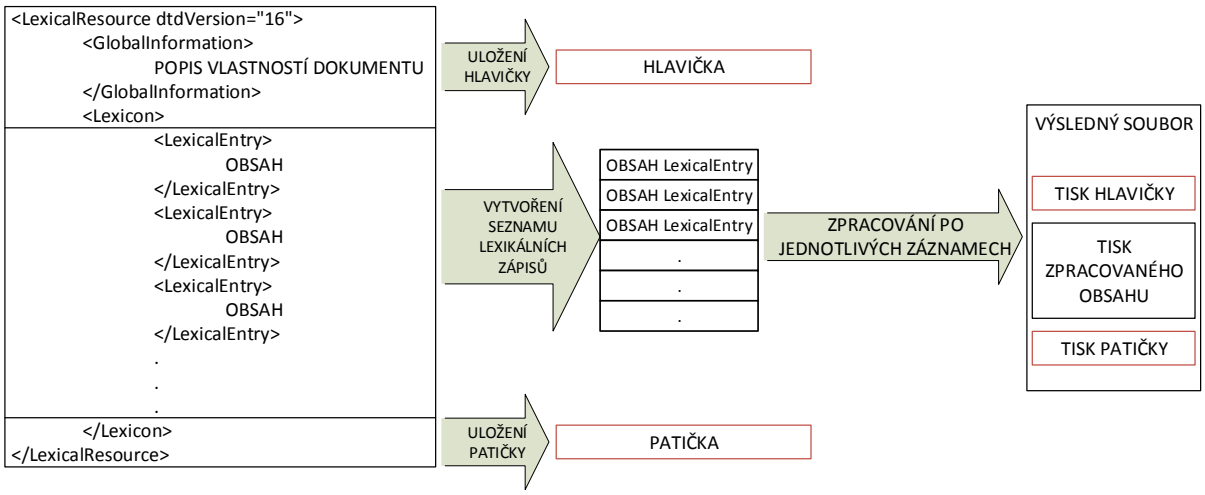
Se zvolenými prostředky je možné přistoupit k tvorbě nástrojů. Nejprve je nutné zpracovat strukturu slovníku a data uložit do vhodných datových struktur k dalšímu zpracování.

4.2.1 Zpracování LMF slovníků

Slovníky ve formátu LMF mají jednoznačně definovanou základní kostru, se kterou lze pracovat. Obecně se dá říct, že LMF soubor se skládá z „hlavičky“, která obsahuje definici základních vlastností slovníku, jako kódování, jazyk, typ dokumentu apod., dále obsahuje jednotlivé lexikální záznamy (značka `LexicalEntry`) neboli hesla slovníku, tedy obsah, se kterým se bude dále pracovat, a „patičku“ v podobě ukončovacích značek.

Pro zpracování jsou podstatné lexikální záznamy. Nástroje budou pracovat na úrovni jednotlivých lexikálních záznam, takže je výhodné text rozdělit právě na jednotlivé lexikální záznamy a ty poté uložit do vhodné struktury, která se poté bude postupně procházet po jednotlivých prvcích, tedy lexikálních záznamech, se kterými se potom bude dále pracovat.

Po zpracování obsahu některým z nástrojů se vytvoří výstupní soubor, kam se nejprve vytiskne hlavička, poté postupně obsah seznamu lexikálních záznamů a na závěr patička.



Obrázek 4.1: Znázornění zpracování vstupního souboru a vytvoření výsledného souboru

Tento systém využívají všechny nástroje, proto je implementován jako knihovna nabízející funkce pro otevření, načtení a rozbor zvoleného slovníku, což zahrnuje rozdělení na výše zmíněné tři základní části a vytvoření seznamu lexikálních záznamů.

Struktura lexikálních záznamů se již liší, a to v závislosti na typu slovníku, množství a typu dat zdrojového slovníku a v neposlední řadě na systému, kterým byl uskutečněn převod do LMF formátu. S rozdílnou strukturou je tedy třeba při následném zpracování počítat u každého nástroje také v závislosti na tom, jakým způsobem se bude s daty pracovat.

