UNIVERZITA KOMENSKÉHO
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
ÚSTAV INFORMATIKY

Ján Michalička

Štatistický strojový preklad veľmi blízkych jazykov (slovenčina – čeština)

Diplomová práca

Školiteľ: Mgr. Ján Habdák

Bratislava 2005

Obsah:

1 Úvod

V minulých rokoch sa kládol veľký dôraz na prekladové štúdie a korpusovo založené prekladové systémy používajúce Štatistický Strojový Preklad (SMT). Základná idea tohoto princípu spočíva vo využití dát jedného jazyka ako aj popárovaných preložených dát (bitextov) na automatické natrénovanie prekladového modelu a jazykového modelu, ktorý môže byť využitý na vývoj dekodéra, ktorý vykonáva samotný preklad.

Strojový preklad medzi rôznorodými jazykmi naráža na mnohé problémy, ktoré súvisia najmä so štruktúrou jazyka a jeho komplexnosťou. Úspešný automatický strojový preklad vyžaduje aplikáciu techník z rôznych oblastí výpočtovej lingvistiky (morfológiu, syntax, sémantiku, analýzu reči, atď.) ako nutnú, ale nie postačujúcu podmienku.

Hlavnou myšlienkou je, že je jednoduchšie vytvoriť MT systém pre dvojicu príbuzných jazykov, nakoľko toto umožní redukciu komplexnosti a tým aj zvyšuje dosiahnuteľnú úspešnosť.

Z existujúcich riešení automatického strojového prekladu sú na trhu dostupné najmä technológie prekladu medzi príbuznými jazykmi ako angličtina, nemčina, francúzština, španielčina... ktoré majú isté spoločné charakteristiky a to napr. pevný slovosled a absencia ohýbania.

V tejto práci si ukážeme ako je to so strojovým prekladom veľmi príbuzných jazykov (obsahujúcich vysoký stupeň ohybnosti – čeština a slovenčina) a porovnáme úspešnosť použitia viacerých existujúcich prístupov ako aj nového slovníkovo-štatistického prístupu vytvoreného za účelom tejto práce.

Taktiež sa pokúsime vyriešiť problém zaobchádzania so slovami, ktoré nevieme preložiť podľa slovníka prekladových párov a to metódou nahradenia vhodným slovom podľa jazykového modelu.

2 Pozadie strojového prekladu

V tejto kapitole si predstavíme základné znalosti z oblasti prekladu a spracovania korpusu, ktoré sa využívajú v rôznych prekladových systémoch. Taktiež si predstavíme spôsoby vyhodnotenia úspešnosti pre dané prístupy.

2.1 Paralelný korpus

V roku 1961 začala po prvýkrát práca na textovom korpuse, ktorý neskôr dostal názov „Brown Corpus" [Francis a Kučera 1964]. Toto bolo míľnikom korpusovej lingvistiky. Použitie termínu „korpusová lingvistika" sa vzťahuje na jazykové materiály, ktoré existujú v elektronickej forme a na rôzne metódy a softvérové nástroje, ktoré sa používajú na analýzu a spracovanie dát. Neustále zvyšujúca sa výpočtové sila a kapacity diskového priestoru neznamenajú len zvyšovanie počtu dostupných textov v korpuse, ale taktiež textový korpus neustále narastá na veľkosti, je viacej členitý a je možné ho efektívnejšie spracovávať.

Korpus:

- znamená kolekciu textov udržiavaných v strojovo čitateľnom formáte s možnosťou automatickej alebo poloautomatickej analýzy rôznymi prístupmi
- neobmedzuje sa len na „písomnosti", ale rozširuje sa aj o hovorený, tak ako aj o písaný text
- môže obsahovať obrovské množstvo textov z rôznorodých zdrojov

Bohužiaľ, korpus vždy bude iba vzorkou jazyka a nikdy nebude reprezentovať jazyk v celku. Sila prirodzeného jazyka nemôže byť prekonaná konečnou množinou vzoriek.

V závislosti od charakteristík textu, korpus môže byť vo všeobecnosti rozdelený do viacerých kategórií. V závislosti od toho, či korpus obsahuje text iba jedného, alebo viac ako jedného jazyka rozlišujeme jednojazyčný a viacjazyčný korpus. Viacjazyčný korpus potom môžeme rozdeliť na paralelný alebo nie-paralelný korpus.

Paralelný korpus je predstavovaný vetami prirodzeného jazyka a ich prekladmi spolu so zarovnaním medzi korešpondujúcimi segmentmi rôznych

jazykov. Paralelný korpus zvyčajne obsahuje zdrojový dokument a jeden, alebo viacero prekladov tohoto dokumentu. Bilingválny paralelný korpus je niekedy nazývaný aj „bitext" [Isabelle 1992] a korešpondujúce segmenty v korpuse sú nazývané bitextovými segmentmi [Ahrenberg 1999].

Existuje veľa aplikácií používajúcich paralelný korpus na prekladové štúdie pre úlohy spracovania viacjazyčného prirodzeného jazyka (natural language processing – NLP). Paralelné korpusy sa stávajú v posledných rokov viac a viac rozšírenými a používanými ako zdroj pre NLP . Štatistický strojový preklad je iba jedna z ciest použitia vyvinutá vďaka rozširujúcej sa veľkosti paralelného korpusu.

Väčšina paralelných korpusov obsahuje iba dva jazyky, zdrojový a cieľový jazyk. Veta zo zdrojového jazyka a k nej prislúchajúci preklad v cieľovom jazyku sa nazýva **vetný pár**. V tejto práci pokiaľ sa bude hovoriť o paralelnom korpuse, tak za zdrojový a cieľový jazyk bude považovaná slovenčina a čeština.

Bohužiaľ existenciu paralelného korpusu medzi týmito dvoma jazykmi sa nepodarilo zistiť, takže za účelom tejto práce bol paralelný korpus zostrojený z dostupných zdrojov, a to z tituliek k filmom dostupných na internete. Na párovanie viet boli použité časové značky titulky vzhľadom k filmu.

Taktiež väčšina Slovákov rozumie veľmi dobre češtine a majoritná väčšina tituliek preto existuje práve v českom jazyku, čoho dôsledkom je, že množstvo existujúcich a správnych párov v slovenskom jazyku nie je príliš vysoké.

Zostrojený paralelný korpus teda pokrýva veľmi malú časť jazyka a je doménovo zameraný najmä na dialógové texty hovorovej reči.

2.2 Zarovnanie viet

Podľa rôznorodosti potrieb môže byť zdrojový text rozdelený do rôznych segmentov, ktoré korešpondujú monotónne k segmentom v cieľovom texte. Ako hlavné segmentačné jednotky sa používajú odstavce a vety.

Zarovnanie je definované ako objekt indikujúci korešpondujúce slová a frázy v paralelnom texte [Brown 1993]. Jednoduchšie povedané, zarovnanie hovorí, ktoré jednotky v zdrojovom texte ukazujú na jednotky v cieľovom texte. Zarovnanie korešpondujúcich viet je nazývané vetné zarovnanie [Ahrenberg 1999]. Ukončovacie prvky sú bodka na konci vety, otáznik a výkričník. Vo vetnom zarovnaní niektoré skupiny viet v jednom jazyku korešpondujú obsahovo k inej

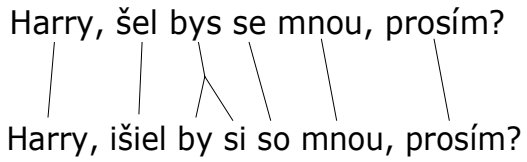
skupine viet v inom jazyku. Najčastejším prípadom býva, že súvetie v jednom jazyku je preložené do dvoch viet v inom jazyku, alebo opačne. Takéto zarovnanie nám potom pomáha prehľadávať paralelný korpus zdrojových a prislúchajúcich textov po častiach. Zvyčajne sa predpokladá, že informácie na úrovni viet sú prezentované v rovnakom poradí v zdrojovom texte, ako aj v cieľovom texte. S týmto predpokladom môže byť vetné zarovnanie prezentované ako zarovnanie bez krížových **liniek**.

Existujú dva hlavné prístupy k problému vetného zarovnania: zarovnanie založené na dĺžke a slovníkovo založené zarovnanie [Tiedimann 2003]. Prístup založený na dĺžke využíva počet písmen [Gale a Church 1991] alebo počet slov vo vete [Brown 1991]. Prístup slovníkovo založeného zarovnania využíva morfologické charakteristiky slov alebo iné lexikálne jednotky, prípadne kombináciu oboch.

2.3 Zarovnanie slov

Zarovnanie slov je proces identifikácie, ktoré slovo z danej vstupnej vety môže byť preložené na ktoré slová v danej cieľovej vete. Zarovnanie slov je základnou úlohou pri budovaní bilingválnych slovníkov.

Príklad jednoduchého zarovnania slov je graficky znázornený v Obrázok 2.1. Linky v obrázku sú nazývanie prepojenia [Brown 1993] zo slov v češtine do slov v slovenčine. Pre jeden pár viet existuje vždy viac ako jedna cesta, ako navzájom poprepájať slová. Jedna z ciest, ako priradiť prepojenia jednotlivým slovám, je aj zarovnanie. Niekedy niektoré slová v zdrojovej vete nemajú svoj pár v cieľovej vete, a taktiež pre niektoré slová v cieľovej vete neexistuje žiadna linka na slovo v zdrojovej vete. Teda niektoré slová pri preklade zanikajú, alebo opačne niektoré slová v cieľovej vete akoby vznikli z ničoho v zdrojovej vete.



Obrázok 2.1 Príklad jednoduchého zarovnania slov

Vytvorenie zarovnania medzi dvoma vetnými pármi môže byť veľmi komplikované. Taktiež aj pre človeka je rozhodnutie, ktoré slová z cieľovej vety korešpondujú ku ktorým slovám v zdrojovej vete, niekedy náročný problém. Hlavnou príčinou tejto situácie je, že význam „korešpondencie" medzi slovami je subjektívny.

Jeden z hlavných cieľov zarovnania slov je produkcia lexikálnych dát pre bilingválne slovníky, zatiaľ čo iným dôvodom môže byť poskytovanie dát pre strojový preklad alebo pre rôzne iné štúdie. Bolo pozorované, že v čistom slovo-na-slovo modeli [Melamed 1995] sa nenachádza veľa správnych lexikálnych jednotiek a to z dôvodu, že mnohé tieto spojenia patria do významovo väčších celkov – fráz. Systémy na zarovnávanie slov, ktoré sú založené na vedomostiach slovného zarovnania, využívajú systémy, ktoré zarovnávajú lingvistické jednotky na úrovni nižšej ako úroveň viet medzi danými dvoma jazykmi. Úlohou takýchto systémov je hľadať všetky korešpondencie na lexikálnej úrovni, ktoré existujú v danom paralelnom texte. Pre všetky typy systémov na zarovnávanie slov je dôležité, aby boli schopné nejakým spôsobom spracovávať viacslovné segmenty v obidvoch textoch, zdrojovom aj cieľovom, avšak pre veľmi blízke jazyky to nemusí byť až také markantné.

Existuje veľa prekladových relácií medzi slovami a frázami v paralelnom korpuse. Slovné poradie vo všeobecnosti nebýva identické pre väčšinu dvojíc jazykov a konce lexikálnych jednotiek tiež nie je jednoduché zistiť, tak ako je to aj pri zisťovaní koncov vetných jednotiek. Neexistuje konzistentná korelácia medzi počtami znakov v korešpondujúcich slovách prinajmenšom pre väčšinu jazykových párov. Avšak prirodzené jazyky musia vo všeobecnosti obsahovať nejakú formu kompozičnosti, aby bolo možné realizovať medzi nimi preklad. Takže je možné nájsť veľa prekladových relácií medzi týmito kompozičnými komponentmi, t.j. slová a frázy, v dokumentoch a ich prekladoch.

Typy relácií medzi slovami v paralelných textoch sa v závislosti na úlohe menia podľa stratégie použitého zarovnávania slov a fráz. Zvyčajne sa zarovnanie slov usiluje o kompletné zarovnanie všetkých lexikálnych jednotiek v korpuse. Stupeň korešpondencie môže byť vyjadrený v termínoch pravdepodobnosti zarovnania, ktorá je užitočná pre štatistický strojový preklad.

Existujú dva prístupy k zarovnaniu slov. Prístup asociovania používa mieru korešpondencie nejakého typu. Taktiež využíva heuristické prístupy [Och and Ney 2003] alebo prístup testovania hypotéz [Hiemstra 1994]. Prístup odhadovania používa pravdepodobnostné modely prekladu. Taktiež je často nazývaný štatistickým zarovnávaním [Och and Ney 2000]. Obidva prístupy využívajú nejaký

druh štatistiky. Tieto dva prístupy si predstavíme a rozoberieme v nasledujúcej časti.

2.3.1 Prístup asociovania

Prístup asociovania zarovnania slov pochádza najmä z skorších štúdií lexikálnej analýzy paralelných dát. V procese extrakcie slovníkových dát nie je nevyhnutné zarovnať všetky výskyty lexikálnych jednotiek na korešpondujúce jednotky pri preklade. Hlavné kroky, ktoré sa používajú pri zarovnaní bitextov použitím prístupu asociovania sú [Tiedemann 2003]:

1. Lexikálna segmentácia: Identifikujeme okrajové hranice lexikálnych jednotiek v oboch jazykoch

2. Zhoda: Možné relácie prekladu medzi lexikálnymi jednotkami identifikujeme vzhľadom na kritérium zhody. Toto zvyčajne vedie k množine ováhovaných liniek slovných typov, t.j. prekladový slovník s asociačnou úspešnosťou pripojený k jednotlivým položkám. Kontextové vlastnosti môžu byť tiež pripojené k tomuto asociačnému slovníku.

3. Zarovnanie a extrakcia: Najviac vhodné preklady prislúchajúce k asociačnému slovníku sú vyznačené v bitextoch. Toto sa zvyčajne realizuje pomocou „greedy" algoritmu použitím jednoduchých vyhľadávacích stratégií ako „prvý najlepší" výskyt v kombinácii s nejakými lingvistickými/heuristickými obmedzeniami. Bilingválne prekladové slovníky môžu byť zhotovené zo zarovnaných jednotiek (extrakcia), ktoré sú zvyčajne „čistejšie" ako tie, ktoré boli získané v predchádzajúcom asociačnom slovníku.

Miera výskytu a podobnosť reťazcov sú dvoma hlavnými technikami zarovnávania používanými v prístupe asociovania [Tiedemann 2003].

Podmienka zarovnávania jedno-na-jedno slovo je nedostatočná pre väčšinu jazykových párov. V zarovnávaní slov boli uskutočnené viaceré štúdie na použitie viac-slovných jednotiek, ktoré sa zameriavajú na slovné sekvencie a skupiny slov [Tiedemann 2003]. Pri použití viac-slovných jednotiek rozoznávame dve hlavné techniky: prioritná identifikácia zoskupení a dynamická konštrukcia viac-slovných jednotiek počas procesu zarovnávania.

Použitie štrukturálnych informácií v procese extrakcie bilingválnych termínov bolo skúmané vo viacerých štúdiách [Van der Eijk 1993]. Taktiež existujú štúdie používajúce štatistický alebo hybridný prístup pre identifikáciu frázových jednotiek

v bitextových segmentoch pred samotným vykonaním zarovnávania slov [Ahrenberg 1998].

2.3.2 Prístupy odhadovania

Prístupy odhadovania sú založené na zarovnávaní slov použitím pravdepodobnostných zarovnávacích modelov, ktoré sú získane s paralelného korpusu. [Tiedemann 2003]. Väčšina práce v tejto oblasti bola inšpirovaná prácou na štatistickom strojovom preklade (SMT), ktorou sa predstavil [Brown 1990]. Princípy štatistického strojového prekladu, aplikácia týchto modelov na zarovnávanie slov a extrakciu slovníkov sú popísané v nasledujúcom texte.

Princípy SMT sú založené na poznatkoch získaných z bilingválneho korpusu, k danej vete v zdrojovom jazyku hľadáme najpravdepodobnejší preklad v inom jazyku. Rozdiel medzi tradičným strojovým prekladom a štatistickým strojovým prekladom je, že tradičný strojový preklad je založený na pravidlách, zatiaľ čo štatistický je založený na zbere dát.

Zdrojový jazyk ***S*** a cieľový jazyk ***T*** si môžeme predstaviť ako náhodné premenné, ktoré produkujú reťazce ako aj vety. K danému reťazcu ***t*** v cieľovom jazyku hľadáme reťazec ***s*** v zdrojovom jazyku, z ktorého prekladač vytvorí ***t***. Vybratím vety ***s***, ktorá je najviac pravdepodobná zvolenému ***t***, minimalizuje možnosť chyby. To znamená, že ***s*** je zvolené tak, aby maximalizovalo ***P(s|t)***.

Z Bayesovho pravidla môže byť rovnosť (2.1) napísaná nasledovne:

$$P(s \mid t) = \frac{P(t \mid s)P(s)}{P(t)} \tag{2.1}$$

Menovateľ ***P(t)*** nezávisí na ***s***, takže postačuje zvoliť ***s***, ktoré maximalizuje súčin ***P(s)P(t|s)***. Prvý faktor v tomto súčine sa nazýva pravdepodobnosť ***s*** v modeli jazyka a druhý faktor je pravdepodobnosť ***t*** daného ***s*** v modeli prekladu.

Vzhľadom na model, najvhodnejším riešením sa zdá byť použitie funkcie ***argmax$_s$***, ktorá vracia argument $\hat{s}$ zo všetkých možných hodnôt ***s*** tak, že maximalizuje nasledujúcu funkciu:

$$\hat{s} = \arg\max_s P(s \mid t) = \arg\max_s \frac{P(t \mid s)P(s)}{P(t)} \tag{2.2}$$

Vďaka faktu, že ***P(t)*** je nezávislé na ***s***, a teda je konštantné pre všetky možné reťazce ***s***, môže byť zanedbané v maximalizačnej procedúre. Základná rovnosť SMT modelu je preto vyjadrená ako nasledujúci problém prehľadávania:

$$\hat{s} = \arg\max_s P(t \mid s)P(s) \tag{2.3}$$

Model prekladu ***P(t|s)*** môže byť vypočítaný z vetovo-zarovnaného paralelného korpusu. Bohužiaľ výpočet ***P(t|s)*** priamo z korpusu je nerealizovateľný z dôvodu riedkeho výskytu dát. Väčšina segmentov v akomkoľvek paralelnom korpuse, povedzme tak veľká ako je možná, bude unikátna. Čo je však ešte horšie, väčšina dostupných viet ***s*** a ***t*** v oboch jazykoch sa nebude vyskytovať v žiadnom trénovacom korpuse a preto nie je možné vypočítať prislúchajúce parametre pre model prekladu. Preto musí byť model prekladu dekomponovaný na distribúciu menších jednotiek, ktoré sa viac frekventovane opakujú v trénovacích dátach a taktiež vystupujú v nevidených dátach. Prvým krokom dekompozície pre všeobecný model prekladu je zvolenie inej náhodnej premennej ***A*** označujúcej zarovnanie medzi podreťazcami (t.j. slovami) v zdrojovom a cieľovom jazyku. Použitím všetkých možných zarovnaní ***a*** medzi ***s*** a ***t*** môžeme vzťah pre model prekladu prepísať nasledovne:

$$P(t \mid s) = \sum_a P(t, a \mid s) \tag{2.4}$$

Zarovnanie v SMT je zvyčajne modelované ako sekvencia skrytých prepojení medzi slovami v reťazci cieľového jazyka a slovami v reťazci zdrojového jazyka. Špeciálne každé slovo v reťazci ***t*** cieľového jazyka je prepojené práve s jedným slovom v reťazci ***s*** zdrojového reťazca, kde si prepojenie môžeme predstaviť vo forme prirodzeného čísla reprezentujúceho pozíciu slova vo vete zdrojového jazyka, na ktoré prepojenie ukazuje.

Väčšina modelov prekladu, ktoré sa používajú v oblasti SMT sú založené na piatich modeloch predstavených v [Brown 1993] pri výskume v IBM, tzv. modely IBM Model 1 až IBM Model 5. Idea skrývajúca sa za týmito piatimi modelmi je začať s veľmi jednoduchým modelom pred prechodom ku komplexnejšiemu. Výstup jednoduchších modelov môže byť použitý k inicializácii nasledujúcich modelov.