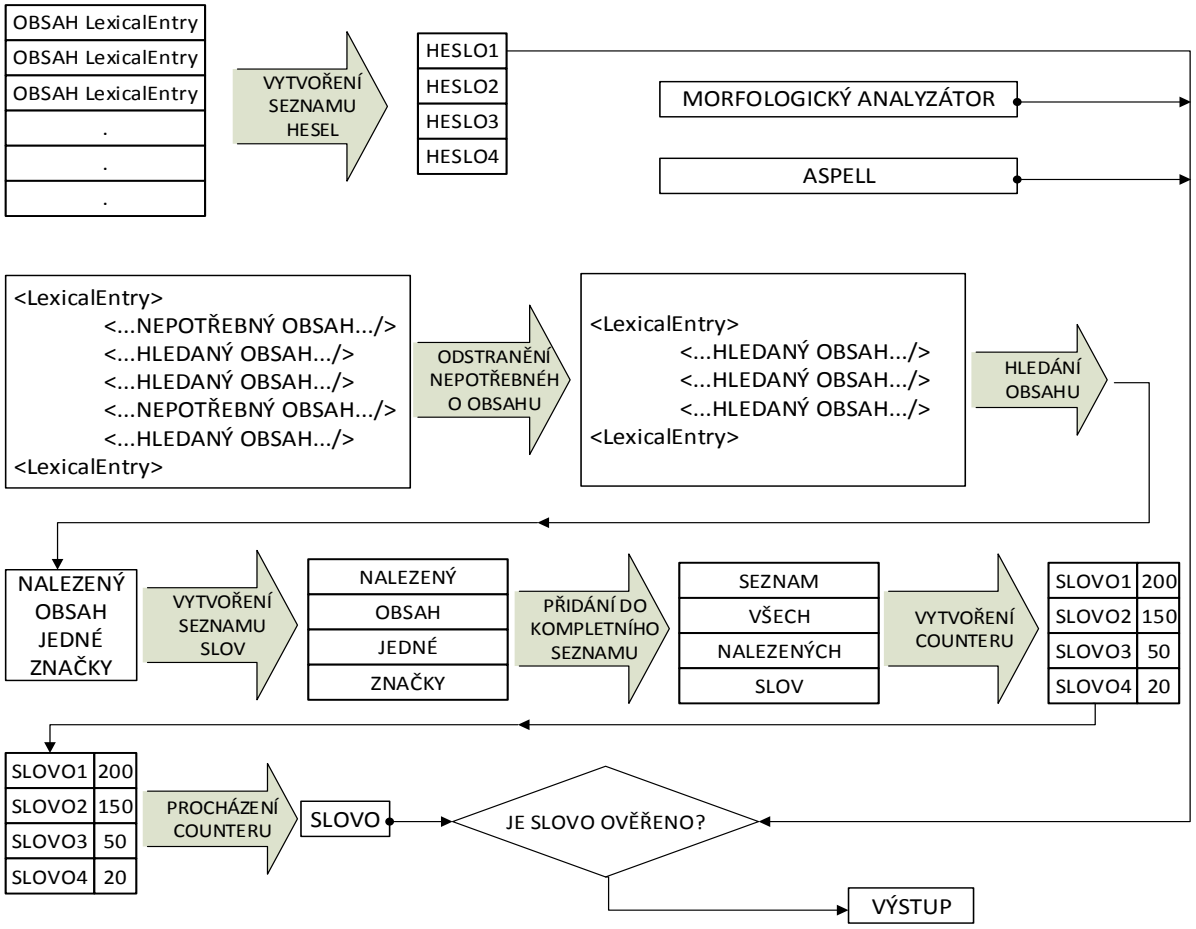
4.2.2 Systém pro výpis potenciálně chybného obsahu značek

Tento nástroj má sloužit k vytvoření statistických podkladů pro odhalení potenciálně chybného obsahu slovníku. Konkrétně je zaměřen na zjištění četnosti výrazů v překladových, příkladových a dalších částech slovníků a ověření, zda jsou dané výrazy obsaženy jako hesla v daném slovníku, v případě překladových slovníků i ve slovníku obráceném.

Postup zpracování dat je následující:

- Nejprve se naleznou všechna hesla daného, případně i obráceného, slovníku a vytvoří se jejich seznamy.
- Poté se začne prohledávat obsah jednotlivých lexikálních záznamů. Pro zpracování je požadována jen část obsahu, a protože samotný obsah není výstupem tohoto nástroje, je možné značky s nepotřebným obsahem odstranit. Ve zbývajících značkách se poté vyhledá jejich obsah, následně se rozdělí na jednotlivá slova a vytvoří se seznam těchto slov.
- V seznamu je poté nutné spočítat výskyty jednotlivých slov. Toho je docíleno využitím datové kolekce Pythonu zvané *Counter*, do které je seznam možné přímo převést
- Na závěr se pro každé slovo ověřuje, jestli je obsaženo v seznamech hesel daného, případně obráceného slovníku. Nalezená slova je také možné kontrolovat pomocí *morfologického analyzátoru* a nástroje *aspell*. [10, 11, 12]

Výstupem je textový soubor obsahující seznam slov s počtem výskytů a údaji o úspěšnosti jednotlivých ověření.