Model 1 je najjednoduchším slovným prekladovým modelom. V Modeli 1 a 2 najprv vyberáme dĺžku pre zdrojový reťazec, pričom sa predpokladá, že všetky dĺžky sú si vzájomne podobné. Následne pre každú pozíciu v zdrojovom reťazci rozhodneme, ako ho prepojiť k cieľovému reťazcu a kde je vhodné umiestniť zdrojové slovo v reťazci. V Modeli 1 majú všetky zarovnania rovnakú pravdepodobnosť. Model 2 používa zarovnanie s nulovým poradím, kde rôzne

pozície zarovnania sú nezávislé jedna od druhej. Hoci použitím Modelov 1 a 2 môžeme získať zaujímavé korelácie medzi niektorými pármi frekventovaných slov v oboch jazykoch, tieto modely často nevedú k uspokojivému zarovnaniu.

V Modeloch 3, 4 a 5 [Brown 1993] je posledným krokom rozhodovanie, kde sa prepojenia medzi zdrojovým reťazcom a cieľovým reťazcom odlišujú. V Modeli 3 pravdepodobnosť prepojenia závisí na pozícii prepojenia a na dĺžke zdrojového a cieľového reťazca. V Modeli 4 pravdepodobnosť prepojenia závisí na doplneniach identít zdrojového a cieľového reťazca a na pozíciách akýchkoľvek iných zdrojových reťazcoch, ktoré sú prepojené na ten istý cieľový reťazec. Model 5 je rovnaký ako Model 4 s tým, že v ňom boli odstránené nedostatky Modelu 4.

V oblasti strojového prekladu bolo uskutočnenej veľa práce. Jedným z najznámejších projektov je **EGYPT** [Knight 1999a], toolkit vyvinutý SMT tímom počas letného workshopu v 1999 v Centre pre spracovanie jazyka a reči na Univerzite John-Hopkinsa (CLSP/JHU). Je založený na štyroch utilitách:

- **Whittle** je utilita na prípravu a rozdelenie bilingválneho korpusu na trénovaciu a testovaciu sadu. Whittle generuje súbory tvaru *.snt (formát vstupných súborov vyžadovaných programom GIZA). Whittle je napísaný v jazyku Perl.
- **GIZA** je trénovací program, ktorý sa naučí modely prekladu z bilingválneho korpusu. GIZA je napísaná v C++ s použitím knižnice STL (testované použitím GNU C++).
- **Cairo** je utilita na vizualizáciu zarovnania slov napísaná v Jave.
- **Cairoize** je utilita na generovanie súborov zarovnania vo formáte *.aln (formát vyžadovaný utilitou Cairo). Cairoize je napísaný v C a Perle.

V tejto práci bol taktiež použitý program GIZA++. **GIZA++** je rozšírením pôvodného program GIZA. Navrhol a vytvoril ho Franz Josef Och [Och 2003]. Sú v ňom použité niektoré modely od výskumného teamu IBM z začiatku 1990-tych rokov. Tento program obsahuje nasledovné rozšírenia v porovnaní s GIZA:

- Model 4
- Model 5
- Modely zarovnania založené na triedach slov (software na produkciu slovných tried bol vyvinutý spolu s týmto programom)
- Implementuje HMM model zarovnávania: Baum-Welch trénovanie, Forward-Backward algoritmus, prázdne slová, závislosti na slovných triedach, presuny v modeloch plodností

- Obsahuje verzie Modelu 3 a Modelu 4, ktoré umožňujú trénovanie parametra p_0
- Rôznorodé zjemňovanie techniky pre plodnosť, parametre rozhádzania/zarovnania
- Výrazne efektívnejšie trénovanie modelov plodnosti
- Správnu implementáciu metódy stabilizácie, ako bola popísaná v [Brown 1993], množinu heuristík na vytvorenie poradia, ako zabezpečiť úspešnosť stabilizácie

2.4 Vyhodnotenie

Pre vývojárov systémov je vyhodnotenie spôsob, ako systém zlepšiť, a pre užívateľov slúži na uvedomenie si silných a slabých stránok systémov. V tejto práci sa budeme zaoberať vyhodnotením použitých systémov. Povedať, ktorý systém je dobrý, nie je jednoduchou záležitosťou. Existuje veľa aspektov, ktoré je potrebné vziať do úvahy.

Zarovnanie medzi zdrojovými a cieľovými vetami môže byť dosť komplikované. V niektorých prípadoch je aj pre človeka náročné rozhodnúť, ktoré slová v danej cieľovej vete korešpondujú ku ktorým slovám v jej zdrojovej vete. Pri správnom ohodnotení viacerých systémov zarovnania slov hrajú podstatnú rolu aj subjektívne ľudské hodnotenia.

Ak použijeme na porovnanie výsledkov dvoch systémov nejaký ohodnocovací program, tak z jedného pohľadu to môže pôsobiť nezmyselne, pretože linky zarovnania slov v oboch systémoch môžu byť obe správne alebo obe nesprávne. Takýto výsledok z ohodnocovacieho programu nám teda môže povedať iba ako sa tieto dva výsledky od seba odlišujú. Neexistuje žiadny štandard ako rozhodnúť, ktorý systém je lepší.

Jednou z ciest vyhodnotenia výsledkov v tomto projekte je porovnanie výsledkov zarovnania, prekladov, použitých systémov s názorom človeka separátne, aby sme videli, ktorý z nich je bližšie k zlatému štandardu.

Ďalším spôsobom vyhodnotenia výsledkov bude porovnanie prekladov viet vytvorených jednotlivými systémami so správnou formou použitej zdrojovej vety. Miera kvality bude závisieť na množte správnych zarovnaní slov v prekladu so slovami v zdrojovej vete. Použitím tohoto prístupu taktiež nezískame objektívnu mieru, nakoľko použité bilingválne slovníky môžu obsahovať chyby a naopak

nemusia obsahovať všetky možné tvary vhodných slov, najmä pokiaľ nejaké slovo má v jazyku viacero rovnako vhodných synoným. Takže preklad môže byť síce správny, ale systém ho ohodnotí kvalitatívne menej, nakoľko ohodnocovací program nebude mať natoľko presné dáta, aby dokázal odhaliť túto skutočnosť.

Taktiež tu zohráva úlohu a sémantická správnosť a to, že daná skutočnosť sa dá preložiť do vzhľadovo úplne odlišnej vety, čím ohodnocovací program opäť danú vetu skôr označí na nesprávnu ako za správnu. Na tieto účely je preto najvhodnejšie mať k dispozícii preklady, ktoré sú syntakticky čo najviac podobné s originálnou vetou. Žiaľ ale zabezpečenie tejto vlastnosti je väčšinou medzi jazykmi netriviálne. Avšak keďže v práci boli použité ako jazyky slovenčina a čeština, tak prítomnosť viet syntakticky podobných v oboch jazykoch bola viac dostupná. Túto skutočnosť podporila aj skutočnosť, že paralelný korpus bol zostrojený z dát získaných z tituliek, kde sa väčšina viet bola syntakticky veľmi podobná, a teda strojové vyhodnotenie dosahovalo väčšiu úspešnosť, ako pre všeobecný text.

Medzi ďalšie možnosti ohodnotenia patrí ešte čas behu programu pri preklade a cena. Tiež môže byť vyhodnotená jednoduchosť práce so systémom a jednoduchosť a veľkosť korpusu, s ktorým systém pracuje, a rôzne ďalšie aspekty.

2.5 Zhrnutie

V tejto časti sme si predstavili základné koncepty a techniky práce s paralelným korpusom. Boli rozobraté hlavné metódy na zarovnanie viet. Spomenuli sme si dve hlavné metódy zarovnania slov a to prístup asociovania a prístup odhadovania. Miera asociácie je široko používanou pre identifikáciu prekladových korešpondencií. V štatistickom strojovom preklade sa na určenie parametrov zarovnania medzi slovami v paralelnom korpuse používajú pravdepodobnostné zarovnávacie modely. V tejto kapitole boli navrhnuté metódy vyhodnotenia. Na konci tejto kapitoly bol predstavený krátky úvod pre spôsob vyhodnotenia s použitím zlatého štandardu.

3 Strojový preklad veľmi blízkych jazykov

Oblasť strojového prekladu ma veľmi dlhú históriu, avšak počet skutočne úspešných systémov nie je veľmi pôsobivý. Za určitých podmienok sa podarilo vyvinúť systém, ktorý ušetril peniaze a ľudskú námahu, ale vo všeobecnosti úspešný automatický strojový preklad vyžaduje aplikáciu techník z rôznych oblastí výpočtovej lingvistiky (morfológiu, syntax, sémantiku, analýzu reči, atď.). Avšak bolo ukázané, že ak vytvoríme systém pre dvojicu príbuzných jazykov, tak môžeme dosiahnuť významne kvalitnejšie výsledky [Koehn 2002].

Jeden zo skúmaných prístupov medzi jazykmi slovenčina a čeština bol aj systém ČESÍLKO [Hajíč 2000], ktorý kombinoval systém strojového prekladu zo systémom MAHT (Machine-aided human translation). Predstavíme si ho v tejto kapitole.

3.1 Česko-Ruský MT systém RUSLAN

Prvý pokus overenia hypotézy, že príbuzné jazyky sa ľahšie prekladajú, bol odštartovaný v polovici 80-tych rokov v Prahe. Projekt sa nazýval RUSLAN a zameriaval sa na preklad dokumentov v doméne operačných systémov pre sálové počítače.

Bol to systém založený na pravidlách, implementovaný v Colmeraurerových Q-systémoch. Pozostával z plne označenej morfologickej a syntaktickej analýzy češtiny, transformácií a syntaktickej a sémantickej tvorby ruštiny. Na začiatku projektu sa neuskutočňovala žiadna transformácia vďaka predpokladu, že oba jazyky sú podobne stavané, ale tento predpoklad sa ukázal ako chybný.

Keď bola práca v roku 1990 ukončená, systém pozostával z hlavného slovníka, ktorý obsahoval okolo 8000 slov, doplnený prevodovým slovníkom pokrývajúcim ďalších 2000 slov. Prevodový slovník bol založený na originálnej myšlienke popísanej v [Kirschner 1987]. Využíval fakt, že technické termíny pôvodne pochádzajú (vo väčšine európskych jazykov) z gréckych alebo latinských výrazov prispôsobených do daného jazyku. Tento fakt dovoľoval pri „preklade" technických termínov využiť myšlienku priameho prepisu produkčných koncoviek a jemné prispôsobenie ohýbania slova. Napríklad, slovenské slovo „lokalizácia" a „diskriminácia" môže byť prepísané do češtiny ako „lokalizace" a „diskriminace", kde produkčná koncovka –ácia sa prepíše na –ace. Zo začiatku sa zdalo, že preklad

pomocou produkčných pravidiel môže byť efektívny, ale neskôr sa táto hypotéza ukázala byť chybnou [Bémová, Kuboň 1990].

Pri hodnotení kvality prekladu systémom RUSLAN sa ukázalo, že hrubých 40% vstupných viet bolo preložených korektne, okolo 40% s drobnou chybou opraviteľnou následným ľudským kontrolovaním a asi 20% vstupných viet vyžadovalo značnú opravu. Prvým faktorom zlyhania bola nekompletnosť hlavného slovníka systému, druhým faktorom bol modul syntaktickej analýzy češtiny a svoju rolu zohralo použitie tzv. „sémantických významov" [Hajíč 2000].

Na druhej strane fakt, že oba jazyky dovoľujú vysoký stupeň voľnosti v preusporiadaní slov prispel k istému zjednodušeniu procesu prekladania. Gramatika sa spoliehala na fakt, že existujú iba málo dôležité rozdiely v preusporiadaní slov medzi češtinou a ruštinou.

Zo systému RUSLAN vyplynulo nasledovné ponaučenie:

- ani relatívne jednoduché komponenty (transformačný slovník) nebol dostatočne komplexný pre preklad z angličtiny do češtiny a z češtiny do ruštiny.
- limitované domény textov neexistujú v reálnom svete, preto je potrebné pracovať so slovníkom pokrývajúcim čím väčšiu oblasť zdrojového jazyka.

3.2 Lokalizácia produktov a preklad použitím prekladovej pamäti

Lokalizácia produktov a ich dokumentácie je veľkým problémom pre spoločnosti, ktoré chcú posilniť svoju pozíciu na cudzojazyčných trhoch. Množstvo textu, ktoré je potrebné lokalizovať, stojí spoločnosti príliš veľa nákladov.

Možným riešením sa zdá byť použitie nejakého medzijazyka, avšak existujú rôzne príčiny prečo lokalizácia a preklad nebýva realizovaná cez medzijazyk. Je to najmä tým, že každý medzistupeň prekladu prináša výsledok viac a viac odlišný od originálu.

Jednou z možností, ako sa vysporiadať s týmto problémom je kombinácia MT systému s komerčným MAHT (Machine-aided human translation) systémom. Bol zvolený systém TRADOS Translator's Workbench, ktorý používa tzv. prekladovú pamäť, ktorá obsahuje páry predtým preložených viet zo zdrojového jazyka do cieľového jazyka. Keď ľudský prekladateľ začne prekladať novú vetu, systém sa pokúša porovnať zdrojovú vetu s vetami už uloženými v prekladovej pamäti. Ak sa

to podarí, odporučí sa tento preklad a ľudský prekladateľ rozhodne, či ho použiť, zmodifikovať alebo zamietnuť. Prekladovú pamäť bolo možné exportovať a importovať, čím v tomto systéme bolo umožnené použitie aj externe pripravených dát v inom systéme.

3.3 Strojový preklad veľmi blízkych slovanských jazykov

Do skupiny slovanských jazykov, ktoré sú viac tesne príbuzné jazyky ako čeština a ruština (odhliadnuc od páru srbského a chorvátskeho jazyka, ktoré sú takmer identické) sú jazyky čeština a slovenčina.

Tento fakt viedol k experimentu s automatickým prekladom medzi češtinou a slovenčinou. Systém ČESÍLKO sa usiloval o maximálnu využitie podobnosti oboch jazykov. Používal priamy preklad slovo-na-slovo. Bol testovaný na textoch z domény dokumentácie firemných informačných systémov. Jeho hlavnou úlohou bolo zabezpečovať podporu pri preklade a lokalizácii rôznych technických textov.

3.4 Systém ČESÍLKO

Najväčším problémom prekladu slovo-na-slovo (pre jazyky z veľmi podobnou syntaxou a poradím slov, ale rozdielnym morfologickým systémov) je problém morfologickej nejednoznačnosti jednotlivých slovných foriem. Typ nejednoznačnosti je odlišný v jazykoch s bohatým skloňovaním (väčšina slovanských jazykov) a v jazykoch, ktoré nemajú takú širokú rozmanitosť foriem odvodených z jednoduchej lemy.

Bez analýzy prinajmenšom nominálnych skupín je často veľmi obtiažne vyriešiť tento problém. Riešením tohoto problému bola aplikácia stochasticky založeného odstraňovania nejednoznačností (morfologické tagovač) pre češtinu, ktorého úspešnosť je blízko 92%. Tento systém preto pozostával z nasledujúcich modulov:

- Import vstupov z tzv. „prázdnej" prekladovej pamäte
- Morfologická analýza češtiny
- Odstránenie morfologických nejednoznačností
- Doménovo založené bilingválne slovníky (obsahujúce jedno- a viacslovnú terminológiu)

- Hlavný bilingválny slovník
- Morfologická syntéza slovenčiny
- Export výstupu do pôvodnej prekladovej pamäti

  Detailný popis jednotlivých modulov nájdeme v [Hajíč 2000].

3.5 Vyhodnotenie

Pri vyhodnocovaní kvality tohoto systému sa využila blízka previazanosť medzi týmto systémom a TRADOS Translator's Workbench. Metóda je jednoduchá – ľudský prekladateľ dostal prekladovú pamäť vytvorenú našim systémom a preložil text pomocou tejto pamäti s tým, že mohol voľne robiť akékoľvek zmeny v texte, ktorý bol poskytnutý prekladovou pamäťou. Cieľový text vytvorený ľudským prekladateľom bol potom porovnaný s textom vytvoreným mechanickou aplikáciou prekladovej pamäti na zdrojový text. TRADOS následne vyhodnotil percentuálnu zhodu v tom istom štýle ako normálne vyhodnocoval percentuálnu úspešnosť zhody zdrojového textu s vetami v prekladovej pamäti. Tento systém dosiahol asi 90%-nú zhodu (definovanú TRADOSovým porovnávacím modulom) s výsledkami ľudského prekladateľa, založeného na relatívne veľkej (viac ako 10 000 slov) testovacej vzorke.

Presnosť prekladu dosiahnutá týmto systémom potvrdila hypotézu, že preklad slovo-na-slovo môže byť riešením pre strojový preklad veľmi blízko príbuzných jazykov. Zostávajúce problémy na riešenie sú problémy prekladu jeden-na-viac alebo viac-na-viac, kde nedostatok informácií v slovníkoch zapríčiňuje vznik chýb typu: nezvyčajne umiestnené slovo tak, kde neparí, prípadne absencia štylisticky vhodného slova medzi dvoma správnymi.

3.6 Zhrnutie

V tejto kapitole sme si predstavili prvé pokusy o vytvorenie strojového prekladača medzi veľmi príbuznými slovanskými jazykmi. Systémy boli pravidlovo založené, prípadne používali prekladový slovník doplnený prekladovou pamäťou.

4 Systémy použité v tejto práci

Táto práca predstavuje najmä prezentáciu mnou vyvinutého prekladového systému ***VocaTrans***. Pre jeho prácu bolo potrebné vytvoriť a pripraviť dáta na použitie, na čo boli použité vlastné podporné prostriedky, utility, ale taktiež aj rôzne iné dostupné toolkity a utility. Taktiež vo fáze hodnotenia som na porovnanie úspešnosti využil výstupy iných dostupných systémov strojového prekladu. V tejto kapitole si ich predstavíme.

4.1 CMU toolkit pre štatistický jazykový model

CMU toolkit pozostáva z viacerých nástrojov pre tvorbu štatistických jazykových modelov (LM) a testovanie ich kvality. Posledná verzia číslo 2 bola vyvinutá autormi P.R. Clarkson a R. Rosenfeld [Clarkson a Rosenfeld 1997] na Carnegie Mellon University. Táto verzia nie je limitovaná na vytváranie iba bigramov a trigramov, ale umožňuje aj vytvárať n-gramy ľubovoľnej veľkosti. Taktiež podporuje viacero spôsobov orezania veľkosti a vyhladzovania okrajových oblastí. Jazykový model vytvorený týmto systémom sa v spojení s prekladovým modelom vytvoreným v systéme GIZA++ [Kapitola 4.2] používa v štatistickom prekladači ISI ReWrite Decoder [Kapitola 4.3].

4.1.1 Problém toolkitu CMU

Za veľmi podstatný problém tohto systému považujem jeho obmedzenie na maximálny počet 65535 slov v slovníku. Následne vytvorené n-gramy už pozostávajú iba z kombinácií slov uvedeného malého slovníka. Tento fakt je spôsobený tým, že jednotlivým slovám sú priradené čísla, ktoré sa ukladajú do dvojbajtových premenných, čo síce umožňuje jeho efektívnu veľkosť a menšie pamäťové nároky, ale množstvo slov, ktoré týmto odstránime z modelu jazyka je výrazné.

Tento jazykový model bol testovaný aj s implementovaným systémom VocaTrans, a dosahoval horšie výsledky, ale iné testované jazykové modely.

4.1.2 Vytvorenie jazykového modelu z korpusu

Základnou formou tvorby jazykového modelu je jeho vytvorenie z korpusu textov daného jazyka. Dôležitým krokom je označenie jednotlivých viet značkami začiatku „<s>" a konca „</s>" vety.