Obrázek 4.2: Princip systému pro výpis potenciálné chybného obsahu

Celkový přínos nástroje je nejen v tom, že pomáhá odhalit obsah chybný, ale i takový, který je potenciálním cílem dalších nástrojů tohoto systému, např. opakované výskyty zkratek. Konkrétním přínosem pro zpracování slovníků v rámci této práce bylo např. odhalení 7379 výskytů řady znaků „---“ v jednom ze slovníků, které způsobovaly komplikace při nahrazování zkratek, nebo právě nalezení opakujících se zkratek, případně opakující se výskyty řetězců typu „-xy-“ značící přítomnost zkrácených zápisů výslovnosti (2651 výskytů „-ty-“, 1613 „-ny-“ ve slovníku spisovné češtiny).

4.2.3 Systém pro hromadnou změnu obsahu značek

Cílem je nalezení a editace obsahu značek. Uživatel zadá hledaný výraz, a výraz, kterým bude hledaný nahrazen. Poté se vyhledají všechny výskyty hledaného výrazu v obsahu slovníku a nahradí se požadovaným výrazem.

K realizaci nástroje pro řešení tohoto problému je možné přistupovat dvěma základními způsoby – buďto bude interaktivní, tzn., že bude uživatele za běhu informovat o nalezených výskytech hledaného výrazu spolu s kontextem jednotlivých výskytů a tím pádem umožní uživateli zvolit, v jakých případech se má nahrazení provést, nebo bude uživatelský vstup přijímat

pouze jako parametry při spuštění a nahrazení potom provede automaticky ve všech kontextech v celém slovníku.

První přístup umožňuje vetší kontrolu nad měněným obsahem a může zabránit nechtěným změnám, zatímco druhý bude pravděpodobně možné lépe optimalizovat a bude tedy efektivnější a také, vzhledem k tomu, že za běhu nebude vyžadovat uživatelský vstup, jej bude možné spouštět automatizovaně.

Vytvořeny byly dva nástroje, každý využívající jeden z těchto způsobů. Testováním byly potvrzeny předpoklady uvedené výše. Interaktivní verze je vhodná spíše pro menší slovníky z důvodu náročnosti zpracování velkého objemu textu regulárními výrazy. Nahrazení trvá několikanásobně déle než v případě automatické verze. Dalším případem, kdy je vhodnější využívat automatickou verzi je, když je třeba změnit obsah vyskytující se v mnoha různých kontextech. V takovém případě interaktivní verze vypíše všechny kontexty a umožňuje je měnit pouze jednotlivě. Je tedy vhodnější tuto verzi používat na konkrétní menší změny ve vymezeném kontextu.

4.2.4 Systém pro rozdělení překladů, příkladů, vysvětlivek apod.

Tento nástroj rozděluje různé překlady, příklady atd. hromadně zapsané jako obsah jedné značky na jednotlivé výrazy zapsané v kopiích této značky. Obsah k rozdělení se určuje podle toho, jestli daná značka obsahuje určené oddělovače. Pokud je v obsahu značky nalezen oddělovač, obsah se rozdělí, jednotlivé části se vloží do kopií značky a tyto kopie se potom zapíšou za nalezenou značku. Umožňuje uživateli definovat množinu oddělovačů a názvy značek, ve kterých se bude obsah hledat.

Určení kontextu, se kterým se má pracovat, je poměrně komplikovaný problém, jehož řešení závisí na množství a uspořádání zanořených značek. Následuje vysvětlení na jednoduchém příkladu.

Pokud se obsah vyskytuje uvnitř rodičovské značky osamoceně:

```
<WordForm>
      <feat att="writtenForm" val="výraz1; výraz2; výraz3" />
</WordForm>
```

Pak není problém vyhledat okolní značky a poté jej rozkopírovat i s nimi:

```
<WordForm>
      <feat att="writtenForm" val="výraz1" />
</WordForm>
<WordForm>
      <feat att="writtenForm" val="výraz2" />
</WordForm>
<WordForm>
      <feat att="writtenForm" val="výraz3" />
</WordForm>
```

Komplikace nastávají, když je uvnitř rodičovské značky další obsah:

```
<WordForm>
	<feat att="script" val="Latn" />
	<feat att="writtenForm" val="výraz1; výraz2; výraz3" />
	<feat att="language" val="cz" />
</WordForm>
```

Potom by vznikl následující obsah:

```
<WordForm>
	<feat att="script" val="Latn" />
	<feat att="writtenForm" val="výraz1" />
	<feat att="language" val="cz" />
	<feat att="script" val="Latn" />
	<feat att="writtenForm" val="výraz2" />
	<feat att="language" val="cz" />
	<feat att="script" val="Latn" />
	<feat att="writtenForm" val="výraz3" />
	<feat att="language" val="cz" />
</WordForm>
```

Výsledný nástroj proto pracuje pouze s aktuální značkou, ve které jsou nalezeny oddělovače, a vznikne tedy:

```
<WordForm>
	<feat att="writtenForm" val="výraz1" />
	<feat att="writtenForm" val="výraz2" />
	<feat att="writtenForm" val="výraz3" />
</WordForm>
```

Tento způsob je nejlepším kompromisem mezi ideálním rozdělením obsahu a vznikem potenciálních chyb.

Úspěšnost nástroje závisí na z velké části na samotném obsahu. Pokud je obsah validní po syntaktické stránce, dojde vždy ke správnému určení mezí značky a jejímu zkopírování. Co se týká děleného obsahu, zde velmi záleží na použitých oddělovačích. Pokud je obsah oddělen specifickými znaky, které se nevyskytují uvnitř textu (např. několik jednoslovných významů oddělených středníkem), obsah se rozdělí bez chyb. Problémem je užívání čárek jako oddělovačů ve slovnících, kde se vyskytují rozsáhlejší popisy nebo vysvětlivky uvedené v souvětích. Nástroj potom rozdělí i tato souvětí, pokud se vyskytují ve stejně pojmenovaných značkách, jako obsah určený k rozdělení.

4.2.5 Systém pro nahrazení zkratek

Tento nástroj slouží k nahrazení zkratek a zástupek za heslové slovo za jejich plné znění.

Zpracování probíhá následovně:

- V aktuálním lexikálním záznamu se najde lemma.
- Zjistí se první písmeno, kvůli vyhledávání zkratek a poslední písmeno pro určení způsobu připojení přípon nebo koncovek ke slovu.
- V lexikálním záznamu se postupně hledají a nahrazují definované typy zkratek.
- K nalezené zkratce se hledá odpovídající tvar (nejbližší předchozí nezkrácený tvar), pokud není nalezen, použije se lemma.

Podle typu zkratky probíhá nahrazení rozdílnými způsoby:

- Zkratky ve tvaru „x.“ kde „x“ je první znak heslového slova – zkratka i s tečkou se pouze nahradí nalezeným plným tvarem slova.
- Předpony (např. heslové slovo je „kopat“, v textu je obsažena předpona „za-“ – význam je „zakopat“) – nalezne se požadovaný tvar slova a to se sloučí s příponou.
- Přípony (např. heslové slovo je „aeroplán“, v textu je obsažena přípona „-plánový“ – význam je „aeroplánový“) – připojení koncovky nebo přípony ke slovu je komplikovanější záležitostí. Nejprve proběhne pokus o mapování, kdy se hledá část přípony obsažená v daném slově. Z přípony se postupně odřezávají koncové znaky, dokud není nalezena shoda. Pokud není nalezena shoda ani v případě samotného prvního znaku, přípona se připojí za slovo. Připojení je pak nutné provést v závislosti na tom, jestli slovo končí souhláskou nebo samohláskou a stejně tak jestli přípona začíná souhláskou nebo samohláskou.
- Zkratky ve tvaru „x-y“ kde „x“ je první písmeno daného heslového slova a „y“ změněná koncovka – hledají se pouze zkratky, kde „x“ je první znak lemmatu. Ke zkratce se najde odpovídající tvar a ověří se, jestli jeho první znak odpovídá „x“ a pokud ano, mapuje se na konec slova „y“ stejným způsobem jako přípona v předchozím případě.
- Vysvětlivky, slovní druhy, určení oboru atd. – úplné tvary těchto zkratek se nedají odvodit z obsahu, využije se tedy tabulka převodů, kterou je nutné manuálně vytvořit.

4.2.5.1 Vyhodnocení výsledků

Při vyhodnocování úspěšnosti nahrazování zkratek je nutné vzít v úvahu několik faktorů ovlivňujících výsledky a také možnosti ověření platnosti výstupu. Automatizovaně se dá ověřit, jestli výsledné slovo je platné, např. pomocí *morfologického analyzátoru* nebo nástroje *aspell*. Určení správnosti použití výsledného tvaru v daném kontextu by vyžadovalo pokročilou analýzu textu, která je nad rámec tohoto nástroje. [11, 12]

Možnosti nástroje značně ovlivňuje způsob zápisu příkladů, vysvětlivek a dalších případů obsahujících zkratky. V překladových slovnících se často vyskytuje heslo, případně nějaký jeho tvar, následované příkladem, ve kterém je použita zkratka. Nahrazení v takovém případě proběhne bez problému i s odpovídajícím tvarem. Problémem jsou příklady a vysvětlivky uvedené na konci lexikálního záznamu, ve kterých jsou hromadně zapsány různé příklady využívající rozdílné tvary slov zapsaných zkratkou. Nalezení správného tvaru je pak současným systémem nemožné.

Následující statistiky popisují úspěšnost ověření výsledných slov nástrojem *aspell*.