Následne použitím utilít:

- *text2wfreq < text > wfreq*
- *wfreq2vocab -top 65530 < wfreq > vocab*
- *text2idngram -n 3 -vocab vocab -temp . < text > idngram*
- *idngram2lm -idngram idngram -vocab vocab -n 3 -binary binlm -buffer 1000 -cutoffs 1 1 -witten_bell -context ccs*

získavane jazykový model zložený z trigramov, s orezaním pre bigramy a trigramy a s vyhladením okrajových oblastí metódou Witten Bell. Nachádza sa v binárnom tvare, ktorý je potrebný pre ISI ReWrite Decoder [Kapitola 4.3]. Pre implementovaný systém VocaTrans použijeme jeho verziu v štandardnom ARPA formáte použitím utility *binlm2arpa*.

Súbory používané pri tvorbe CMU jazykového modelu obsahujú nasledovné informácie:

- *wfreq* – obsahuje slová a ich početnosť výskytu v korpuse
- *vocab* – slovník maximálne 65535 slov, ktoré budú použité v LM
- *wngram* - súbor s n-gramami tvaru: *n-gram početnosť*
- *idngram* – súbor s n-gramami, kde slová sú reprezentované číslami podľa poradia v slovníku *vocab*
- *ccs* – súbor obsahujúci kontextové značku <s> a </s>
- *binlm* – jazykový model v binárnom formáte
- *arpa* – jazykový model v štandardizovanom formáte ARPA

4.1.3 Vytvorenie jazykového modelu z frekvenčných tabuliek

Pokiaľ nemáme priamo prístup k dostatočne veľkému jazykovému korpusu, je možné využiť na získanie potrebných dát aj miestny jazykový ústav, ktorý

spravuje lokálny alebo národný jazykový korpus v danej krajine. Štandardný formát poskytnutých dát býva vo forme, kde „početnosť" je počet výskytov vzorky v korpuse:

pre unigramy: *slovo početnosť*

pre bigramy: *slovo slovo početnosť*

pre trigramy: *slovo slovo slovo početnosť*

Aj na dáta tohoto formátu môžeme využiť CMU toolkit na zostavenie jazykového modelu. Frekvenčnú tabuľku unigramov pomenujeme *wfreq,* frekvenčnú tabuľku trigramov pomenujeme *wngram*. Následne použitím krokov:

- *wfreq2vocab -top 65530 < wfreq > vocab*
- *wngram2idngram -n 3 -write_ascii -buffer 1000 -vocab vocab -temp . < wngram > idngram*
- *idngram2lm -idngram idngram -vocab vocab -n 3 -arpa arpa -buffer 1000 -cutoffs 0 0 -witten_bell -context ccs -ascii_input -four_byte_counts*

získame maximálny jazykový model v ARPA formáte, bez orezania, ktorý je možné naimportovať pre implementovaný systém ***VocaTrans***, ale bude mať obmedzenie na slovník 65530 slov, ktoré je dané práve CMU toolkitom.

Taktiež je možné vytvoriť jazykový model aj s inými formami zjemnenia okrajov, prípadne orezaním. Viac informácii v dokumentácii [Clarkson 1997].

4.2 Systém GIZA++

GIZA++ je rozšírením pôvodného programu GIZA (súčasť SMT toolkitu EGYPT), ktorý bol vyvinutý tímom Štatistického strojového prekladu počas letného workshopu v roku 1999 v Centre pre spracovanie jazyka a reči na Univerzite Johna Hopkinsa (CLSP/JHU). Je to program, ktorý sa učí štatistické strojové modely z bitextov. GIZA++ zahrňuje viacero dodatočných vylepšení, ktoré navrhol a implementoval Franz Josef Och [Och 2003].

4.2.1 Vstupné formáty programu GIZA++

Prvým dôležitým krokom je spustenie programu ***plain2snt.out***, ktorý z textu vytvorí súbory ****.vcb*** a ****.snt***. Program ***plain2snt.out*** je jednoduchý

nástroj na transformáciu čistého textu do formátu potrebného pre GIZA++. Tieto súbory sú vstupnými súbormi pre základný beh GIZA++.

VCB je skratka je slovníkový súbor (Vocabulary file). Každý záznam je uložený v jednom riadku nasledovne:

unikátne_id1 reťazec1 počet výskytov1
unikátne_id2 reťazec2 počet výskytov2
unikátne_id3 reťazec3 počet výskytov3
...

SNT je bitextový súbor. Každý jazykový pár je uložený v troch riadkoch. Vezmime si napríklad česko-slovenský jazykový pár:

Už jste měl prsten dost dlouho
Myslím, že máte ten prsteň u seba už príliš dlho

Potom SNT súbor pre tento pár bude vyzerať nasledovne:

1
2 3 4 5 6 7
2 3 4 5 6 7 8 9 10 11

Prvý riadok predstavuje počet výskytov tohoto jazykového páru. V druhom riadku sa nachádza zdrojová veta, v ktorej je každé slovo nahradené unikátnym id zo slovníkového súboru a v treťom riadku je cieľová veta v rovnakom formáte.

Po spustení programu ***mkcls*** sa vytvoria ďalšie štyri súbory. V súbore ****.vcb.classes*** sú vypísané všetky slová unikátne z celého textu a počet výskytov daného slova. V súboroch ****.vcb.classes.cats*** sa zasa nachádza informácia do ktorej triedy prislúcha každé slovo.

Spustenie ***mkcls*** je voliteľný krok, ktorý nie je nevyhnutný. Utilita ***mkcls*** slúži na trénovanie tried použitím kritéria Maximum-likelihood. Výsledné slovné triedy sú špeciálne navrhnuté pre jazykové modely alebo modely štatistického strojového prekladu. Program ***mkcls*** napísal Franz Josef Och [Och 2001].

4.2.2 Výstupné formáty programu GIZA++

GIZA++ vytvorí po spustení 17 výstupných súborov. V nasledujúcom texte si krátko predstavíme časť dôležitých výstupných formátov na poskytnutie základnej myšlienky, v čom spočívajú výstupy programu GIZA++.

- **a3.final**

  Každý riadok v súbore a3.final pozostáva z nasledovných údajov:

  i j l m p(i | j, l, m)

  kde i, j, l, m sú všetky typu integer a

  j = pozícia v cieľovej vete

  i = pozícia v zdrojovej vete

  l = dĺžka zdrojovej vety

  m = dĺžka cieľovej vety

  p(i | j, l, m) je pravdepodobnosť, že slovo v zdrojovej vete na pozícii i je premiestnené na pozíciu j vo vetnom páre dĺžky l a m.

- **t3.final**

  Každý riadok je nasledovného formátu: *s_id t_id P(t_id | s_id)*, kde

  s_id = unikátne id pre slovo v zdrojovej vete

  t_id = unikátne id pre slovo v cieľovej vete

  P(t_id | s_id) = pravdepodobnosť, že sa s_id preloží na t_id

  Podobné súbory sú vytvorené s prefixom ***prob_table.actual.xxx***, ktoré obsahujú namiesto idčok patričné slová. Podobný princíp je uplatnený aj pre tabuľky plodnosti. Pre finálnu tabuľku je taktiež vytvorená tabuľka s inverznými pravdepodobnosťami a má infix ***ti***.

- **Revidované slovníkové súbory (*.src.vcb, *.trg.vcb)**

  Revidované slovníkové súbory sú podobného formátu ako originálne slovníkové súbory (VCB). Jediným rozdielom je, že frekvencia pre každé slovo je počítaná z poskytnutého korpusu, ktorý nie je požadovaný na vstupe.

  V súbore ****.src.vcb*** každý riadok pozostáva z informácie zo zdrojového korpusu:

  Id_slova
  Slovo
  Počet výskytov v korpuse

V súbore ***.trg.vcb*** každý riadok pozostáva z podobných informácií, ale z cieľového korpusu v cieľovom jazyku.

- **d4.final**

Tento súbor obsahuje informácie o pozícii pre hlavičkové slová a pre slová v tele vety. Pre slová v hlavičke obsahuje nasledovné informácie:

Trieda zdrojového slova
Trieda cieľového slova
Pozícia
Pravdepodobnosť

Pre slová v tele vety to vyzerá nasledovne:

Trieda cieľového slova
Pozícia
Pravdepodobnosť

- **ti.final**

Tu je zaznamenaná pravdepodobnosť prekladu, ale v inverznom smere. Každý riadok obsahuje:

Id cieľového slova
Id zdrojového slova
Pravdepodobnosť prekladu

- **n3.final**

Tu je tabuľka plodnosti. Každý riadok v tomto súbore má nasledovný tvar:

id_zdrojového_slova p0 p1 p2 ... pn

p0 je pravdepodobnosť, že zdrojové slovo má nulovú plodnosť, *p1* pre plodnosť jedna a *pn* predstavuje maximálnu plodnosť definovanú programom.

- **p0-3.final**

Tento súbor obsahuje iba jeden riadok s jedným reálnym číslom, ktoré predstavuje hodnotu P0, pravdepodobnosť nevkladania slova NULL.

- ***.gizacfg**

Tento súbor obsahuje všetky nastavenia parametrov, ktoré boli použité v danom poradí pri trénovaní. To znamená, že pri spustení programu GIZA s parametrom tohto súboru sa uskutoční (môže uskutočniť) rovnaké trénovanie.

- **Perplexita (*.perp)**

Tento súbor sa vytvorí na konci trénovania. Sumarizuje hodnoty perplexity pre každú iteráciu trénovania. Rovnaký formát je aj pre krížovú entropiu. Ak nebol poskytnutý žiadny testovací korpus, hodnoty budú nastavené na N/A.

- **A3.final**

V každom procese iterácie a pre každý jazykový pár v trénovacej množine sa do tohto súboru zapíše najlepšie zarovnanie (zarovnanie Viterbi), ak je to príslušne nastavené v parametroch. Súbor zarovnania je pomenovaný ***prob_table.An.i***, kde *n* je číslo modelu ({1, 2, 2to3, 3, 4}) a *i* je číslo iterácie. Formát súboru zarovnania je znázornený na nasledujúcom príklade:

 # Sentence pair (1) source length 9 target length 10 alignment score : 1.09989e-05
Radšej by som zomrel vo vezení než ho nechal ísť
NULL ({ }) Radši ({ 1 }) bych ({ 2 3 }) zemřel ({ 4 }) ve ({ 5 }) vězení ({ 6 }) než ({ 7 }) ho ({ 8 }) nechal ({ 9 }) jět ({ 10 })
Sentence pair (2) source length 10 target length 10 alignment score : 2.16436e-05
Jediné miesto kde pôjde je naše skúšobné centrum na Falklandách
NULL ({ }) Jediné ({ 1 }) místo ({ 2 }) kam ({ 3 }) půjdete ({ 4 }) je ({ 5 }) naše ({ 6 }) zkušební ({ 7 }) centrum ({ 8 }) na ({ 9 }) Falklandech ({ 10 })

V súbore zarovnania je každý vetný pár reprezentovaný troma riadkami. V prvom riadku je označenie, ktoré môže byť použité, okrem iného, ako popis do programu na vizualizáciu zarovnania. Obsahuje informáciu o poradovom čísle vety v trénovacom korpuse, dĺžky viet a pravdepodobnosť zarovnania. Druhý riadok predstavuje cieľovú vetu, v tomto príklade v slovenčine. Tretí riadok je zdrojová veta. Za každým riadkom v zdrojovej vete sa nachádza množina nula alebo viac čísiel. Tieto čísla reprezentujú pozície v cieľovej vete, na ktoré slová je prepojené dané slovo zdrojovej vety v závislosti na zarovnaní.

4.3 ISI ReWrite Decoder

Proces prekladu v štatistickom strojovom preklade sa nazýva aj „dekódovanie". Pomocou vytvoreného jazykového modelu a modelu prekladu sa vytvára samotný preklad medzi jazykmi. Avšak problém vytvorenia správneho prekladu je NP-úplný problém, a preto sa pri reálnej prevádzke používajú aproximačné greedy algoritmy, ktoré umožňujú vytvoriť optimálne riešenie, nie nutne to najsprávnejšie, v reálnom čase.

Jedným z týchto systémov je aj ISI ReWrite Decoder [Germann 2001], ktorý pre svoj preklad využíva práve jazykové modely vytvorené systémami CMU a GIZA++.

Systém beží ako služba na zvolenom porte a prijíma vstup v xml formáte. Po preklade poskytne výsledok ako odpoveď na zaslaný vstup v nadviazanom spojení.

4.4 SMT QuickRun

SMT QuickRun [Cuřín 1.2] je balíček skriptov na prípravu trénovacích dát do potrebného formátu pre použitie na tréning jazykového a prekladového modelu pomocou utilít CMU a GIZA++ a následné vytvorenie prekladového servera ISI ReWrite Decoder.

Tento balíček vyžaduje ako vstupy jazykový korpus cieľového jazyka a na trénovanie modelu prekladu bilingválne texty. Skripty natrénujú jednotlivé modely a pripravia konfiguračné súbory a spúšťacie skripty pre prekladač.

SMT QuickRun vytvoril Ján Cuřín pôvodne na preklad medzi jazykmi angličtina-čeština. Obsahuje aj vzorové dáta 10000 viet z kanadského Handsard korpusu pre vytvorenie malého prekladača medzi jazykmi angličtina-francúzština.

4.5 PC Translator 2004

Na porovnanie úspešnosti jednotlivých systémov boli použité aj automatické preklady z komerčného systému PC Translator 2004 [Teos 2004]. Tento systém nepoužíva pri preklade štatistické dáta, používa sa slovníkový preklad, pričom obsahuje preklady jeden-na-jeden ako aj preklady viac-na-viac slov.

Tento systém obsahuje slovníky medzi desiatimi jazykmi, kde jednou z dostupných kombinácií je aj česko-slovenská verzia obsahujúca 190 000 slovníkových párov.

Systém umožňuje prácu v dvoch režimoch:

- manuálny režim
- automatický preklad

Pri manuálnom režime systém rieši problém nejednoznačností (viacerých možností cieľového prekladu slova) zobrazením prvých x možností zo slovníka, kde tento počet je konfigurovateľný. Užívateľ si následne vyberá z poskytnutých možností a až následne je zostavený preklad.

Pri automatickom preklade sa do prekladu zahrnie prvá nájdená možnosť. Metóda výberu prvej nájdenej možnosti intuitívne dáva tušiť, že vytvorený preklad nie je veľmi použiteľný, čo si aj následne ukážeme pri porovnaní jednotlivých systémov [Kapitola 6].

Medzi ďalšie súčasti programu patrí multimediálny slovník a online prekladač webových stránok, ktorý je určený na orientačný preklad www stránok.

4.6 Zhrnutie

V tejto kapitole sme si predstavili základné prostriedky použité pri príprave dát a tvorbe prekladu. Sú to systémy, z ktorých výsledkami budú porovnávané výsledky implementovaného systému ***VocaTrans***.

Na čisto štatistický strojový preklad sa využívajú systémy:

- CMU Statistical Language Modeling Toolkit
- GIZA++
- ISI ReWrite Decoder
- SMT QuickRun

Na čisto slovníkový preklad boli pre porovnanie použité preklady zo systému:

- *PC Translator 2004*

V nasledujúcej kapitole si ukážeme ako tieto dva prístupy skĺbiť do ešte efektívnejšieho systému slovníkovo-štatistického prístupu implementovaného v systéme ***VocaTrans***.

5 Slovníkovo-štatistický systém VocaTrans

Základným problémom pri slovníkovom preklade sú nejednoznačnosti. Ideálnym prípadom je, keď jedno dané slovo má práve jednu formu prekladu, ale v štandardnom prípade má takmer každé slovo viacero možných prekladov. Je to dané najmä tým, že slovo sa môže prekladať na viacero synomým cieľového slova. V tomto prípade to nemusí mať pri výbere ľubovoľného synonyma dopad na význam prekladu. Problémom ale je, že dané slovo má v cieľovom jazyku viacero sémantických významov, čo znamená, že ak je pri preklade zvolený nesprávny význam, vo väčšine prípadov výsledný text nemusí dávať rovnaký význam, prípadne cieľový text nedáva zmysel vôbec.

Pri čisto štatistickom prístupe sa problém výberu nejednoznačností rieši pravdepodobnostnými metódami výberu vhodných slov na základe jazykového modelu. Problém zhotovenia množiny nejednoznačností pre zdrojové slovo sa zase získava pomocou modelu prekladu.

Kvalitný jazykový model je možné získať z národného korpusu, ale ukázalo sa, že získanie kvalitného modelu prekladu pre zvolenú kombináciu jazykov (slovenčina-čeština) až taký jednoduchý nie je. Spočíva to najmä v tom, že väčšina osôb rozprávajúcich jedným z týchto jazykov rozumie veľmi dobre aj druhému jazyku, a tak množstvo zdrojov bilingválnych česko-slovenských textov je minimálne a ich dostupnosť je obmedzená.

Na základe týchto skutočností bola pre implementáciu zvolená nasledovná forma prekladača, ktorý by vyriešil oba nedostatky:

- na jednej strane problém nejednoznačnosti výberu správneho tvaru cieľového slova pre preklad sa vylepší využitím dostupných dát, a to dát z korpusu jazyka – štatistickým jazykovým modelom
- na druhej strane nedostupnosť bilingválnych textov, potrebných na natrénovanie modelu prekladu, sa odstráni použitím slovníkových jazykových párov.

V tejto kapitole si predstavíme podporné utility a hlavné myšlienky použité pri implementácii slovníkovo-štatistického strojového prekladača ***VocaTrans***.

5.1 Vybudovanie prekladového slovníka

Na identifikáciu nejednoznačností sa teda budú používať slovníkové páry. Štandardne slovníky obsahujú medzi dvoja jazykmi najmä páry pre slovo v základnom tvare. Avšak slovenčina aj čeština sú jazyky s bohatou ohybnosťou a pre jedno slovo je prípustných tvarov vo všetkých prípustných morfologických kategóriách niekedy aj zopár desiatok. Aby sme v našom slovníku našli aspoň nejakú nejednoznačnosť pre dané vstupné slovo, potrebujeme aby slovník obsahoval aspoň po jednom zázname pre každý tvar každého slova, ktoré je prípustné v jazyku, alebo aspoň pre čo najväčšiu množinu slov v jazyku.

5.1.1 Tvorba prípustných slov v jazyku

Na začiatok je potrebné identifikovať prípustné slová v obidvoch jazykoch. Tu sa ako vhodná možnosť naskytli slovníky pre programy na kontrolu pravopisu, ktoré obsahujú informácie o všetkých možných prípustných tvaroch je každé kmeňové slovo.

Pri implementácii boli použité slovenské a české slovníky na kontrolu pravopisu z open source projektov ispell a aspell [Kuenning], ktoré sa používajú napr. aj vo voľne dostupnom kancelárskom balíčku OpenOffice.

Tieto slovníky obsahujú základné tvary slov s flagmi, ktorými sa označujú možnosti ohybu daného slova. Systém umožňuje pre konkrétne slovo s flagmi vygenerovať všetky jeho tvary podľa určených flagov. Týmto získame oveľa väčšie pokrytie slovnej zásoby jazyka, ako len základnými tvarmi.

Samozrejme, oveľa dôkladnejšie pokrytie získavame z jazykového korpusu, avšak pri vygenerovaní tvarov pomocou slovníka na kontrolu pravopisu sme získali k daným slovám ešte dodatočný sémantický význam, a to identifikáciu všetkých tvarov daného slova s jeho kmeňovým slovom, čo v jazykovom korpuse nie je obsiahnuté.

Následne zo získaných tvarov slov potrebujeme vygenerovať bilingválny prekladový slovník. Identifikáciu korešpondujúcich množín slov nám zabezpečí bilingválny slovník základných tvarov, ale aby sme obmedzili množstvo kombinácií korešpondencií medzi jednotlivými tvarmi, použijeme trocha morfológie.

5.1.2 LexSkloňovač pre slovenský jazyk

Keďže slovenčina aj čeština majú veľmi podobnú morfológiu, až na zopár výnimiek, tak priradenie morfologických kategórií slovám nám umožní ich korektnejšiu korešpondenciu. Vychádzame tu zo základného faktu, že rovnaký pád v slovenčine sa mapuje na rovnaký pád v češtine, podobne je to aj s časmi pri slovesách a stupňoch a rodoch pri prídavných menách.

Na morfologické tagovanie slov z vygenerovaného slovníka prípustných slovenských slov bol použitý systém LexSkloňovač, ktorý vytvoril v roku 2001 Michal Čerešňa [Čerešňa].