Tabulka 4.1: Úspěšnost ověření nahrazených zkratek nástrojem aspell pro všechny nalezené výskyty (i opakované)

Slovník	Počet nalezených zkratek	Počet ověřených	Selhalo	Úspěšnost (%)
anglicko-český	1989	1697	292	85,32
americké angličtiny	2153	1671	482	77,61
česko-anglický	8579	7221	1358	84,17
spisovné češtiny	13089	9947	3142	76,00
spisovné češtiny	23575	16093	7482	68,26
spisovné češtiny	304809	234701	70108	77,00
			Průměrná úspěšnost (%):	78,06

Tabulka 4.2: Úspěšnost ověření nahrazených zkratek nástrojem aspell pro unikátní nálezy

Slovník	Počet unikátních slov	Počet ověřených	Selhalo	Úspěšnost (%)
anglicko-český	852	680	172	79,81
americké angličtiny	852	453	399	53,17
česko-anglický	2621	2224	397	84,85
spisovné češtiny	9085	6350	2735	69,90
spisovné češtiny	23091	15725	7366	68,10
spisovné češtiny	142021	89138	52883	62,76
			Průměrná úspěšnost (%):	69,77

Z výše uvedených tabulek vyplývá, že v průměru je pouze 65 % výsledných ověřeno *aspellem*, což může vypadat jako nízká úspěšnost, ale je třeba počítat s tím, že např. slovník pro češtinu využívaný tímto nástrojem neobsahuje všechna slova. Jako příklad uvedu výstup ze slovníku spisovného jazyka českého, kde v různých vzorcích 50 slov, která *aspell* nerozpoznal, je obsaženo přibližně 10 - 20 platných. Číselné vyjádření úspěšnosti je tedy značně komplikované.

Chyby vznikají zejména při zápisu přídavného jména vyjádřeného zkratkou v hesle uvádějícím podstatné jméno. Např. pod heslem „abatyše“ je uveden příklad „a-á hodnost“ bez odpovídajícího tvaru „abatyšský“. Nástroj vytvoří tvar „abatyšá hodnost“, ale správný tvar je „abatyšská hodnost“, což nelze určit automaticky a vyžadovalo by to analýzu kontextu a pravidel při odvozování slovních druhů. Dalším problémem jsou přejatá slova s různými pravidly skloňování. Např. heslo „absolutismus“ a tvar „a-u“. Správný tvar je „absolutismu“, ale nástroj vytvoří „absolutismusu“.

Nástroj tedy funguje správně v případě, že je ve zkratce uvedena celá část slova, která se má změnit pro odpovídající tvar a to pro česká slova skloňovaná podle obvyklých pravidel, případně pro jazyky, ve kterých se slova neskloňují. Pro odstranění nedostatků by systém bylo nutné rozšířit o rozpoznávání kontextu a slovních druhů a vytvořit speciální případy pro pravidla skloňování přejatých a cizích slov, případně využít možností morfologickéo analyzátoru pro generování různých tvarů slov.

4.2.6 Systém pro převod zápisu výslovnosti do jednotné podoby

Účelem tohoto nástroje je převod zápisu výslovnosti ze zkrácené do úplné podoby. Je zaměřen zejména na překladové slovníky z češtiny do cizích jazyků a české výkladové slovníky. S mezinárodní fonetickou abecedou pracovat neumí. K převodu využívá dva základní postupy:

- Mapování znaků úseku výslovnosti na psanou formu – vyhledání krajních znaků úseku výslovnosti v psané formě a vložení do psané formy na pozici mezi nalezenými shodnými znaky.
- Využití tabulky přepisů jednotlivých znaků nebo skupin znaků – podle definované tabulky se zápis výslovnosti převede tak, aby odpovídal psané formě, ve které se pak tento úsek vyhledá a na jeho místo je vložen původní úsek výslovnosti.

Obě tyto metody je možné kombinovat. Pokud mapování selže, úsek výslovnosti se převede pomocí tabulky přepisů a mapování se zopakuje.