Systém umožňuje pre vstupné slovo a určený vzor vygenerovať podľa tohoto vzoru všetky tvary vstupného slova. Keďže LexSkloňovač používal na skloňovanie len pre podstatné mená 120 vzorov a slovníky na kontrolu pravopisu obsahovali len základných 12 vzorov, bolo potrebné v LexSkloňovači identifikovať správny vzor pre dané vstupné slovo.

Na tento účel bola použitá pravdepodobnostná metóda výberu správneho vzoru podľa množstva správne vyskloňovaných slov v závislosti od množiny možných tvarov slova poskytnutej zo systému na kontrolu pravopisu. Použitá metóda, až na niekoľko výnimiek, fungovala korektne, čím sme získali pre každé slovenské slovo z množiny získanej zo slovníkov kontroly pravopisu všetky jeho vyskloňované tvary aj s určenými kategóriami.

Príklad výstupu zo systému LexSkloňovač:

vyskloňované slovo	kategória	kmeňové slovo	slovný druh
adresár	*Nsg*	*adresár*	*k1*
adresára	*Gsg*	*adresár*	*k1*
adresáru	*Dsg*	*adresár*	*k1*
adresár	*Asg*	*adresár*	*k1*
adresári	*Lsg*	*adresár*	*k1*
adresárom	*Isg*	*adresár*	*k1*
adresáre	*Npl*	*adresár*	*k1*
adresárov	*Gpl*	*adresár*	*k1*
adresárom	*Dpl*	*adresár*	*k1*
adresáre	*Apl*	*adresár*	*k1*
adresároch	*Lpl*	*adresár*	*k1*
adresármi	*Ipl*	*adresár*	*k1*

Príklad 5.1 Ukážka výstupu z programu LexSkloňovač

5.1.3 Morfologický analyzátor češtiny AJKA

Pre morfologické tagovanie slov českého jazyka bol zvolený program AJKA, ktorý vytvoril v roku 1999 Radek Sedláček [Sedláček 1999]. Tento analyzátor pracuje odlišným spôsobom ako LexSkloňovač.

Systém umožňuje pri patričnom nastavení prepínačov pre dané vstupné slovo vygenerovať zoznam všetkých kmeňových slov a k nim prípustných morfologických kategórií:

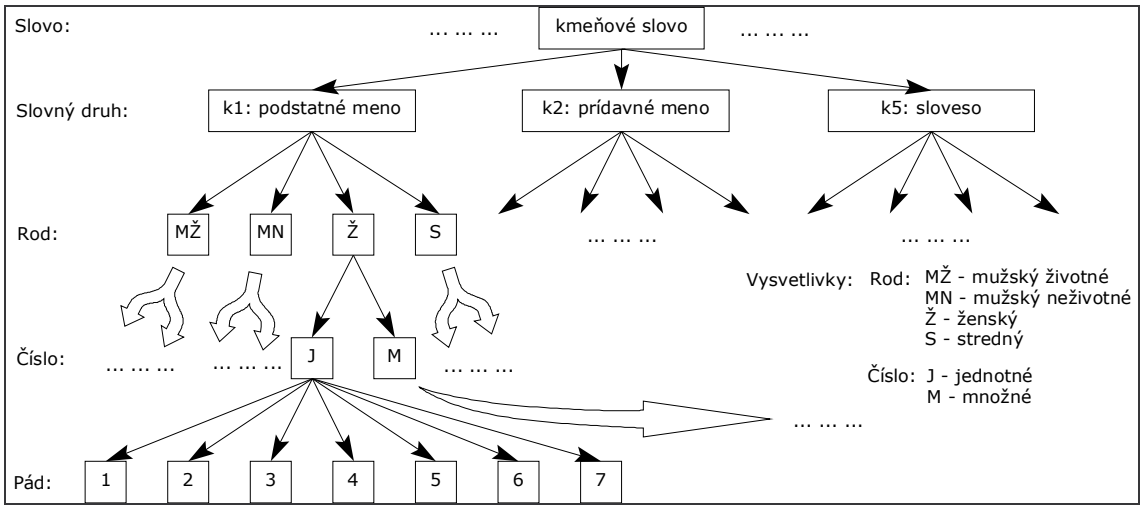
bachratí <l>bachratit <c>k5mItPp3nPaI
<l>bachratit <c>k5mItPp3nSaI
<l>bachratý <c>k2gMnPc1d1 <c>k2gMnPc5d1
<l>bachratět <c>k5mItPp3nSaI <c>k5mItPp3nPaI
<l>bachratět <c>k5mItPp3nSaI <c>k5mItPp3nPaI

Príklad 5.2 Ukážka výstupu z morfologického analyzátora češtiny AJKA

(význam jednotlivých identifikačných značiek sa nachádza v dokumentácii k programu AJKA [Sedláček 2005])

Výstupu programu AJKA bol reorganizovaný tak, aby jednotlivé slová boli zoskupené podľa kmeňového slova. Na reorganizáciu bola implementovaná utilita *CategorizeCZ*.

Táto utilita postupne zo vstupu zaraďuje jednotlivé slová na základe kmeňového slova a jednotlivých identifikačných značiek do stromu nasledovne:



Obrázok 5.1 Triediaca štruktúra úpravy výstupu programu AJKA do podoby vhodnej na párovanie s LexSkloňovačom

5.1.4 Identifikácia korešpondujúcich párov podľa morfológie

Vzniknuté dáta tvaru „koreň slovo kategória" boli následne popárené pomocou bilingválneho slovníka základných tvarov.

Pre bilingválny koreňový pár boli vybrané príslušné množiny vyskloňovaných slov s príslušnými kategóriami. Následne sa korešpondujúce jazykové páry identifikovali podľa zhodnosti signifikantných značiek morfologických kategórií, čím sa výrazne zredukovalo množstvo kombinácií, ktoré by sme inak mali medzi všetkými tvarmi príslušného slova.

Použitie signifikantných morfologických značiek (pád, rod, druh, číslo) pre určenie zhodnosti bolo možné kvôli dostatočnej zhodnosti týchto kategórií medzi jazykmi čeština a slovenčina.

Pre párovanie bola použitá implementovaná utilita *PairByWordList*, ktorá na vstupe očakáva:

- súbor s bilingválnym slovníkom koreňových párov
- identifikáciu smeru tvorby: cz alebo sk
- súbor s českými slovami s kategóriami
- súbor so slovenskými slovami s kategóriami
- meno súboru do ktorého za zapíše výsledný slovník pre zvolený smer

Formát bilingválneho slovníka koreňových párov:

- české slovo zarovnané vpravo medzerami na 39 znakov
- oddeľovač „|"
- slovenské slovo zarovnané vpravo medzerami na 39 znakov
- oddeľovač „|" + koniec riadka

Pozícia:	1	2	3	4	5 38	39	40	41	42	43	44	45 79	80
Koreňový pár:	ž	l	u	č			|	ž	l	č			|

Obrázok 5.2 Schéma formátu bilingválneho slovníka koreňových párov

Identifikácia smeru je implementovaná kvôli tomu, že v slovníku koreňových párov sa môže nachádzať viacero synoným, potom vo výsledku sa budú nachádzať príslušné kombinácie pre všetky cieľové tvary.

Po skončení programu získavame popárený prekladový slovník pre zvolený smer prekladu. Pre preklad oboma smermi je potrebné vygenerovať zvlášť prekladový slovník pre každý smer prekladu.

5.2 Databázový model

Pri tomto projekte využívame najmä štatistické dáta, a aby bolo pokrytie daných jazykov čo najúčinnejšie, je potrebné, aby pri behu bola použité dáta, ktoré budú obsahovať vzorky dát dostatočnej jemnosti, čím sa dostávame k desiatkam miliónov záznamov.

Taktiež vyhľadávanie potrebných údajov v takomto množstve dát je netriviálne, a tak kvôli jednoduchosti neboli implementované žiadne vlastné dátové štruktúry za týmto účelom. Systém ***VocaTrans*** používa ako úložisko dát niektorú z dostupných databáz. Keďže systém VocaTrans je implementovaný v programovacom jazyku Java, je možné použiť akúkoľvek databázu, na ktorú existuje JDBC rozhranie.

Databáza nám poskytuje dostatočne robustný aparát na rýchle získanie potrebných údajov pri preklade. Po prvotnom importe údajov používaných pri preklade (prekladové slovníky, jazykový model) sa databáza používa už len na vyhľadávanie.

Pre každý smer preklady jazyka obsahuje databáza 4 tabuľky:

- prekladový slovník: *zdroj2ciel*
- unigramy jazykového modelu cieľového jazyka: *ciel_words*
- bigramy jazykového modelu cieľového jazyka: *ciel_bigram*
- trigramy jazykového modelu cieľového jazyka: *ciel_trigram*

Štruktúra databázových tabuliek (zdroj=f, cieľ=sk):

f2sk	
id_	int PK
source_	varchar(31)
target_	varchar(31)
category_	varchar(12)

sk_words	
id_	int PK
word_	varchar(31)
count_	int
prob_	float

sk_bigram	
id_	int PK
word1_	int
word2_	int
count_	int
prob_	float

sk_trigram	
id_	int PK
word1_	int
word2_	int
word3_	int
count_	int
prob_	float

Tabuľka 5.1 Typy položiek v databázových tabuľkách

Za účelom efektívnosti spracovania a úspore potrebného miesta sú jednotlivé slová v dátových štruktúrach pre bigramy a trigramy reprezentované vo forme poradového čísla slova z tabuľky unigramov. Pre každý n-gram máme v databáze uchovanú informáciu o počte výskytov z korpusu a príslušnej pravdepodobnosti výskytu slova v korpuse v logaritmickom tvare.

Údaje o pravdepodobnosti sú v databáze uložené v logaritmickom tvare preto, lebo na veľkom jazykovom korpuse dosahujú veľmi malú pravdepodobnosť, ktorá z dôvodu nepresnosti reprezentácie reálnych čísiel v počítačoch by mohla byť značne skreslená. Z tohoto dôvodu sú tieto údaje logaritmované a normalizované. [Knight 1999b].

Z pohľadu efektivity vyhľadávania pre databázu v takomto type aplikácie je veľmi podstatné, aby nad ňou boli postavené indexy pre typizované operácie použitia. Počas testovania boli okrem primárnych indexov identifikované aj nasledovné indexy, ktoré výrazne zvyšujú výkon pri vyhľadávaní (zdroj=f, cieľ=sk):

tabuľka	index
f2sk	source_
sk_words	word_
sk_bigram	word1_
sk_bigram	word2_
sk_bigram	word1_, word2_
sk_bigram	word2_, word1_
sk_trigram	word1_, word2_, word3_
sk_trigram	word3_, word2_, word1_

Tabuľka 5.2 Zoznam použitých indexov nad databázovými tabuľkami

Poznámka: Počas implementácie utilít na naplnenie databázy predmetnými údajmi bolo zistené, že vkladanie údajov pomocou SQL príkazov pracuje pri danom množstve príliš neefektívne, a preto všetky utility pripravujúce dáta pre import vytvárajú výstupy vo forme CSV (comma separated values) súborov.

5.3 Spracovanie frekvenčných tabuliek z národných korpusov

V tejto práci boli na zostrojenie štatistického jazykového modelu použité dáta z národných jazykových korpusov:

- slovenský národný korpus [Korpus prim0.2]
- český národní korpus [Korpus SYN2000]

Poskytnuté dáta sú vo formáte: *vzorka početnosť*

Korpusové dáta sú otagované kontextovými značkami začiatkov „<s>" a koncov viet „</s>". Kvôli množstvu dát bol zvolený cuf-off pre bigramy a trigramy na 1. Na prípravu CSV súborov zo vstupných dát boli implementované utility, ktoré ako vstup očakávajú jazykový identifikátor a príslušný korpusový súbor:

- *ngram.AddUnigram* – pre spracovanie unigramov
- *ngram.AddBigram* – pre spracovanie bigramov
- *ngram.AddTrigram* – pre spracovanie trigramov
- *ngram.ARPA2csv* - prípravu dát pre import vytvorených z korpusu pomocou CMU toolkitu [Kapitola 4.1]

Ak databáza nepodporuje priamy import CSV súborov, je možné ich naimportovať pomocou *util.ImportCSVtoDB*. Táto utilita vkladá údaje z CSV súboru do databázy postupne po riadkoch samostatným SQL príkazom. Podľa skúseností je to značne pomalšie ako import CSV priamo prostriedkami databázy.

5.4 Prekladový engine systému VocaTrans

Hlavnou programovou súčasťou systému VocaTrans je práve systém na preklad medzi dostupnými jazykmi, v našom prípade slovenčina (označovaná „sk") a čeština (označovaná „f"). Na základe identifikačných znakov jazyka je možné systém neskôr rozšíriť aj pre iné jazykové kombinácie, po doplnení patričných dátových štruktúr (slovníky prekladových párov a korpus cieľového jazyka) medzi danou jazykovou kombináciou.

V tejto kapitole si predstavíme hlavnú funkcionalitu jadra systému, ako aj sprievodných utilít, ktoré využívajú pre svoju činnosť funkcie implementované v jadre prekladového engine. Ďalej si predstavíme utility na analýzu a hodnotenie vstupov a výstupov z prekladu.

5.4.1 Použité dátové štruktúry

Ako vstup do prekladu sa predpokladá blok textu, ktorý môže byť napr. jedna veta, prípadne viac. Vstupná veta sa doplní kontextovými značkami začiatku a konca vety, ak nie sú prítomné, zároveň sa tieto značky vsunú medzi identifikáciu konca vety a začiatok ďalšej, ak vstupný text pozostáva z viacerých viet.

Kontextové značky označujúce vety sú dôležité z pohľadu korpusu premietnutého do jazykového modelu, nakoľko štatistické pravdepodobnosti umiestnenia slova v kontexte začiatku, alebo konca vety sú iné ako v prípade pozície niekde v rámci vety.

Blok textu sa rozdelí na tokeny podľa „white space" znakov. Takýto token sa obalil do štruktúry *SuitableWords*.

SuitableWords v danom tokene identifikuje prefix a sufix zo znakov, ktoré nie sú alfabetické. Tento prefix a sufix za odstráni z tokenu, čím vzniká čisté slovo. Ďalej sa pre slovo identifikuje, či je celé zložené z veľkých písmen, alebo má iba prvé veľké písmeno. Tieto údaje sa taktiež zapamätajú a použijú sa neskôr pri vrátení cieľového tvaru tokenu po preklade.

Medzi ďalšie identifikačné príznaky v tejto štruktúre patrí:

- *unknown* - či k slovo existuje aspoň jeden záznam v slovníku prekladových párov
- *substituted* – či slovo stavu *unknown* bolo následne nahradené iným slovom podľa jazykového modelu
- *checked* – či bolo slovo kontrolované za účelom substitúcie

Jazykový model a prekladové slovníky sú tvorené zo slov, ktoré sú zložené iba z malých písmen abecedy a obsahujú iba alfabetické znaky. Toto zjednodušenie je možné použiť medzi jazykmi slovenčina a čeština, nakoľko majú takmer identickú stavbu viet:

- ak slovo začína na veľké písmeno v zdrojovom jazyku (začiatok vety, mená osôb, pomenovania, ...), tak bude slovo začínať na veľké písmeno aj v cieľovom jazyku
- ak je slovo v zdrojovom jazyky zložené iba z veľkých písmen (skratky, ...), tak aj v cieľovom jazyku bude zložené celé z veľkých písmen
- ak slovo obsahuje nealfabetický prefix v jednom jazyku (pomlčka, úvodzovky, ...), tak rovnaký prefix bude obsahovať aj preklad v cieľovom jazyku
- ak slovo obsahuje nealfabetický sufix v jednom jazyku (pomlčka, úvodzovky, interpunkčné znamienka, ...), tak rovnaký sufix bude obsahovať aj preklad v cieľovom jazyku

Uvedené pravidlá môžu mať nejaké výnimky, ale pravdepodobnosť výskytu je tak malá, že tieto javy môžeme zanedbať bez viditeľného zhoršenia kvality prekladu.

Následne sa pre identifikované „čisté slovo" vyhľadajú v slovníku prekladových párov všetky dostupné preklady, čím nám vznikne množina nejednoznačností. Všetky identifikované nejednoznačnosti sa uložia pre prislúchajúce slovo k zdrojovému slovu v objekte *SuitableWords*.

Každá nejednoznačnosť je obalená do samostatného objektu ***VocabularyWord***. Tento objekt slúži najmä uskladnenie atribútov nejednoznačnosti a taktiež informácií o kvalite nejednoznačnosti a mapovaní do databázy. Zoznam atribútov:

- *id* – ukazovateľ na unigram v cieľovom jazykovom modeli
- *source* – zdrojový tvar prekladaného slova
- *target* – cieľový tvar prekladu slova, nejednoznačnosti
- *prob* – pravdepodobnosť pre danú nejednoznačnosť, v závislosti od použitia uložená pravdepodobnosť je viacerých typov (normálna, logaritmická)
- *count* – počet výskytov pri trénovaní, prípadne v jazykovom modeli

Nadstavbovou dátovou štruktúrou nad *VocabularyWord* je ***WordPairVector***, ktorý umožňuje efektívne uskladnenie viacerých prekladových párov podľa rovnakého zdrojového slova do separátnych množín, podobne ako množina nejednoznačností. Táto dátová štruktúra sa najmä využíva pri trénovaní nových prekladových párov [Kapitola 5.4.8], ako aj pri preklade s použitím doplňujúceho natrénovaného slovníka.

Štruktúra *WordPairVector* zabezpečuje, aby každý jazykový pár sa v nej nachádzal práve jeden krát. Pri vkladaní jazykových párov automaticky aktualizuje štatistiky početnosti a dokáže identifikovať nové pravdepodobné jazykové páry.

Po naplnení *SuitableWords* nájdenými nejednoznačnosťami *VocabularyWord* sú takého *SuitableWords* zostavené spätne do postupnosti, v ktorej sa realizuje vyhľadanie výsledného tvaru prekladu.

TranslationVector je objekt, ktorého obsahom je postupnosť objektov typu *SuitableWords*. Táto dátová štruktúra zastrešuje nasledovné činnosti:

- vyhľadanie a zostavenie výslednej formy prekladu

- na základe jazykového modelu identifikuje najpravdepodobnejšiu nejednoznačnosť, ktorá sa použije ako správny preklad
- v prípade aktivovania funkcie nahradzovania neznámych slov [Kapitola 5.4.5] zabezpečí nahradenie neznámeho slova za najpravdepodobnejšie náhradné slovo na základe jazykového modelu
- počíta štatistiky súhrnného počtu neznámych a substituovaných slov

Trieda ***SQL*** zabezpečuje prácu s databázou a uchováva nastavenia programu zo súboru *vt.properties* [Kapitola 5.4.4], ktoré sa používajú pri realizácii prekladu.