Mapování znázorněné na příkladu „abdikace“ uvedeném výše:

- psaná forma: „abdikace“
- úsek výslovnosti: „-dy-“
- mapování:
  - krajní znaky výslovnosti jsou „d“ a „y“, „d“ je v psané formě nalezeno, hledá se „y“, které nalezeno není
  - podle tabulky přepisů se zjistí, že „y“ je možný zápis výslovnosti pro „i“
  - „y“ se nahradí za „i“, nyní se pracuje s úsekem výslovnosti „di“
  - krajní znaky výslovnosti jsou „d“ a „i“, „d“ je v psané formě nalezeno, hledá se „i“, které je také nalezeno
  - určí se pozice v psané formě mezi nalezenými znaky a na toto místo je poté vložen původní úsek výslovnosti „dy“, vznikne úplná forma výslovnosti „abdykace“

Při mapování je nutné řešit několik problémů:

- Prvním z nich je možnost opakovaného výskytu prvního znaku úseku výslovnosti v psané formě slova. Např. slovo „zautomatizování“ a zápis výslovnosti „-ty-“. První shoda je nalezena na čtvrté pozici a s použitím tabulky přepisů je shoda koncového „i“ nalezena na deváté pozici. Pak by vzniklo „zautyzování“.
- Dalším problémem je naopak nalezení shody v příliš krátkém úseku. Např. slovo „benediktin“ a výslovnost „-dyktý-“. S tabulkou přepisů se dojde k tvaru výslovnosti „-dykti-“ a shoda koncového znaku „i“ je nalezena ihned za nalezeným první znakem „d“, výsledek by tedy byl „benedyktýktin“.

Tyto problémy je možné řešit omezením vzdálenosti mezi nalezenými shodami v závislosti na délce úseku výslovnosti. Délka ovšem nelze určit přesně, protože některé dvojice znaků se vyslovují jako jeden (např. „establishment“, kde se „sh“ vyslovuje jako „š“) nebo naopak jeden znak se vyslovuje jako dva (např. „olympiáda“, kde se „i“ vyslovuje jako „ij“). Vhodným kompromisem je tedy omezit vzdálenost mezi nalezenými shodami na „délka úseku - 1“ až „délka

úseku + 1“. Tento způsob nepokryje všechny možnosti, ale omezí chyby typu výše zmíněné na minimum.

- Problémem může být i možnost opakovaného výskytu stejného úseku výslovnosti. Např. slovo „investigativní“ a výslovnost „-ty-ty-“. V prvním průchodu je s využitím tabulky přepisů nahrazen první výskyt a vznikne „investygativní“, pro další výskyt je poté i bez tabulky nalezena shoda na místě prvního výskytu a k druhému se systém již nedostane. To je vyřešeno porovnáním, jestli část slova na místě shody není shodná s úsekem výslovnosti a pokud ano, opětovným hledáním prvního znaku dále ve slově.

Při přepisu výslovnosti na začátku nebo konci slova může hledání shody probíhat postupným „ořezáváním“ počátečních, resp. koncových znaků, dokud není nalezena shoda úseku výslovnosti s částí psané formy. Problémy s nalezením shody na špatné pozici jsou opět řešeny vymezením vzdálenosti nalezené shody od začátku nebo konce slova v závislosti na délce úseku výslovnosti.

Například:
- psaná forma: „potpourri“
- úsek výslovnosti: „-puri“
- mapování:
  - proběhne pokus o vyhledání (od konce) „puri“ v „potpourri“, není nalezeno
  - ořízne se koncový znak, vznikne úsek „pur“
  - hledání a ořezávání se opakuje, dokud není nalezena shoda
  - po nalezení shody „p“ se čtvrtým znakem v „potpourri“ se ořízne část slova od tohoto znaku (včetně) a připojí se výslovnost – vznikne „potpuri“.

Optimalizací je vytvoření speciálních případů pro úsek výslovnosti délky dvou znaků. V takovém případě je možno říci, že délka úseku je shodná s délkou měněné části slova. Potom je v závislosti na pozici měněné části ve slově možné určit, jestli se taková možnost vyskytuje ve slově pouze jednou a pokud ano, přímo provést nahrazení. U změn na začátku slova se porovná shoda prvního znaku úseku výslovnosti s prvním znakem výslovnosti, případně totéž pro druhé znaky, a pokud se jedna z dvojic shoduje, úsek se nahradí. Stejným způsobem se porovnávají koncové znaky při nahrazení na konci slova. V případě změny uvnitř slova se zjistí, jestli se první znak výslovnosti vyskytuje ve slově pouze jednou a pokud ano, provede se nahrazení na místě shody s následujícím znakem.

4.2.6.1 Vyhodnocení výsledků

Systém byl v průběhu vývoje testován na několika slovnících obsahujících výslovnost zapsanou ve zkrácené podobě a na základě výsledků upravován. Byly vyzkoušeny různé postupy a zjišťován jejich přínos. Následuje vyhodnocení výsledků s použitím samotné tabulky přepisů bez mapování, naopak mapování bez využití tabulky přepisů a kombinace obou metod na pěti vybraných slovnících obsahujících dostatečné množství vzorků.