5.4.2 Zložená n-gramová pravdepodobnosť

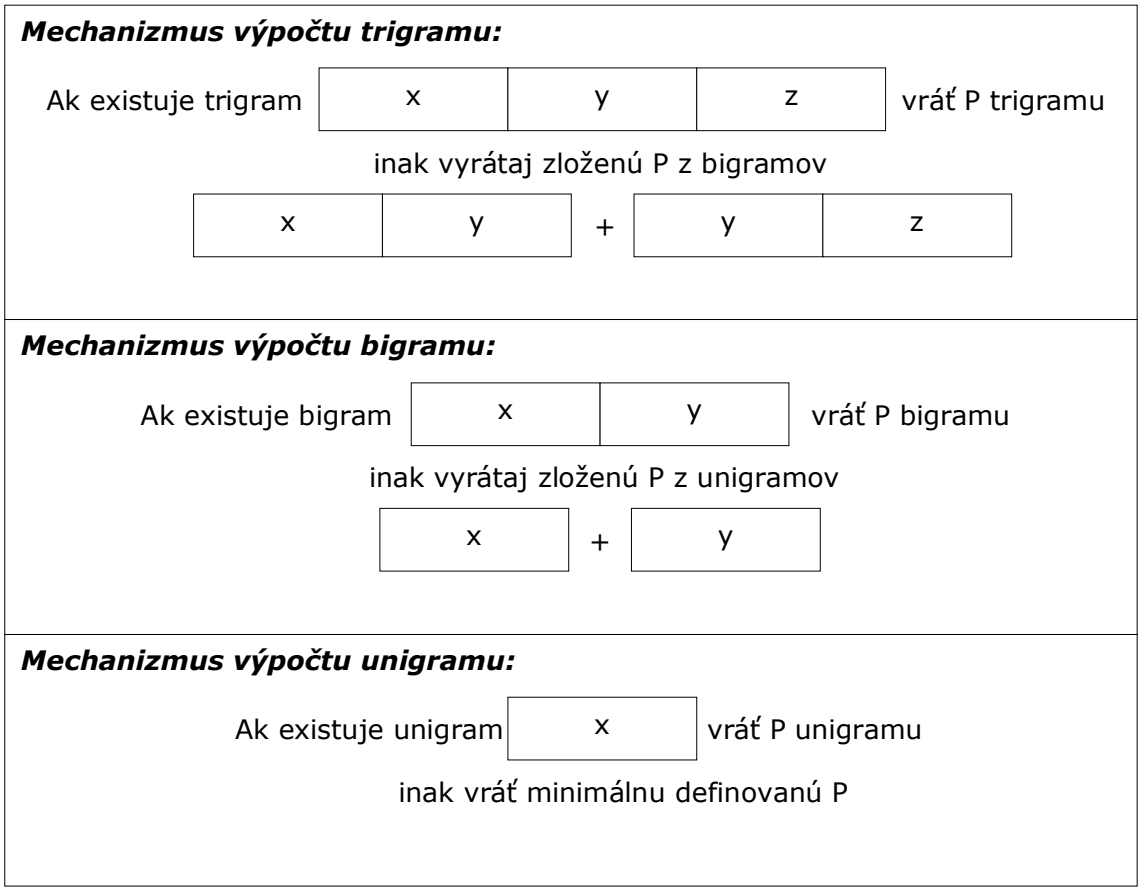
Keďže v jazykovom modeli sme obmedzený na prítomnosť n-gramov veľkosti najviac n=3 a my potrebujeme rátať pravdepodobnosť dlhšieho úseku, výslednú pravdepodobnosť získame ako zloženú pravdepodobnosť podľa nasledujúcej schémy:

P(Nesneží a ty si na lyžiarskom zájazde) ≈
b(Nesneží | koniec_vety začiatok_vety) *
b(a | začiatok_vety Nesneží) *
b(ty | Nesneží a) *
...
b(zájazde | na lyžiarskom) *
b(koniec_vety| lyžiarskom zájazde) *
b(začiatok_vety | zájazde koniec_vety)

kde *b(z | x y)* je pravdepodobnosť, že slovo *z* nasleduje za slovami *x y*. Tieto pravdepodobnosti sa nachádzajú v jazykovom modeli získaného z korpusu.

N-gramový model avšak priraďuje vetám, ktoré nikdy nevidel, pravdepodobnosť 0. Neznamená to, že ak v jazykovom modeli nemáme daný n-gram, že tento n-gram nutne neexistuje, a preto v tomto prípade nahrádzame pravdepodobnosť n-gramu zloženou pravdepodobnosťou (n-1)-gramov:

b(z | x y) >≈ b(y | x) * b(z | y), prípadne b(y | x) >≈ b(x) * b(y)



Obrázok 5.3 Spôsob rekurzívneho výpočtu n-gramovej pravdepodobnosti pomocou menších jednotiek

V tejto forme vyjadrenia pravdepodobnosti máme zároveň aj zahrnutú penalizáciu za neexistenciu daného n-gramu. Z testov totiž vyplynulo, že n-gram zložený z jeho (n-1)-gramov má zvyčajne horšiu pravdepodobnosť, ako keby existoval daný n-gram priamo v modeli. Toto nám zároveň umožňuje zvýhodnenie tých cieľových slov, pre ktoré existujú potrebné trigramy priamo v modeli.

Mechanizmus vyhodnotenia pravdepodobnosti trigramov je realizovaný transparentne s využitým bigramov a unigramov v prípade potreby, pričom ak trigram existuje v modeli, tak sa uprednostní práve jeho pravdepodobnosť.

Po priradení množiny nejednoznačností každému slovu sa všetky trigramy, bigramy a unigramy hromadne vložia do dočasnej cache. Tento spôsob bol zvolený za účelom minimalizácie dotazov do databázy. Pre každý typ n-gramu sa za do databázy pošle iba jeden dotaz. Taktiež dočasná cache umožňuje efektívne prehľadávanie stromu možností bez potreby viacnásobného dotazovania na rovnaký typ n-gramu.

úroveň	slovo[id]	= pravdepodobnosť
+	veriť[1446215]	= -10.59219779861247
++	že[34494]	= -11.465902692741114
+++	sa[12119]	= -13.164789337428264
+	ze[12991]	= -11.231846295071357
++	sa[12119]	= -14.250645514631579
+++	to[12275]	= -16.908393691231446
+	so[12153]	= -6.689398869945281
++	to[12275]	= -15.827995112514179
+	ten[729666]	= -7.672373125577719
+	<s>[380428]	= -2.658555331967836
++	<s>[380428]	= -10.033057810045296
+++	musím[490281]	= -17.601540871791393
++	musím[490281]	= -10.741699167582079
+++	veriť[1446215]	= -17.196075763683226
	...	
+	musím[490281]	= -9.499200798243209
++	veriť[1446215]	= -16.37453881888225
+++	že[34494]	= -16.908393691231446
+	to[12275]	= -5.14411144522264
++	zlepší[1026833]	= -14.582779349654194
+++	<s>[380428]	= -15.036591514329855
+	zlepší[1026833]	= -11.942433109916658
++	<s>[380428]	= -13.474116725132582
+	uveriť[932868]	= -11.734967371321677
++	že[34494]	= -12.573955375188312
+++	sa[12119]	= -14.65710189262495

V tomto príklade úroveň predstavuje n-gram, ktorý si zostavíme spätne zo slov v nadradenej úrovni. napr. P(„<s> musím veriť“) = -17.196075763683226

Príklad 5.3 Výpis z cache možných trigramov na prehľadávanie stromu nejednoznačností pre zdrojovú vetu: *Musím věřit že se to zlepší.*

Pravdepodobnosti v jazykovom modeli sú v databáze uložené v logaritmickom tvare [Kapitola 5.2]. Následne sa pri výpočte zloženej pravdepodobnosti namiesto násobenia jednotlivých pravdepodobností používa logaritmická aritmetika [Knight 1999b]. Nech P(f) je výsledkom viacerých faktorov f1, f2, f3, ... potom:

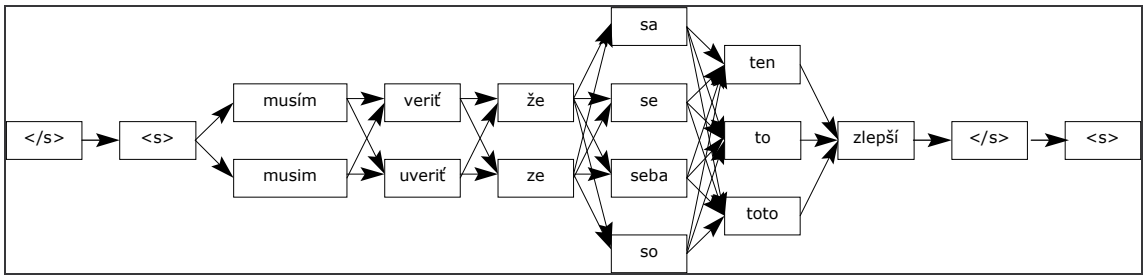
$$log(P(f))=log(f1 * f2 * f3 *...* fn)=log(f1) + log(f2) + log(f3) +...+ log(fn)$$

5.4.3 Mechanizmus výberu cieľového slova prekladu

Ako vstup do prekladu dostávame teda vetu aj s patričnými kontextovými značkami. Takýto reťazec sa následne rozdelí na tokeny. Ako tokenizačné znaky sú použité medzera, tabulátor a podobné „white space“ znaky.

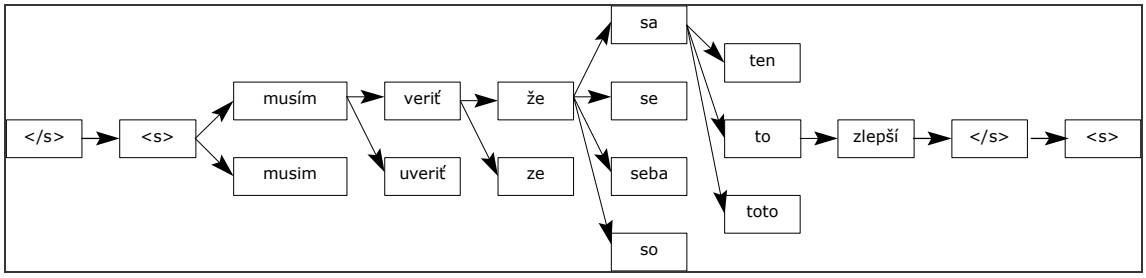
Každý token sa umiestni do štruktúry *SuitableWords*, ktorá transparentne zabezpečí obalenie tokenu funkciami a značkami na ďalšiu identifikáciu. Zároveň táto štruktúra očistí token od nepísmenových znakov (úvodzovky, pomlčky, interpunkčné znamienka, ...) a identifikuje atribúty uppercase. Týmto nám vznikne čisté lowercase slovo, na základe ktorého sa následne vyberajú dáta s databázy.

Pre každé očistené slovo sa identifikuje z prekladového slovníka množina nejednoznačností, tieto sa uložia ako samostatné objekty *VocabularyWord* do príslušného *SuitableWords*. V prípade neexistencie prekladového páru v slovníku je ako nejednoznačnosť poskytnuté zdrojové slovo s tým, že sa označí ako *unknown*. Systém umožňuje následne tieto neznáme slová v preklade zvýrazniť na uppercase. Týmto sa zostrojí prehľadávací strom prípustných kombinácií slov pri preklade.



Obrázok 5.4 Získaný prehľadávací strom pre zdrojovú vetu: *Musím věřit, že se to zlepší*

Prvotná implementácia obsahovala greedy algoritmu, ktorý vyberal cieľový preklad slova na pozícii n na základe predtým vybraných slov na pozíciách (n-2), (n-1). Teda sa v databáze hľadalo 3-gramové okno tvaru: [n-2][n-1][n]



Obrázok 5.5 Prehľadávanie pomocou greedy algoritmu

Používame trigramový model a teda sa vybral lokálny najúspešnejší trigram, trigram s tým cieľovým slovom, ktorý dosahoval v jazykovom modeli najväčšiu úspešnosť.

zdrojové slovo	vybrané n-2	vybrané n-1	testované slová	zvolené	správne
musím	</s>	<s>	musím, musim	musím	musím
věřit	<s>	musím	veriť, uveriť	veriť	veriť
že	musím	veriť	že, ze	že	že
se	veriť	že	sa, se, seba, so	sa	sa
to	že	sa	ten, to, toto	to	to
zlepší	sa	to	zlepší	zlepší	zlepší

Tabuľka 5.3 Prehľadávané 3-gramové okná pri greedy algoritme

Ukázalo sa, že táto forma implementácie je, čo sa týka kvality, málo úspešná. Ak sa raz vybralo slovo, ktoré síce malo najväčšiu úspešnosť v danom trigrame podľa jazykového modelu, prípadne sa daný vhodný trigram v jazykovom modeli nenachádzal a preto bol vybraný iný trigram, ktorý sa síce v jazykovom modeli nachádzal, ale v skutočnosti bol menej pravdepodobný, tak takto zle vybrané slovo následne figurovalo aj pri výbere ďalších dvoch slov, nakoľko trigramové okno má veľkosť tri. Následne tieto testované trigramy s takto zle zvoleným slovom mali opäť malú pravdepodobnosť výskytu a tým aj úspešnosť v jazykovom modeli s aktuálne testovaným slovom, čo spôsobovalo, že sa opäť vybralo nesprávne slovo na ďalšej pozícii. Týmto štýlom sa chyba ťahala už od začiatku.

zdrojové slovo	vybrané n-2	vybrané n-1	testované slová	zvolené	správne
musím	</s>	<s>	musím, musim	musim	musím
věřit	<s>	musim	veriť, uveriť	uveriť	veriť
že	musim	uveriť	že, ze	ze	že
se	uveriť	ze	sa, se, seba, so	so	sa
to	ze	so	ten, to, toto	ten	to
zlepší	so	ten	zlepší	zlepší	zlepší

Tabuľka 5.4 Prehľadávané 3-gramové okná pri greedy algoritme s prenášanou chybou

Finálne implementovaný algoritmus hľadá najlepšie riešenie vzhľadom na dlhší úsek, jednotlivé celé vety, prípadne podradené vety, z ktorých je zložené súvetie. Kontextové značky, ktoré sú vždy v páre „koniec začiatok" vety nám umožňujú bezproblémový prechod medzi jednotlivými časťami, nakoľko tieto kontextové značky nemajú množinu nejednoznačností. Za účelom minimalizácie rizika prekladu príliš dlhej vety, v ktorej vďaka rôznym množinám nejednoznačností nám môže vzniknúť príliš veľký prehľadávací strom, má systém implementovanú voliteľnú dĺžku maximálne prekladaného úseku naraz. Týmto sa zabezpečí prípustná veľkosť prehľadávacieho stromu, na základe požadovaných rýchlostných obmedzení. Zvolené obmedzenie pri dostatočne veľkých úsekoch (napr. 10) nemá

výrazný vplyv na kvalitu, tak ako to mal systém v pôvodnej forme implementácie (greedy algoritmus = veľkosť bloku 1).

Pre každú cestu v strome možností sa ráta zložená pravdepodobnosť úspešnosti danej vety v jazykovom modeli [Kapitola 5.4.2]. Ako preklad je zvolená veta, ktorá túto pravdepodobnosť maximalizuje.

Hľadanie správneho prekladu je realizované v triede *TranslationVector*. Prehľadávanie sa uskutočňuje smerom do hĺbky nasledovne:

```
bestScore = 0;
item = postupnosť nejednoznačností;
start = identifikuj začiatok partície;
end = identifikuj nasledovný koniec partície;
while (end < size(items))
{
        depthSearch(item[1], item[2], item[3], 0);
        start=end+1;
        end=nasledujúci koniec partície;
}
return zapamätaná najlepšia forma nájdeného prekladu;
```

Algoritmus 5.1 Inicializačná procedúra prehľadávania stromu nejednoznačností

V prehľadávajúcej procedúre maximalizujeme skóre, teda dosiahnutú zloženú pravdepodobnosť od začiatku partície po koniec partície. Keďže pravdepodobnosti máme v normovanom logaritmickom tvare, všetko sú to záporné čísla a pri použití logaritmickej aritmetiky [Kapitola 5.4.2] tieto čísla sčítavame. Teda maximalizujeme:

$$score = \sum_{3}^{n} ngram(item[n-2], item[n-1], item[n])$$

Keďže sa pohybujeme v záporných číslach, najviac pravdepodobný preklad podľa jazykového modulu = *argmax score*

```
procedure depthSearch(item[n-2], item[n-1], item[n], score)
{
        //nemá zmysel ďalej hľadať ak napr. score=-100 a bestScore=-80
        if (score < bestScore) then exit;

        //sme na konci postupnosti prehľadávania
        if (n = size(item)) then
        {
                //našli sme lepší preklad podľa jazykového modelu
                if (score > bestScore) then
                {
                        bestScore = score;
                        zapamätaj si aktuálnu formu prekladu partície;
                }
                exit;
        }

        //ak nie sme na konci pokračujeme v prehľadaní dostupných možností
        for each nejednoznačnosť v item[n] do
        {
                newScore = score + P(item[n-2], item[n-1], item[n]);
                depthSearch(item[n-1], item[n], item[n+1], newScore);
        }
}
```

Algoritmus 5.2 Prehľadávanie stromu možností prekladu

Po prehľadaní stromu nejednoznačností sa nájdené zapamätané najlepšie riešenie poskytne ako preklad na výstup.

Zdrojová veta:

Musím věřit, že se to zlepší.

Postup prehľadania stromu možností:

Musím veriť, že sa ten zlepší. -99.37909657031787

Musím veriť, že sa to zlepší. -89.42899236828347

Cieľový preklad:

Musím veriť, že sa to zlepší.

Referenčný preklad:

Musím veriť, že sa to zlepší.

Príklad 5.4 Postupnosť hľadania najpravdepodobnejšieho prekladu

5.4.4 Parametre ovplyvňujúce beh strojového prekladu

Pri implementácii boli vyskúšané rôzne techniky prekladu a heuristiky ovplyvňujúce výsledok jednotlivých častí programu podieľajúcich sa pri preklade.

Nastavenia týchto rutín a heuristík ovplyvňujú výslednú kvalitu prekladu, a taktiež rozhodujú o smere vykonávania prekladu.

Jednotlivé nastavenia sú hromadne konfigurovateľné pomocou konfiguračného súboru *vt.properties*, ktorý sa nachádza v programovej úrovni aplikácie. V konfiguračnom súbore sú obsiahnuté nasledovné nastavenia:

Nastavenia databázy:	
vt.jdbc.url=jdbc:odbc:vocatrans	Zdrojová databáza
vt.jdbc.user=sa	Meno užívateľa, ktorý má prístup na DB
vt.jdbc.password={heslo}	Heslo pre používateľa DB
Nastavenia prekladu:	
vt.language.source=[sk|f]	Zdrojový jazyk prekladu
vt.language.target=[f|sk]	Cieľový jazyk prekladu
tq.uppercase.unknown=true	Nepreložené slová veľkými písmenami
tq.reverse.translation=true	Hľadaj preklad/prekladaj odzadu
tq.check.equal.sentence.length=true	Pri vyhodnotení kvality hodnoť iba vety s rovnakým počtom slov
tq.bilingual.word_pairs.use=true	Použi natrénovaný externý slovník prekladových párov
tq.bilingual.word_pairs.file={súbor}	Cesta a názov súboru, ktorý obsahuje natrénovaný slovník
tq.bilingual.switch_load=true	Prehodenie smeru natrénovaného slovníka pri preklade opačným smerom, ako je vytvorený natrénovaný slovník
vt.tag.context.start=<s>	Kontextová značka začiatku vety
vt.tag.context.end=</s>	Kontextová značka konca vety
tq.max.partition.length=12	Pri preklade použi stromy možností na prehľadávanie maximálne uvedenej hĺbky
vt.min_prob_value=-20	Hodnota pravdepodobnosti nenájdeného n-gramu v jazykovom modeli
Preklad neznámych slov:	
tq.substitute.unknown=true	Nahraď nepreložené slová najvhodnejšími podľa jazykového modelu.
tq.use.ldistance=true	Kandidát na nahradenie sa musí istou mierou podobať na zdrojové slovo
tq.use.word_length_discrimination=true	Kandidát na nahradenie musí mať dĺžku v pomere k zdrojovému slovu
vt.query.limit=30	Vyber z databázy maximálne zvolený počet najvhodnejších slov na nahradenie
Nastavenia pre utility:	
vt.result.dir={priečinok}	Cesta na ukladanie výsledkov
vt.train.iter_count=5	Počet iterácii pre trénovaciu utility pre identifikáciu nových prekladových párov
Ladiace nastavenia:	

vt.debug.sql=true	Zobraz dotazy posielané do DB
vt.debug.count.size=true	Zobraz množstvo nejednoznačností pre slovo
vt.debug.word.choice=true	Zobraz nejednoznačnosti pri výbere cieľového slova
vt.debug.word.choice.dump=true	Zobraz prípustné nejednoznačnosti pre zdrojové slovo
vt.debug.substitute=true	Zobraz proces nahradzovania neznámych slov
vt.debug.aligner=true	Zobraz informácie o úspešnosti spárovania zdrojovej a cieľovej vety pri hodnotení kvality
vt.debug.trace=true	Zobraz použitie jednotlivých metód jadra systému
vt.debug.aligner.log=true	Detailnejší prehľad procesu výberu cieľového slova do súboru
vt.align.interactive=true	Pri trénovaní sa pýtaj na korektný tvar pre doplnenie slovníka prekladových párov

5.4.5 Substitúcie neznámych slov

Medzi ďalšie problémy pri preklade patrí aj otázka, čo robiť so slovami, pre ktoré nemáme v slovníku prekladových párov ani jeden záznam. Medzi elementárne spôsoby riešenia tejto situácie patrí ponechanie daného slova v nepreloženom tvare, teda cieľové slovo bude zhodné so zdrojovým.