Tabulka 4.3: Úspěšnost převodu výslovnosti s použitím tabulky přepisů bez mapování

	Tabulka přepisů bez mapování			
Slovník	Počet nalezených	Počet nahrazených	Selhalo	Úspěšnost (%)
spisovné češtiny	450	360	90	80,00
česko-anglický	4371	4097	274	93,73
výkladový anglický	5637	4683	954	83,08
spisovné češtiny	11065	9117	1948	82,39
výkladový český	11515	9477	2038	82,30
			Průměrná úspěšnost (%):	84,30

Výsledky převodu pouze s využitím tabulky přepisů ukazují poměrně vysokou úspěšnost zejména u česko-anglického slovníku, pro který byla tabulka nejvíce přizpůsobována. Při dodání dalších pravidel specifických pro jednotlivé slovníky by se úspěšnost zvýšila i u ostatních, ale vyžadovalo by to podrobnou analýzu jednotlivých slovníků.

Tabulka 4.4: Úspěšnost převodu výslovnosti mapováním bez použití tabulky přepisů

	Mapování bez tabulky přepisů			
Slovník	Počet nalezených	Počet nahrazených	Selhalo	Úspěšnost (%)
spisovné češtiny	450	318	132	70,67
česko-anglický	4371	3325	1046	76,07
výkladový anglický	5637	5375	262	95,35
spisovné češtiny	11065	7865	3200	71,08
výkladový český	11515	8183	3332	71,06
			Průměrná úspěšnost (%):	76,85

Samotné mapování vykazuje o něco menší úspěšnost, ale stále velmi dobrou, vezmeme-li v úvahu, že pracuje automaticky, pouze s aktuálními zdroji, tedy znaky nalezených slov a úseků výslovnosti a bez využití manuálně definovaných pravidel nebo externích dat.

Tabulka 4.5: Úspěšnost převodu výslovnosti mapováním s použitím tabulky přepisů

	Kombinace metod			
Slovník	Počet nalezených	Počet nahrazených	Selhalo	Úspěšnost (%)
spisovné češtiny	450	406	44	90,22
česko-anglický	4371	4341	30	99,31
výkladový anglický	5637	5472	165	97,07
spisovné češtiny	11065	10131	934	91,56
výkladový český	11515	10522	993	91,38
			Průměrná úspěšnost (%):	93,91

Kombinace obou metod již přináší velmi dobré výsledky s průměrnou úspěšností přes 90 %. Využití tabulky přepisů se opět nejvýrazněji projevilo u česko-anglického slovníku, pro který je

nejlépe uzpůsobena. Úspěšnost nalezení shody je zde přes 99 %. Analýzou neúspěšných převodů a doplněním dalších pravidel do tabulky by se úspěšnost dala ještě zvýšit.

Tyto statistiky vyhodnocují úspěšnost nalezení shody úseku výslovnosti ve slově a následného převodu a jsou výstupem samotného systému. Jsou zde zahrnuty i případy, kdy byl ve slovníku špatně uveden zápis výslovnosti (např. měl patřit k jinému slovu v rámci lexikálního záznamu). Výpis neúspěšných převodů se tedy částečně dá využít i k odhalení chybného obsahu slovníků a odstranění těchto chyb by dále zlepšilo úspěšnost.

Ověření správnosti výsledných převedených tvarů nelze realizovat automatizovaně, protože tyto tvary nejsou platnými slovy zahrnutými v databázích využívaných různými systémy pro kontrolu slov. Manuální procházení a kontrola slov je při počtech nálezů v řádech tisíců až desetitisíců také poměrně náročný úkol. K odhalení alespoň části chyb může přispět vypsání výsledků, kde se délka výslovnosti výrazně liší od délky původního slova a mohlo tedy dojít k chybě v mapování. Tímto způsobem byly v rámci vyhodnocovaných slovníků odhaleny jednotky chyb, přičemž část byla opět způsobena chybným obsahem slovníku. Chyby způsobené převodem vycházejí ze systému mapování a vyžadovaly by vytvoření speciálních případů, případně by jejich oprava vytvořila jiné chyby.

5 Závěr

Cílem práce bylo vytvořit systém pro převod slovníků ve formátu LMF do podoby lépe využitelné pro strojový překlad. Pro tento účel byla vytvořena sada nástrojů určených k řešení konkrétních problémů. Nejprve bylo nutné seznámit se s problematikou uložení slovníků v elektronické podobě a jejich převodu do jednotného formátu LMF, následně analyzovat problémy a chyby obsažené v takto převedených slovnících a vymezit konkrétní problémy, které znesnadňují strojovou čitelnost slovníků a budou předmětem řešení. Tyto problémy jsou hledání chybného obsahu značek a samotný chybný obsah, značky obsahující více hodnot, které by měly být rozděleny, používání zkratek a zástupek za heslové slovo v některých částech slovníku a zkrácený zápis výslovnosti. Po analýze jednotlivých problémů bylo možné přistoupit k návrhu jejich řešení. Pro jednotlivé vytvořené nástroje byla popsána jejich funkcionalita, způsob řešení daného problému a byla analyzována jejich úspěšnost.

Z výsledků vyplývá, že nástroje svůj účel plní s dobrou úspěšností s přihlédnutím k tomu, že zpracovávané slovníky obsahují další chyby, jejichž řešení nebylo předmětem této práce. Vytvoření statistik pro odhalení potenciálně chybného obsahu proběhne bez problémů a z výsledků je patrné, když se v obsahu slovníku vyskytují opakované chyby. Úspěšnost rozdělení obsahu značek závisí hlavně na samotném obsahu a na typu oddělovačů obsahu. Zásadní problém vzniká při použití znaku čárky jako oddělovače. V takovém případě dochází k nechtěnému rozdělování souvětí. Prostor pro případná vylepšení či rozšíření je zejména u systémů pro převod zkratek a výslovnosti, které by mohly využívat učení ke zlepšování úspěšnosti. Převod zkratek by dále bylo dobré rozšířit o možnost rozpoznání kontextu a tím pádem vložení správného tvaru výsledného slova. Systém pro rozdělování obsahu značek by bylo možné rozšířit o možnost určení rodičovského prvku zpracovávané značky.

Práce pro mě byla velkým přínosem v několika oblastech. Jednou z nich je organizace práce a časového rozvrhu a také nutnost podrobné analýzy problému a vytvoření vhodného návrhu před započetím práce. Dále jsem získal zkušenosti se zpracováním strukturovaného textu a s programováním v jazyce Python. Také jsem získal nové znalosti o slovnících a jejich elektronické podobě. Zpracování přirozeného jazyka je celkově zajímavý a obsáhlý obor a zpracování slovníků, jako prostředku k vysvětlení významu slov, je jeho významnou součástí. Věřím, že tato práce přispěje k řešení problémů s tímto oborem spojených.

Literatura

[1] ŠIMEČKOVÁ, Alena. Úvod do studia jazykovědné germanistiky. Vyd. 1. V Praze: Karolinum, 2004, 176 s. ISBN 80-246-0595-3.

[2] Ottův slovník naučný: Ilustrovaná encyklopedie obecných vědomostí. Dvacátýtřetí díl. Schlossar - Starowolski. S 28 přílohami a 231 vyobrazeními v textu. fotoreprint pův. vyd. Praha: Sdružení pro Ottův slovník naučný, 2000, 1064 s. ISBN 80-720-3324-7.

[3] KOSEK, Jiří. XML pro každého: podrobný průvodce. 1. vyd. Praha: Grada, 2000, 163 s. Průvodce (Grada). ISBN 80-716-9860-1.

[4] FRANCOPOULO, Gil. LMF: Lexical Markup Framework. London: ISTE, 2013, xiv, 266 p. ISBN 978-1-84821-430-9.

[5] SÝKORA, Michal. Formát XML pro značkování slovníků: XML Dictionary Tagging. Brno: Vysoké učení technické, Fakulta informačních technologií, 2009. 1 elektronický optický disk [CD-ROM / DVD].

[6] HOLZNER, Steven. XSLT: příručka internetového vývojáře. Vyd. 1. Praha: Computer Press, 2002, 515 s. ISBN 80-722-6600-4.

[7] SUMMERFIELD, Mark. Python 3: výukový kurz. Vyd. 1. Překlad Lukáš Krejčí. Brno: Computer Press, 2010, 584 s. ISBN 978-80-251-2737-7.

[8] LUTZ, Mark. Learning Python. 5th ed. Beijing: O'Reilly Media, 2013, 1540 s. ISBN 978-1449355739.

[9] GOYVAERTS, Jan a Steven LEVITHAN. Regulární výrazy: kuchařka programátora. Vyd. 1. Brno: Computer Press, 2010, 381 s. ISBN 978-80-251-1935-8.

[10] Python 3.4.0 documentation [online]. 2014, 15. května [cit. 2014-05-15]. Dostupné z:https://docs.python.org/3/

[11] Morfologický slovník a morfologický analyzátor pro češtinu. *KNOT Wiki* [online]. 2013, 6. prosince [cit. 2014-05-17]. Dostupné z:http://knot.fit.vutbr.cz/wiki/index.php/Morfologick%C3%BD_slovn%C3%ADk_a_morfologick%C3%BD_analyz%C3%A1tor_pro_%C4%8De%C5%A1tinu

[12] ATKINSON, Kevin. *GNU Aspell* [online]. 2004 [cit. 2014-05-17]. Dostupné z: http://aspell.net/

Seznam příloh

Příloha 1. CD se soubory zdrojových kódů v jazyce Python, uživatelskou příručkou k jednotlivým nástrojům a elektronickou verzí této práce