Systém štandardne používa tento spôsob zaobchádzania s nepreloženými slovami, ale počas implementácie bol skúmaný aj nasledovný spôsob:

Pre každé nepreložené slovo sa v jazykovom modeli vyhľadá slovo, ktoré keď sa dosadí do trigramu spolu s okolitými slovami, tak pravdepodobnosť takto vzniknutého trigramu bude najvyššia.

Aby použité slovo bolo v preklade čo najvhodnejšie systém vyhľadáva kandidátov na substitúciu tak, aby okolie kandidáta malo čo najviac preložených slov a teda aby vzniknutý trigram obsahoval pokiaľ možno všetky zvyšné slová z cieľového jazyka. Prítomnosť iba slov z cieľového jazyka zabezpečí, že takýto trigram sa bude skôr nachádzať v jazykovom modeli, ako keby niektoré slová v trigramovej vzorke obsahovali nepreložené slovo v zdrojovom jazyku. Jazykový model, v ktorom vyhľadávane, totiž obsahuje trigramy v cieľovom jazyku:

```
procedure substituteUnknown()
{
        for i=4 downto 1 do
        {
                while existuje nepreložené slovo so známym okolím veľkosti i do
                {
                        nájdi možné cieľové slová s neznámym slovom na pozícii 1;
                        nájdi možné cieľové slová s neznámym slovom na pozícii 2;
                        nájdi možné cieľové slová s neznámym slovom na pozícii 3;
                        ohodnoť vyhľadané možnosti pravdepodobnosťou vhodnosti;
                        nahraď testované nepreložené slovo najpravdepodobnejším
                        cieľovým slovom podľa jazykového modelu cieľového jazyka;
                }
        }
}
```

Algoritmus 5.3 Substitúcia nepreložených slov podľa jazykového modelu

Metódu výberu najpravdepodobnejšieho slova pri substitúcii je možné konfigurovať [Kapitola 5.4.4] na použitie heuristík výberu vhodného tvaru nielen podľa pravdepodobnosti, ale aj podľa nasledujúcich charakteristík:

- nepoužiť dodatočné heuristiky
- pomer dĺžky zdrojového a možného cieľového slova nesmie byť menší ako 50%
- Levenshteinova vzdialenosť [Gilleland] zdrojového a možného cieľového slova v pomere k súčtu dĺžok týchto slov nesmie byť menšia ako 25%
- použitie oboch heuristík

Totiž pri hľadaní najpravdepodobnejšieho cieľového slova na nahradenie sa stávalo, že dané slovo sa nahradilo najmä spojkou „a", nakoľko táto spojka s vysokou pravdepodobnosťou figuruje medzi akýmikoľvek slovami.

Heuristika pomeru dĺžky slov nám čiastočne zabezpečí odstránenie takýchto situácii. Zvolená jazyková kombinácia (čeština-slovenčina) na dostatočne podobné slová, aby sa vošli do daného pomeru.

Levenshteinova vzdialenosť, často nazývaná aj „edit distance" nám zase hovorí, koľko transformácií (zmena znaku, vloženie, vymazanie) je minimálne potrebných, aby sme zdrojový reťazec transformovali na cieľový. Uvedená metrika nám umožňuje vyberať slová, ktoré sú si do istej miery podobné. Opäť v našej jazykovej kombinácii sú väčšinou jednotlivé prekladové páry naozaj dosť podobné. Rozdiel medzi zdrojovým a cieľovým slovom býva v lokálnych transformáciách, prípadne zmene koncoviek.

Avšak nezávisle na použitých dodatočných heuristikách, výsledný preklad s použitím substitúcie neznámych slov vždy dosiahol v porovnaní s referenčným prekladom horšie výsledky ako preklad, kde sme neznáme slová nechali v nepreloženom tvare. Často sa totiž stávalo, že slovo nebolo preložené, pretože jeho preklad bol totožný v oboch jazykoch a v slovníku prekladových párov chýbal preň z tohoto dôvodu jazykový pár. Následne keď sa takéto slovo išlo nahradiť, pravdepodobnosť nahradenia rovnakým slovom v cieľovom jazyku bola zvyčajne menšia a teda v porovnaní s referenčným prekladom takýto preklad dosahoval nižšie skóre.

Ďalším problémom aj je, že nahradením neznámych slov za iné podľa jazykového modelu sa zvyčajne pozmenil aj význam textu. Dosadené slovo niekedy úspešne vo vete dávalo aj zmysel, ale po sémantickej rovine veta znamenala niečo iné.

Zdrojová veta:

Musíme všichni jít do Palace.

Preklad po použití slovníka:

Musíme všetci ísť do Palace.

Preklady po použití substitúcie nepreložených slov:

Musíme všetci ísť do Nato.

Musíme všetci ísť na Slovensko.

Príklad 5.5 Ukážky prekladov pri aktívnej funkcii substitúcie neznámych slov

5.4.6 Prekladač „Translate“

Hlavným aktuálne implementovaným front-endom systému ***VocaTrans*** je utilita *Translate*. Tento prekladač má v sebe integrovaných viacero funkcií:

- preklad jedného vstupného súboru
- preklad všetkých súborov obsiahnutých v nejakom priečinku
- logovanie priebehu prekladu
- vyhodnotenie štatistických údajov o uskutočnenom preklade

Utilita *Translate* má teda dva možné formáty vstupných parametrov:

- názov súboru
- názov adresára + prípona súborov, ktoré sa majú prekladať

Logovanie prebieha do súboru s názvom zdrojového súboru s pridanou príponou „log", cieľový preklad sa zaznamenáva do súboru s názvom zdrojového súboru s pridanou príponou „vct".

Do logu sa ukladajú nasledovné údaje:

- dátum a čas začiatku prekladu
- názov vstupného súboru
- postupne zdrojová veta na preklad
- preklad použitím slovníka a výberu správneho tvaru z množiny nejednoznačností na základe jazykového modelu
- ak je aktívna funkcia nahradzovania nepreložených slov [Kapitola 5.4.5], tak výsledný tvar prekladu po substitúcii takýchto slov
- štatistické údaje
- použité nastavenia
- dátum a čas konca prekladu

Sekcia štatistických údajov obsahuje:

- celkový počet slov prekladu
- počet nepreložených slov s pomocou slovníka
- počet substituovaných neznámych slov
- percentuálny pomer nepreložených slov

```
Start: 2005-04-26 23:14:21.379
Source file: test_subtitle_1.plain.sk

TRANSLATION:
[SRC=]Čo je?
[VCB=]Co je?
[SUB=]Co je?
[SRC=]Rozumieš?
[VCB=]Rozumíš?
[SUB=]Rozumíš?
[SRC=]- Čo je? - Čo to robíš?
[VCB=]- Co je? - Co to děláš?
[SUB=]- Co je? - Co to děláš?
[SRC=]Ako ste mu to hovoril?
[VCB=]Jak jste je to říkal?
[SUB=]Jak jste je to říkal?
[SRC=]pohybujem sa, a cítim sa slobodná.
[VCB=]pohybuju se, a cítím se svobodná.
[SUB=]pohybuju se, a cítím se svobodná.
[SRC=]Bože, nie som tak zlá.
[VCB=]Bože, není jsem tenkrát zlá.
```

```
[SUB=]Bože, není jsem tenkrát zlá.
[SRC=]Viem, že by to otec zariadil.
[VCB=]Vím, že by to otče zařídil.
[SUB=]Vím, že by to otče zařídil.
...
[SRC=]Vraj sa tam naozaj dobre tancuje.
[VCB=]Snad se tam jistě dobře tancuje.
[SUB=]Snad se tam jistě dobře korun.
[SRC=]Ja viem.
[VCB=]Já vím.
[SUB=]Já vím.
[SRC=]Čo to deláš, chceš utiesť k cirkusu?
[VCB=]Co to deláš, budeš utiesť k cirkusu?
[SUB=]Co to děláš, budeš mít k cirkusu?

STATISTICS:
Total words: 4879
Total unknown words: 518
Total substituted words: 441
Not translated ratio: 10.616929698708752%
Not translated after substitution: 1.5781922525107603%

USED PROPERTIES:
tq.max.partition.length=12
tq.calculate.prob=true
tq.use.word_length_discrimination=true
tq.use.ldistance=true
tq.bilingual.switch_load=true
tq.bilingual.word_pairs.file=trained_word_pairs.vef
tq.substitute.unknown=true
tq.bilingual.word_pairs.use=true

End: 2005-04-26 23:21:41.256
```

Príklad 5.6 Ukážka informácií obsiahnutých v logu vytváraného pri preklade

Rýchlosť prekladu je závislá od použitej databázy a od výkonu použitej výpočtovej techniky. Počas testovania bola dosahovaná rýchlosť 0,5 MB textu/hod.

5.4.7 Analyzátor kvality bilingválnych textov

Jediným zdrojom bilingválnych textov použitých v tejto práci boli texty z tituliek k filmov, ktoré sú voľne dostupné na internete. Tieto texty majú tú výhodu, že sa dajú efektívne popárovať, nakoľko jednotlivé dialógy obsahujú časové značky zobrazenia titulky vzhľadom k filmu. Štandardne sa na zarovnanie používajú napr. štatistické zarovnávače bilingválnych textov, ale vďaka časovým značkám bolo možné v tomto prípade dosiahnuť oveľa vyššiu úspešnosť zarovnania jednotlivých prislúchajúcich viet, ako vo všeobecných textoch.

Bitexty tohoto charakteru majú ešte jednu výhodu a to, že ak boli navzájom prekladané z jedného jazyka do druhého, tak získané texty mali aj vysoké percento viet, ktoré obsahovali texty medzi jazykmi v takmer identickom usporiadaní slov. Táto vlastnosť bola obzvlášť výhodná pri procese hodnotenia kvality prekladu, nakoľko získaný referenčný preklad sa dal efektívne porovnávať s dosiahnutým prekladom a umožňoval reálnejšie zhodnotenie dosiahnutých výsledkov.

Druhou kategóriu bitextov získaných z tituliek boli texty, kde titulky do patričného cieľového jazyka vznikli ako preklad z angličtiny. V tomto prípade sa často prejavovala fantázia autorov prekladov a cieľové texty medzi jazykmi neboli veľmi vhodné na použitie za účelom zdroja referenčného prekladu.

Za účelom zistenia vhodnosti bilingválneho textu ako zdroja referenčného prekladu bol implementovaný modul *QualityAnalyzer* a jeho front-end *SubtitleQuality*.

Tento modul analyzuje prislúchajúce jazykové vetné páry z bitextu pomocou zisťovania rozdielu dĺžok týchto vetných párov. Použitý jazykový pár slovenčina-čeština sú veľmi blízke jazyky a platí pre ne, že zdrojová veta a preklad sú si čo do dĺžky – počtu slov – zvyčajne dosť podobné. Preto pre zaistenie efektívnej práce učiaceho algoritmu na rozširovanie prekladového slovníka (viď. ďalej) ako aj na efektívne porovnávanie kvality prekladu boli použité texty, kde počet vetných párov rovnakej dĺžky dosahoval v bitexte výskyt aspoň 70%.

Výsledok súhrnnej analýzy 150 bitextov trénovacej vzorky:

```
[ -4]=[0%]
[ -3]=*[1%]
[ -2]=**[2%]
[ -1]=*******[9%]
[  0]=**********************************************************[73%]
[  1]=*****[7%]
[  2]=*[2%]
[  3]=[0%]
```

Výsledok súhrnnej analýzy 36 bitextov testovacej vzorky:

```
[ -3]=[0%]
[ -2]=*[1%]
[ -1]=*******[8%]
[  0]=*****************************************************[81%]
[  1]=***[4%]
[  2]=[1%]
[  3]=[0%]
```

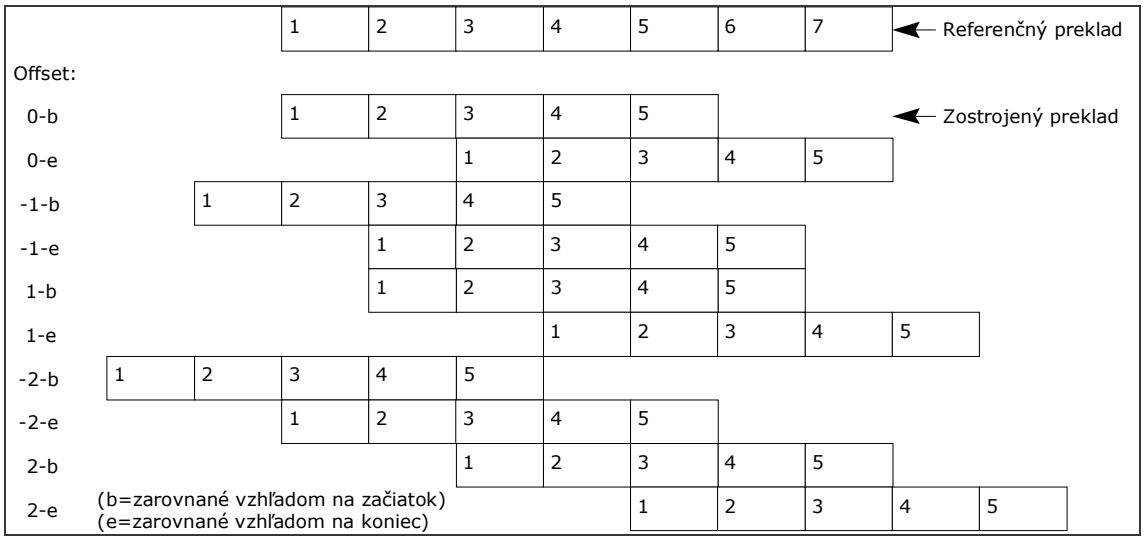
5.4.8 Algoritmus rozširovania prekladového slovníka

Naplnený slovník prekladových párov obsahuje značný počet záznamov, ale stále v ňom chýba väčší počet často používaných slov, najmä takých, ktoré majú zhodný tvar aj v zdrojovom aj v cieľovom jazyku. Pri použití metódy ponechania nepreložených slov v pôvodnej forme takýto preklad dosahuje väčšiu úspešnosť, ale ak sa rozhodneme použiť iné metódy práce s nepreloženými slovami [Kapitola 5.4.5], je pravdepodobné, že kvalita výsledného prekladu bude horšia.

Za účelom rozširovania slovníka prekladových párov o takéto prekladové páry, ako aj o iné slová, ktoré sa nenachádzajú v danom slovníku, bolo potrebné implementovať algoritmus na identifikáciu nových prekladových párov z bilingválnych textov. Na túto činnosť v systéme ***VocaTrans*** slúži modul *Aligner* a jeho front-end pre titulkové bitexty *SubtitleParser*.

Hlavná metóda modulu ***Aligner*** dostáva ako parametre vetu v zdrojovom jazyku a referenčný preklad v cieľovom jazyku. Následne sa pre zdrojovú vetu vyhotoví preklad do cieľového jazyka, a potom sa aktivuje medzi prekladom a referenčným prekladom funkcia párovania, ktorej výsledkom je pole indexov v referenčnom preklade označujúcich pozíciu párového slova v zhotovenom preklade.

Metóda párovania spočíva v postupnom porovnaní slov referenčného prekladu na indexoch prislúchajúcich zhodným pozíciám v zhotovenom preklade s tým, že testované pozície sú posunuté v intervale [-2, 2]. Zarovnanie prekladu pre realizované vzhľadom na začiatok aj koniec vety. Najvyššiu prioritu má nulový posun a najnižšiu prioritu má posun o -2 a 2.



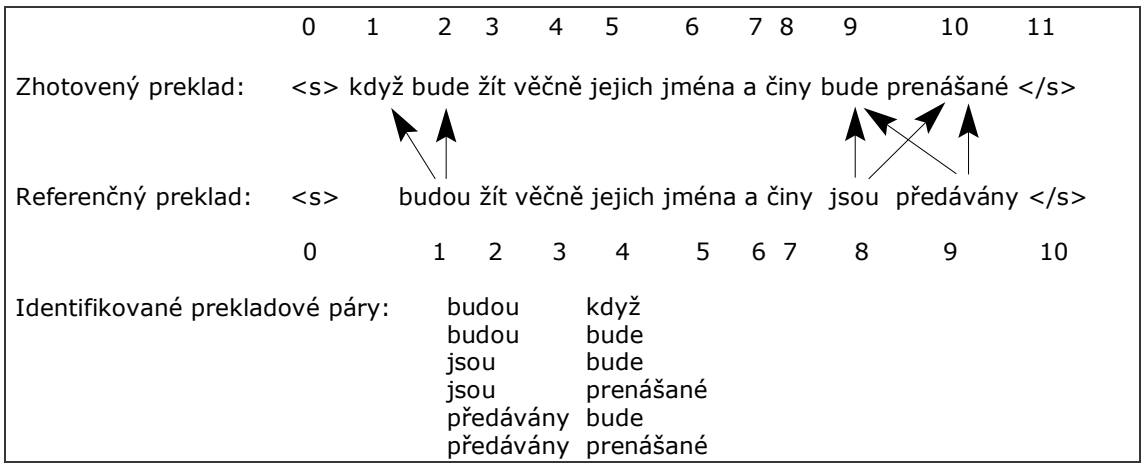
Obrázok 5.6 Schéma postupného testovania pozícií pri párovaní

Za úspešné zarovnanie pozície sa považuje, ak slovo v referenčnom preklade na testovanej pozícii sa zhoduje so slovom v zostrojenom preklade na pozícii danej ofsetom a zarovnaním vzhľadom na začiatok alebo koniec referenčného prekladu.

Zdrojová veta:
{0}=<s> {1}=keď {2}=budú {3}=žiť {4}=večne {5}=ich {6}=mená {7}=a {8}=činy {9}=budú {10}=prenášané {11}=</s>
Referenčný preklad:
{0}=<s> {1}=budou {2}=žít {3}=věčně {4}=jejich {5}=jména {6}=a {7}=činy {8}=jsou {9}=předávány {10}=</s>
Zhotovený preklad:
{0}=<s> {1}=když {2}=bude {3}=žít {4}=věčně {5}=jejich {6}=jména {7}=a {8}=činy {9}=bude {10}=prenášané {11}=</s>
Zarovnanie prekladov:
{0}=0 {1}=-1 {2}=3 {3}=4 {4}=5 {5}=6 {6}=7 {7}=8 {8}=-1 {9}=-1 {10}=11
Slová na pozíciách 1, 8 a 9 referenčného prekladu neboli identifikované ako pár.
Z týchto pozícií budú následne identifikované nové možné prekladové páry.

Príklad 5.7 Ukážka zhotoveného zarovnania prekladov

V zhotovenom zarovnaní sú nezarovnané pozície identifikované ukazovateľom pozíciu „-1". Z tohoto zarovnania sa následne identifikujú nové možné prekladové páry, pričom sa vychádza z toho, že poradie slov je s veľkou pravdepodobnosťou veľmi blízke v oboch prekladoch, a teda možný jazykový pár pre slová z nezarovnaného intervalu [m, n] sa bude nachádzať na pozíciách [alignment(m), alignment(n)].



Obrázok 5.7 Identifikácia nových prekladových párov pomocou indexov zarovnania

Ak interval [alignment(m), alignment(n)] má veľkosť aspoň jedna, potom sa do slovníka možných prekladových párov pridajú všetky kombinácie, na ktorých by predmetný jazykový pár mohol figurovať [Obrázok 5.7]. Identifikované jazykové páry sa ukladajú vo dátovej štruktúry *WordPairVector* [Kapitola 5.4.1]. V tejto štruktúre sa automaticky inkrementujú počítadlá výskytov identifikovaného prekladového páru počas doby trénovania.

Trénovanie prebieha vo viacerých iteráciách. Na konci trénovacej iterácie je slovník možných jazykových párov štatisticky vyhodnotený a do ďalšej iterácie sú ako slová zo slovníka prekladových párov poskytnuté len vybrané najlepšie identifikované páry. Aby slovo patrilo k relevantným jazykovým párom, musí spĺňať aspoň jedno z nasledovných kritérií:

- úspešnosť výskytu testovaného páru musí dosahovať zastúpenie aspoň 50% v množine prekladových párov s rovnakým zdrojovým slovom
- počet výskytov testovaného páru je maximálny vzhľadom na množinu prekladových párov s rovnakým zdrojovým slovom a existuje aspoň jedno slovo s menším počtom výskytov

zdrojové slovo	možný preklad	počet výskytov	pravdepodobnosť
změní	ludi	5	0.04347826
změní	zmenia	20	0.17391305
změní	vás	5	0.04347826
změní	zmení	70	0.6086956
změní	zmenila	5	0.04347826
změní	myš	5	0.04347826
změní	ktori	5	0.04347826

Tabuľka 5.5 Príklad identifikovaných prekladových párov pre slovo *změní*

Na príklade v tabuľke vyššie vidíme, že kritériá výberu spĺňa iba prekladový pár „změní -> zmení", a to hneď obe. Má najvyšší počet výskytov aj najväčšiu pravdepodobnosť nad 0,5 (čo je úspešnosť nad 50%).

Tabuľky identifikovaných prekladových párov sa po každej iterácii ukladajú na disk do súboru ***.vef***, a vybrané najlepšie do ***.bef***. Uložený iteračný súbor ***.vef*** je možné použiť ďalej pri preklade ako doplňujúci slovník prekladových párov [Kapitola 5.4.4]. Vybrané najlepšie je zasa možné manuálne zrevidovať, a potom prípadne natvrdo pridať do databázového slovníka prekladových párov.

5.4.9 Analyzátor kvality prekladu

Automatizované hodnotenie kvality strojového prekladu je netriviálny problém. V prirodzenej reči je možné tú istú vetu preložiť viacerými spôsobmi, syntax prekladov bude rôzna, a pritom si zachová rovnaký sémantický význam. Medzi hlavné skutočnosti, ktoré spôsobujú rozdielnosť jednotlivých správnych prekladov patrí:

- zmenené poradie slov
- použitie iného slova s rovnakým významom (synonymá)
- vloženie alebo vynechanie pomocných bezvýznamových slov

Ľudské ohodnotenie takýchto prekladov môže byť rovnaké, ale pre výpočtovú techniku, ktorá zatiaľ nedokáže identifikovať sémantiku textov, sú tieto preklady rôzne odlišné. Ďalším nepriaznivým faktorom ľudského hodnotenia je, že extenzívne, drahé a zaberá veľa času, takže opätovné hodnotenia vzoriek, prípadne hodnotenia veľkých vzoriek sú často nerealizovateľné.

Výskum firmy IBM ukázal, že existuje silná korelácia medzi automaticky generovaným hodnotením a ľudským hodnotením kvality prekladu [Papineni 2001]. Metóda je založená na štatistickom vyhodnotení výskytov totožných n-gramov zhotoveného prekladu vzhľadom na referenčný preklad.

Štandardne dostupnými technikami založenými na tomto princípe sú hodnotiace metódy BLEU [Papineni 2001] a NIST [NIST 2002], ktoré ale realizujú kontrolu kvality až s výsledne zhotoveného textu.

Pre potreby tejto práce bol implementovaný podobný mechanizmus, obsiahnutý v utilite *TranslationQuality*, ktorý v jednoduchšej forme ráta úspešnosť výskytu unigramov zhotoveného prekladu vzhľadom na referenčný preklad, pričom zároveň zohľadňuje aj detailnejšie štatistiky úspešnosti implementovaných metód substitúcie neznámych slov [Kapitola 5.4.5].

TranslationQuality očakáva na vstupe adresár s titulkovými bitextami. Podľa nastavení konfigurácie prekladového engine ***VocaTrans*** [Kapitola 5.4.4] zhotovuje postupne preklady zdrojovej vety bitextu, a tento preklad zarovnáva pomocou modulu ***Aligner*** [Kapitola 5.4.8] s referenčným prekladom. Úspešnosť prekladu je vyjadrená ako pomer počtu popárovaných slov medzi prekladmi v závislosti od počtu slov zdrojového textu.

Titulkové bitexty majú nasledovný formát:

- názov súboru končí na „bilingual.txt"
- strieda sa: veta v českom jazyku s prefixom „CZ:"
- prislúchajúca veta v slovenskom jazyku s prefixom „SK:"
- voľný riadok

```
CZ:Musím věřit, že se to zlepší.
SK:Musím veriť, že sa to zlepší.

CZ:Kuba
SK:Kuba

CZ:Co je?
SK:Čo je?

CZ:Rozumíš?
SK:Rozumieš?
```

Príklad 5.8 Ukážka formátu titulkového bitextu používaného na trénovanie a testovanie v systéme VocaTrans

Po spracovaní celej testovacej vzorky sa výsledok analýzy ukladá do adresára určeného v nastaveniach systému. Výstup obsahuje nasledovné údaje:

- dátum a čas začiatku analýzy
- počty databázových položiek prítomných v databáze pri preklade
- tabuľku štatistických údajov samostatne pre každý súbor testovanej vzorky
- priemernú dosiahnutú kvalitu na celej vzorke
- použité nastavenia systému pri analýze
- výpis kvality bitextov použitej vzorky [Kapitola 5.4.7]
- dátum a čas ukončenia analýzy

Tabuľka štatistických údajov pre každú vzorku zobrazuje:

- celkový počet slov zdrojového textu v danej vzorke
- celkový počet spárovaných slov danej vzorky
- úspešnosť správneho prekladu dosiahnutého na danej vzorke
- celkový počet slov, ktoré mohli byť spárované do správneho prekladu
- úspešnosť, aká mohla byť dosiahnutá, ak by boli vybrané správne slová
- celkový počet slov v zhotovenom preklade na danej vzorke

- počet nepreložených slov zanechaných v pôvodnom tvare
- počet substituovaných neznámych slov
- percentuálna miera preložených slov pomocou slovníka + substitúcie
- percentuálna miera množstva zanechaných slov bez prekladu
- percentuálna miera počtu neznámych slov nahradených substitúciou
- názov súboru danej testovanej vzorky

Za účelom rovnakého ohodnotenia textov, ktoré vznikli ako výstup z iných prekladových systémoch bola implementovaná utilita ***TranslationQualityEx***. Táto utilita používa rovnaký mechanizmus párovania zhotoveného prekladu s referenčným prekladom, ale nie je závislá na použití bilingválnych titulkových textov.

Na porovnanie sa používajú jednotlivé zdrojové texty ako aj preklady v samostatných súboroch, pričom prislúchajúce vety sa nachádzajú v daných súboroch na rovnakých číslach riadkov. *TranslationQualityEx* očakáva na vstupe štyri parametre:

- adresár s testovanými súbormi
- príponu súboru so zdrojovým textom
- príponu súboru s referenčným prekladom
- príponu súboru s cieľovým textom

```
TranslationQualityEx ./compare/ .sk .cz .sk.vct
```

Príklad 5.9 Ukážka parametrov volania programu TranslationQualityEx

TranslationQualityEx vo výsledku miesto informácií o databáze zobrazuje adresár s testovanými súbormi. Taktiež údaje o počte neznámych a substituovaných slov v tomto prípade nehrajú úlohu, nakoľko predmetné informácie nie je možné z už zhotovených prekladov získať.

5.5 Zhrnutie

V tejto kapitole sme si predstavili jadro tejto práce a to implementovaný prekladový systém ***VocaTrans***, ktorý kombinuje metódy slovníkového prekladu a štatistického prístupu k prekladu. Predstavili sme si postupnosť činností na vybudovanie rozšíreného prekladového slovníka, spôsob uloženia dát v databáze, spôsob vytvorenia jazykového modelu z národných korpusov a najmä možnosti implementovaného systému *VocaTrans*: použité dátové štruktúry, mechanizmus prekladu, možnosti konfigurácie, front-end prekladača, učiaci algoritmus na identifikovanie nových prekladových párov, možnosti práce s nepreloženými slovami a prostriedky použité na vyhodnotenie kvality prekladu.

6 Vyhodnotenie

Na účely vyhodnotenia bol implementovaný vlastný nástroj na určenie kvality, ktorý meria kvalitu tým, že ráta percentuálnu zhodu unigramov vzhľadom na referenčný preklad, ktorý sa nachádza v bilingválnych textoch [Kapitola 5.4.9]. Kvalita dosiahnutého prekladu teda zobrazuje dolnú hranicu úspešnosti, nakoľko pri preklade mohlo byť vybrané iné slovo, ktoré významovo zodpovedá požadovanému prekladu, ale v referenčnom preklade sa namiesto neho nachádzalo iné.

Avšak vzhľadom na techniku a možnosti, ktoré boli k dispozícii pre ostatné formy prekladu, možno tvrdiť, že reálna kvalita prekladu nie je signifikantne rozdielna od zistenej dolnej hranice. Objektivita výsledkov je zaručená tým, že všetky systémy boli porovnávané rovnakým nástrojom.

Dôležitými charakteristikami pri slovníkovom preklade vzhľadom na možnosť dosiahnutia výsledného prekladu sú najmä:

- či v slovníku existuje prekladový pár pre dané slovo
- následne forma výberu správneho tvaru, ak množina nejednoznačností obsahuje viacej možností prekladu daného slova
- podpora prekladu typu viac-na-viac slov, ktorý v aktuálnej práci nie je podporovaný, čo ale pre veľmi blízke jazyky nemusí nutne znamenať výrazné zhoršenie

6.1 Dáta použité pri meraní

Pri finálnej forme spôsobu prekladu a to použitým prehľadávania najvhodnejšej možnosti vzhľadom na jazykový model bolo v databáze dostupných nasledovný počet položiek (bigramy a trigramy majú cut-off = 1):

tabuľka	CZ -> SK	SK -> CZ
prekladové páry	613371	339652
unigramy	1506749	492949
bigramy	10128300	7392326
trigramy	15323917	8590450

Tabuľka 6.1 Množstvo databázových údajov použitých pri vyhodnotení

K prekladovým párom dostupným zo slovníka bolo následne doplnených ďalších 20000 prekladových párov, ktoré systém identifikoval v rámci

implementovaného učiaceho sa trénovacieho algoritmu [Kapitola 5.4.8] na vzorke 5 MB bilingválnych textov. V prekladovom slovníku väčšina prekladových párov obsahuje najmä mapovania medzi jednotlivými formami rovnakého slova, čo najviac určuje jeho veľkosť a ukázalo sa, že identifikovaných 20000 dotrénovaných jazykových párov zlepšilo kvalitu prekladu, z čoho môžeme vyvodiť, že vo vytvorenom bilingválnom slovníku sa stále nenachádza dosť veľká časť dôležitých prekladových párov.

6.2 Zistené výsledky na celej testovacej vzorke

Na testovacej vzorke 670 KB zdrojového textu boli zistené nasledovné výsledky súhrnnej úspešnosti pre 1-gramy:

použitý program / technika prekladu	CZ -> SK	SK -> CZ
maximálny dosiahnuteľný preklad s použitým slovníkom	86,13%	86,70%
preklad na vetách, kde zdrojová a cieľová majú rovnakú dĺžku	66,18%	73,89%
implementovaná metóda, dotrénovaný slovník, cut-off = 0,1,1	64,82%	70,86%
implementovaná metóda, dotrénovaný slovník, cut-off = 0,0,0	63,48%	68,69%
jazykový model ARPA, 65534 unigramov, cut-off = 0,0,0		65,01%
implementovaná metóda, bez dotrénov. slovníka, cut-off = 0,1,1	62,54%	67,10%
greedy algoritmus, dotrénovaný slovník, cut-off = 0,1,1	58,47%	62,92%
implementovaná metóda, substitúcia, Levenshtein, dĺžka slov	56,00%	59,43%
PC Translator 2004	51,36%	55,78%
GIZA++		23,51%

Tabuľka 6.2 Úspešnosť prekladu rôznymi metódami na celej testovacej vzorke

Implementovaný systém VocaTrans používa na určenie správnej formy prekladu, z množiny nejednoznačností, štatisticky najvhodnejšie slovo vzhľadom na kontext okolia vety, čím sa dosahuje jeho zvýšená úspešnosť vo formovaní najpravdepodobnejšieho tvaru vety, ktorá na základe jazykového modelu presnejšie zobrazuje jazykové charakteristiky daného jazyka do výslednej formy prekladu.

Výraznou mierou na tento výber vplýva aj kvalita poskytnutého korpusu. ARPA korpus bol obmedzený na použitie iba 65 tisíc najviac používaných slov v jazyku, čo sa ale prejavilo na výslednej nižšej kvalite zostavených cieľových prekladov. Tak výrazné obmedzenie slovníka ani nenahradili použité discounting stratégie [Clarkson 1997] zjemnenia okrajových pravdepodobností. Kvalita prekladu dosiahnutého na ARPA jazykovom modeli dosahuje horšiu kvalitu o približne 5%.

PC Translator používa na výber výslednej možnosti z množiny nejednoznačností metódu prvého výskytu, čo mu avšak zabezpečuje iba priemernú

pravdepodobnosť úspechu a pri hranici 50% sa nedá uvažovať o úspešnom preklade.

Na rozdiel od toho, VocaTrans v optimálnom nastavení s vyhľadaním štatisticky najpravdepodobnejšieho riešenia, dosahuje úspešnosť až o 15% lepšiu ako je metóda výberu prvej možnosti, ktorá je použitá v systéme PC Translator.

Na druhej strane podľa zistených výsledkov systém VocaTrans ešte stále identifikoval približne 15% slov v inej forme ako sa nachádzali v referenčnom preklade. Maximálna dosiahnuteľná úspešnosť s použitým slovníkom znamená, že správny tvar prekladu mohol byť pomocou slovníka identifikovaný. Pre zdrojové slovo existoval v databáze prekladový pár, ale systém na základe jazykového modelu nakoniec vybral inú formu. Ako bolo spomenuté, zvolenie inej formy, inej ako v referenčnom preklade, nemusí nutne znamenať, že zvolené slovo nie je správne zo sémantického významu, avšak túto skutočnosť je problematické overiť automatizovanou metódou hodnotenia kvality prekladu.

tabuľka	CZ -> SK	SK -> CZ
prekladové páry	613371	339652
unigramy	1506749	492949
Bigramy	33914568	26302456
trigramy	93103872	51735183

Tabuľka 6.3 Množstvo databázových údajov v jazykovom modeli bez cut-off

V databáze bez cut-off bolo prítomných asi 6 krát viacej trigramov. Z výsledkov zistených na tejto databáze vidno, že pri použití jazykového modelu bez cut-off boli dosiahnuté asi o 2% horšie výsledky ako pri aktívnom cut-off na bigramy a trigramy. Pri hodnotení zloženej pravdepodobnosti [Kapitola 5.4.2] je uprednostňovaná pravdepodobnosť trigramu, pred menšími jednotkami. Keďže v tomto prípade bolo prítomných 6 krát viacej trigramov, úspešnosť nájdenia toho pravého bola vyššia, ale takto získaná pravdepodobnosť zrejme pôsobila horšie ako zložená pravdepodobnosť z menších jednotiek, bigramov, ktoré v zloženej forme zabezpečili trigramu lepší výsledok.

Ďalej z výsledkov vidno, že metóda substitúcie preklad skôr pokazila, ako zlepšila. Pri preklade do slovenčiny je pozorované zhoršenie o 9% ale v opačnom smere až takmer o 12%. Je to spôsobené tým, že pri preklade bola zistená menšia úspešnosť aj štandardne realizovaného prekladu, čím vznikal priestor pre zvýšenú mieru neznámych slov, kandidátov na substitúciu, ale realizovaná substitúcia nezodpovedala požadovanému prekladu. Ak by boli tieto slová ponechané

v pôvodnej forme, blízka príbuznosť jazykov čeština-slovenčina, v ktorej sa nachádza silná množina totožných slov aj po preklade, by zabezpečila vyššiu efektivitu, ako keď bolo takto správne slovo nahradené iným.

Taktiež sa potvrdilo, že vo vytvorenom slovníku sa stále viacero prekladových párov nenachádza. Po pridaní slovníka vytvoreného implementovaným učiacim algoritmom [Kapitola 5.4.8] stúpla kvalita prekladu o 2 až 4%.

Najmenšiu kvalitu dosiahol systém GIZA++. Avšak tento systém nemožno objektívne hodnotiť s ostatnými výsledkami, GIZA++ je čisto štatistický systém, ktorý naviac mal na trénovanie dostupných veľmi málo bitextov (5 MB), čím nebolo možné dosiahnuť, aby systém dôkladne dokázal identifikovať prekladové páry, vytvoriť model prekladu, ktorý by pokrýval a správne vyberal cieľové slová na testovacej vzorke.

6.3 Vybrané ukazovatele pri jednotlivých vzorkách

V tejto kapitole si predstavíme výsledky z detailnejšieho pohľadu podľa dosiahnutej úspešnosti na jednotlivých testovacích vzorkách. V tejto kapitole sú kvôli jednoduchosti použité vždy výsledky pre smer prekladu SK->CZ.

V nasledujúcej tabuľke vidíme dosiahnutú úspešnosť na vybraných vzorkách:

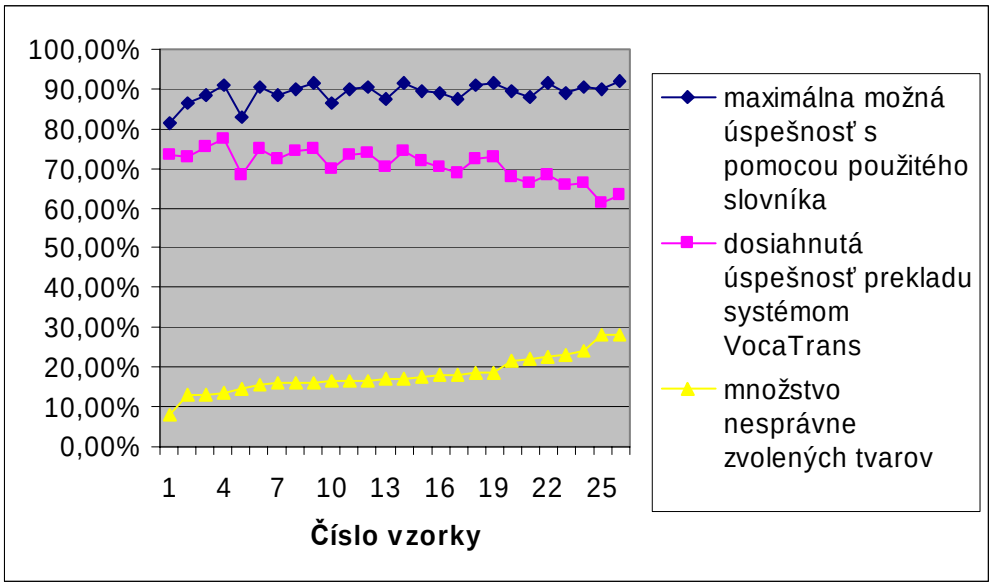
Implemen. metóda, dotrén. slovník, cut-off = 0,1,1	vzorka A	vzorka B	vzorka C
Maximálna úspešnosť z nejakého hľadiska	81,58%	90,73%	91,76%
dosiahnutá úspešnosť prekladu systémom VocaTrans	73,32%	77,38%	63,46%
množstvo nesprávne zvolených tvarov	8,26%	13,35%	28,31%

Tabuľka 6.4 Porovnanie vzoriek s maximálnymi hodnotami

Vzorka A ukazuje preklad, kedy sa podarilo takmer presne odhadnúť všetky formy nejednoznačností na správne. Zostávajúcich 8,26% zvolených foriem prekladu nezodpovedalo správnym formám, ktoré mohli byť vybrané na základe použitého prekladového slovníka.

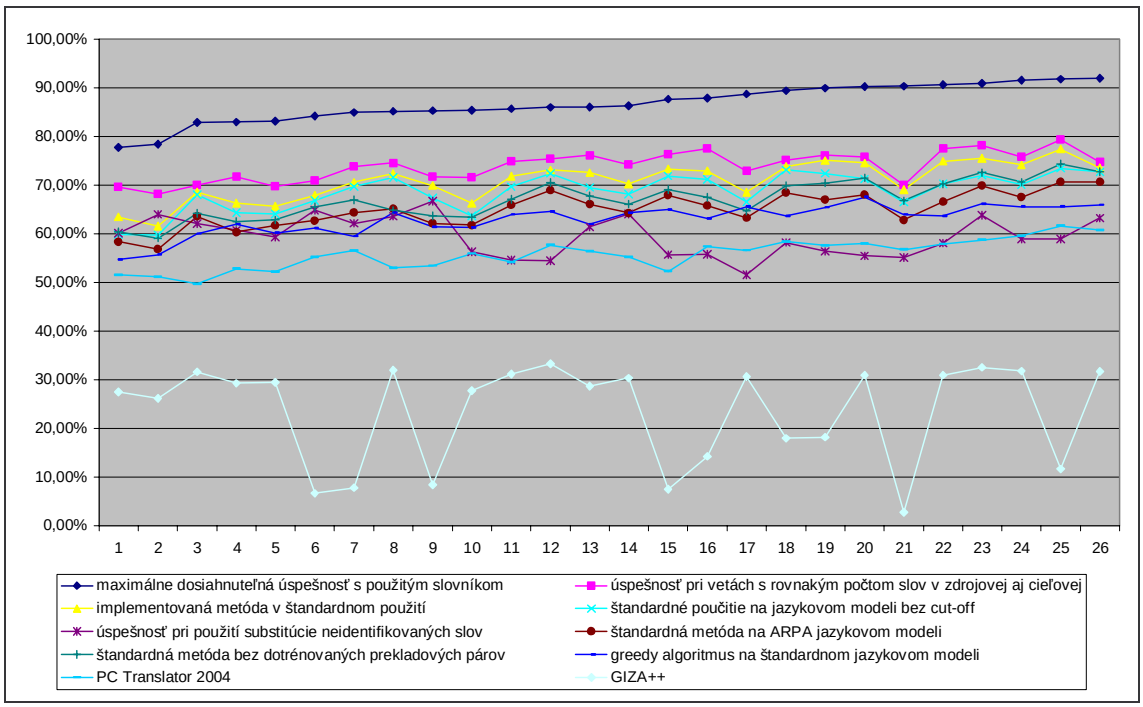
Vzorka B zobrazuje preklad, kedy systém VocaTrans dosiahol najväčšiu úspešnosť prekladu vzhľadom na referenčný preklad.

Vzorka C obsahuje informáciu o preklade, kde na základe slovníka mohlo byť preložených maximálny počet slov, ale úspešnosť výberu správnej formy v tomto prípade nebola veľmi výrazná.



Graf 6.1 Úspešnosť prekladov na vybraných vzorkách

Ako vidno z grafu, množstvo nesprávne určených tvarov slov na testovacej vzorke variuje vzhľadom na referenčný preklad okolo priemernej hodnoty 18,11% z priemernej možnej dosiahnuteľnej kvality prekladu 88,97%.



Graf 6.2 Úspešnosť prekladov na jednotlivých vzorkách

Tento graf nám zase ukazuje úspešnosti dosiahnuté podľa jednotlivých testovaných vzoriek vo všetkých systémoch. Vzorky sú usporiadané podľa

maximálne možnej dosiahnuteľnej kvality pomocou použitého slovníka. Ako vidíme na vzorkách číslo 10 a 21 sa výraznejšie prejavuje zvýšená neúspešnosť prekladu pri všetkých nastaveniach systému VocaTrans, z čoho možno usudzovať, že tieto vzorky pravdepodobne obsahujú referenčný preklad, ktorý nie je veľmi blízky jazykovému modelu. Na druhej strane systém PC Translator na týchto vzorkách takmer nedosahoval odchýlku v kvalite vzhľadom sa rastúcu tendenciu možnej dosiahnuteľnej kvality prekladu, dokonca na vzorke číslo 10 v pomere k systému VocaTrans dosiahol viditeľne lepší výsledok.

6.4 Výsledky merania metódami BLEU a NIST

Na určenie kvality prekladov boli použité aj meracie techniky BLEU [Papineni 2001] a NIST [NIST 2002], ktoré rátajú úspešnosť prekladu vzhľadom na výskyt n-gramových vzoriek. Vo výstupoch si predstavíme výsledky dosiahnuté pre smer prekladu SK->CZ:

```
NIST score = 17.4140  BLEU score = 1.0000 for system "orig"
NIST score = 10.5832  BLEU score = 0.4374 for system "vct"
NIST score = 7.3554   BLEU score = 0.2469 for system "pct"
NIST score = 2.9580   BLEU score = 0.0552 for system "giza"
```

Tabuľka 6.5 Súhrnná úpriemerná spešnosť pre 4-gramy

Vysvetlivky:

- *orig* – pôvodný referenčný preklad
- *pct* – čisto slovníkový systém PC Translator 2004
- *giza* – čisto štatistický systém GIZA++
- *vct* – implementovaný slovníkovo-štatistický systém VocaTrans

Tabuľka 6.5 nám zobrazuje úspešnosť hlavných systémov porovnávaných v tejto práci. Z dosiahnutých výsledkov vidíme, že implementovaný slovníkovo-štatistický systém VocaTrans dosahuje úspešnosť 0,4374 bodu, čo je takmer 2 krát viacej ako čisto slovníkový systém PC Translator.

Pre porovnanie zoberieme do úvahy výsledky dosiahnuté medzi inými veľmi príbuznými jazykmi. V práci *Europarl: Multilingválny korpus pre vyhodnotenie strojového prekladu* [Koehn 2002] sa autor zameral na preklad medzi 11-timi jazykmi štátov európskej únie. Preklad medzi týmito jazykmi bol implementovaný

pomocou systému GIZA++. Dosiahnuté výsledky [Koehn 2002, strana 8] je možné porovnať, nakoľko sú testované rovnakou metódou hodnotenia kvality BLEU [Papineni 2001] ako je sekundárne použitá aj v tejto práci.

Najzaujímavejším jazykovým párom, relevantným k strojovému prekladu veľmi blízkych jazykov, je jazykový pár ***španielčina-portugalčina***. Ako mohlo byť očakávateľné, systém medzi jazykmi španiečina-portugalčina dosiahol najlepšie výsledky. Je to zrejmé kvôli ich silnej príbuznosti.

Za povšimnutie stojí aj dosiahnuté skóre, ktoré pre preklad španielčina-portugalčina dosiahlo 0,3572 bodu. Avšak implementovaný systém VocaTrans dosahuje ešte lepšie skóre: 0,4374 bodu. Keďže systémy aj jazykové páry sú rozdielne, nedá sa určiť či lepší výsledok medzi jazykmi slovenčina-čeština je spôsobený väčšou blízkosťou týchto jazykov, alebo metóda slovníkovo-štatistického prístupu umožňuje vyššiu efektivitu prekladu, ale z tabuľky medzi ostatnými jazykmi vidno, že kvalita dosiahnutá systémom VocaTrans je v priemere 2x lepšia ako medzi akýmikoľvek inými jazykovými párm zo skúmaných 11 jazykových párov.

Na záver uvádzam ešte detailnejší rozpis výsledkov zistených metódami BLEU a NIST:

	1-gram	2-gram	3-gram	4-gram	5-gram	6-gram	7-gram	8-gram	9-gram	
NIST:	10.9260	5.0381	1.2422	0.1815	0.0262	0.0050	0.0021	0.0012	0.0013	"orig"
NIST:	5.1915	1.7544	0.3654	0.0406	0.0035	0.0005	0.0000	0.0000	0.0000	"pct"
NIST:	2.5249	0.3798	0.0474	0.0053	0.0006	0.0001	0.0000	0.0000	0.0000	"giza"
NIST:	7.2210	2.6548	0.6163	0.0818	0.0093	0.0015	0.0005	0.0002	0.0000	"vct"
BLEU:	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	"orig"
BLEU:	0.5451	0.3230	0.1908	0.1107	0.0639	0.0375	0.0219	0.0143	0.0093	"pct"
BLEU:	0.2222	0.0808	0.0344	0.0151	0.0068	0.0031	0.0015	0.0006	0.0001	"giza"
BLEU:	0.6959	0.5088	0.3796	0.2841	0.2140	0.1620	0.1228	0.0938	0.0710	"vct"

Tabuľka 6.6 Individuálne n-gramové skórovanie

	1-gram	2-gram	3-gram	4-gram	5-gram	6-gram	7-gram	8-gram	9-gram	
NIST:	10.9260	15.9641	17.2063	17.3878	17.4140	17.4190	17.4210	17.4222	17.4235	"orig"
NIST:	5.1915	6.9459	7.3113	7.3519	7.3554	7.3559	7.3559	7.3559	7.3559	"pct"
NIST:	2.5249	2.9047	2.9521	2.9575	2.9580	2.9581	2.9581	2.9581	2.9581	"giza"
NIST:	7.2210	9.8758	10.4921	10.5738	10.5832	10.5847	10.5852	10.5854	10.5854	"vct"
BLEU:	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	"orig"
BLEU:	0.5451	0.4196	0.3227	0.2469	0.1885	0.1440	0.1100	0.0853	0.0667	"pct"
BLEU:	0.2222	0.1340	0.0852	0.0552	0.0363	0.0241	0.0162	0.0108	0.0065	"giza"
BLEU:	0.6887	0.5888	0.5069	0.4374	0.3784	0.3279	0.2845	0.2474	0.2151	"vct"

Tabuľka 6.7 Kumulatívne n-gramové skórovanie

7 Záver

Výsledky dosiahnuté implementovaným slovníkovo-štatistickým prekladovým systémom VocaTrans potvrdili hypotézu, že je možné ešte zlepšiť metódu slovníkového prekladu skombinovaním so štatistickými informáciami z jazykoveho modelu cieľového jazyka prekladu.

Ukázali sme, že príbuznosť veľmi blízkych jazykov značne ovplyvňuje kvalitu prekladu k lepšiemu. Taktiež sa ukázalo, že veľmi vhodnou metódou zaobchádzania s nepreloženými slovami je ich zanechanie v pôvodnom nepreloženom tvare. Tak príbuzné jazyky ako slovenčina a čeština obsahujú takýchto slov vysoké percento a ponechanie pôvodného tvaru zabezpečilo až o 10% kvalitnejší preklad, ako keby sme ich zamenili za iné. Metóda nahradzovania za najpravdepodobnejšie lokálne slovo z okolia podľa jazykového modelu sa ukázala za nevhodný spôsob zaobchádzania s takýmito slovami, pre ktoré sme v slovníku prekladových párov nemali žiadny záznam.

Z dosiahnutých výsledkov vyplynulo, že systém slovníkovo-štatistického prístupu dosahuje zreteľne lepšie výsledky ako systémy na čisto slovníkový preklad. Taktiež v porovnaní s iným veľmi blízkym jazykovým párom španielčina-portugalčina sa ukázalo, že čisto štatistický preklad medzi týmito jazykmi dosahuje o 0,08 bodu horší výsledok (podľa metódy BLEU [Papineni 2001]) ako dosiahol implementovaný systém VocaTrans v meraní touto metódou.

Vo všeobecnosti môžeme povedať, že implementovaný slovníkovo-štatistický systém je podstatne úspešnejší ako iné systémy. V čisto slovníkovom prístupe bola pozorovaná úspešnosti okolo 50%, avšak pri takejto kvalite prekladu je preklad veľmi hrubý a nepoužiteľný. Systém VocaTrans dosahoval úspešnosť cez 70%, čo sa už dá považovať za zrozumiteľnejší preklad v porovnaní s dosiahnutým čisto slovníkovým prekladom.

Ukázali sme, že existujú metódy na zlepšovanie existujúcich prístupov. Ďalšími možnosťami zlepšenia kvality sa ukazuje skvalitnenie slovníka, doplnenie absentujúcich prekladových párov a taktiež vyladenie jazykového modelu. Tieto metódy sú zároveň odporúčaniami pre možnosti ďalšieho vývoja.

Ďalšie informácie, vytvorený systém VocaTrans a dáta je možné nájsť na webe: http://www.jimi.sk/vocatrans/

8 Ďalšie možnosti vývoja

Aktuálne systém obsahuje v prekladovom slovníku iba výrazy tvaru jeden-na-jeden. Pre preklad medzi veľmi blízkymi jazykmi, akými sú slovenčina a čeština, to nemusí znamenať výrazný rozdiel, ale pri porovnaní s referenčným prekladom, ktorý obsahuje aj preklady jeden-na-viac, prípadne viac-na-viac sa kvalita javí horšou. Systém v aktuálnej fáze implementácie je čiastočne pripravený na podporu takýchto konštrukcií.

Taktiež sa ukázalo, že v slovníku sa stále nenachádzajú niektoré dôležité jazykové páry. Odporúča sa rozšíriť prekladový slovník buď pomocou revidovaných slovníkov získaných trénovacím algoritmom, prípadne inými metódami.

Ďalšou formou zlepšenia kvality je implementácia rôznych metód vyhladenia pravdepodobností pre n-gramy s veľmi malým počtom výskytov. Boli robené pokusy, kde sa ukazuje, že to, že n-gram má okrajové zastúpenie v korpuse nemusí nutne znamenať, že jeho pravdepodobnosť výskytu je naozaj až taká malá. Medzi najznámejšie techniky vyhladenia (discounting methods [Clarkson 1997]) patria:

- Good-Turing discounting
- Witten Bell discounting
- Absolute discounting
- Linear discounting

Z hľadiska práce s nepreloženými slovami môže byť skúmané nahradenie metódy substitúcie nepreložených slov [Kapitola 5.4.5] za iné slová na základe jazykového modelu za metódu podobnú tej, aká sa používa pri preklade, teda najprv zostavenie väčšieho stromu možných slov na substitúciu a následné zrátanie globálne vhodnejšej kombinácie, nakoľko terajšia metóda odzrkadľuje iba lokálne najúspešnejšie riešenie. V prípade, že sa ako vhodný kandidát v cieľovom jazykovom modeli nájde aj slovo rovnakého tvaru ako zdrojové slovo, tak by bolo uprednostnené.

V aktuálnom stave je implementovaný front-end na prekladanie súborov. Medzi plány patrí vytvorenie webovej služby pre aplikáciu, aby bola dostupná cez internet. A ako ďalší stupeň začlenenia do prostredia internetu by bolo možné vytvoriť online prekladač cudzích webových stránok.

9 Slovník pojmov

Bitext	Abstraktný koncept odkazujúci na dva texty (monolingválne texty), kde jeden je pôvodný text a druhý je prekladom pôvodného textu. Tieto dva texty sa zvyčajne nazývajú zdrojový a cieľový.
Cieľový text	Preklad zdrojového textu.
Cut-off	Odstránenie n-gramov s veľmi malým počtom výskytov. Rapídne zmenší veľkosť databázy a tým aj zrýchli činnosť celého systému. Testy ukázali dokonca lepšie výsledky, ako pri databáze bez cut-off.
Čisté slovo	Slovo, z ktorého boli zo začiatku a z konca odstránené nealfabetické znaky. Databáza výrazov obsahuje kvôli zmenšeniu kombinácií iba mapovania medzi čistými slovami.
EGYPT	Softvérový toolkit pre spracovanie korpusu, trénovanie atď.
GIZA	Softvérová utilita z toolkitu EGYPT, ktorá sa používa na trénovanie modelu prekladu.
GIZA++	Vylepšenie utility GIZA. Program, ktorý vytvára model prekladu, s podporou až do IBM Model 5 a zahrňuje veľa ďalších nových vlastností v porovnaní s utilitou GIZA.
Heuristika	Koncept „heuristiky" znamená zvyčajne vyriešenie problému metódou založenou na použití nejakého elementárneho alebo komplexnejšieho spôsobu hľadania riešenia. Niektoré systémy môžu kombinovať viacero jednoduchších heuristík na vytvorenie riešenia.
Jazykový model	Obsahuje štatistický vzorku jazyka s určenými pravdepodobnosťami pre n-gramy.
Jednotka	V tomto materiály jednotka označuje segment monolinguálneho textu ako slovo, fráza, veta atď., ale tiež korešpondujúci segment v bitexte, t.j. prekladové jednoty.

Koreňové slovo	Základný tvar slova v jazyku, ktorý podporuje ohýbanie slov
Množina nejednoznačností	Množina cieľových slov, ktoré vznikajú pri slovníkovom preklade, keď sa jedno zdrojové slovo môže preložiť na viacero rozdielnych cieľových slov.
Model prekladu	Obsahuje pravdepodobnostné ohodnotenie možných prekladových párov medzi jazykmi.
Nejednoznačnosť	Predstavuje jedno cieľové slovo z neprázdnej množiny rozdielnych cieľových slov, ktorá vzniká pri slovníkovom preklade, keď jedno zdrojové slovo sa môže preložiť na viacero rozdielnych cieľových slov.
N-gram	Vzorka jazyka zostavená s n po sebe idúcich slov.
Paralelný korpus	Vety prirodzeného jazyka a ich preklady so zarovnaním medzi korešpondujúcimi segmentmi v rozdielnych jazykoch.
Prekladový pár	Dvojica výrazov používaná pri slovníkovom preklade. Určuje možnosť prekladu zdrojového výrazu na cieľový vyraz. Pri preklade slovo-na-slovo určuje preklad zdrojového slova na cieľové slovo.
Prepojenia	Výsledok procesu zarovnávania. Prepojenia ukazujú na korešpondujúce jednotky medzi zdrojovým a cieľovým textom.
Referenčný preklad	Korektný preklad zhotovený väčšinou ľudským prekladateľom. Používa sa pri porovnávaní kvality zhotoveného iného prekladu vzhľadom na tento preklad.
Slovník prekladových párov	Používa sa pri slovníkovom preklade. Obsahuje prekladové páry medzi zdrojovým a cieľovým jazykom.
Strom nejednoznačností	Dátová štruktúra udržiavajúca všetky možné preklady pre každé zdrojové slovo. Slúži na efektívne prehľadávanie za účelom nájdenia komplexne najpravdepodobnejšieho prekladu podľa jazykového modelu.
Vetný pár	Vety v zdrojovom a cieľovom jazyku, ktoré sú si

	navzájom prekladom.
Viac-slovné jednotky	Slovné sekvencie alebo skupiny slov.
Vyhodnotenie	Spôsob akým systémový vývojári zisťujú kde a čo je potrebné v systéme zlepšiť, užívatelia zasa zisťujú slabé a silné stránky systému.
Zarovnanie	Proces výberu korešpondujúcich jednotiek v zdrojovej a cieľovej vete. Zarovnanie je identické s párovaním, ktoré vytvára prepojenia v bitextoch.
Zdrojový text	Pôvodný text, ktorý sa prekladá na cieľový text.
Zlatý štandard	Vzorka bitextov, ktoré boli manuálne zarovnané jednou alebo viacerými osobami, a následne použité na automatické zarovnanie výstupu.

10 Literatúra

- Ahrenberg, L., M. Merkel, and M. Andersson (1998). A simple hybrid aligner for generating lexical correspondences in parallel texts. In Christian Boitet and Pete Whitelock, editors, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (ACL) and the 17th International Conference on Computational Linguistics (ACL/COLING), pages 29-35, Montreal, Canada, 1998.Morgan Kaufmann Publishers.
- Ahrenberg, L., M. Merkel, A. SÍgvall Hein, and J. Tiedemann (1999). Evaluation of LWA and UWA. Technical Report 15, Department of Linguistics, Uppsala University, Uppsala, Sweden.
- Bémová, A. and V. Kuboň (1990). Czech-to-Russian Transducing Dictionary; In: Proceedings of the XIIIth COLING conference, Helsinki 1990
- Brown, P. et al. (1990). A statistical approach to machine translation. Computational Linguistics 16(2), 1990:79-85.
- Brown, P.F., J.C. Lai, and R.L. Mercer (1991). Aligning sentences in parallel corpora. Proceedings from the 29th Annual Meeting of the Association for Computational Linguistics (ACL-91): 169-176.
- Brown, P., S. Della Pietra, V. Della Pietra, and R. Mercer (1993). The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics, 19(2).
- Clarkson, P.R. and R. Rosenfeld (1997). Statistical Language Modeling Using the CMU-Cambridge Toolkit. http://mi.eng.cam.ac.uk/%7Eprc14/toolkit.html From Proceedings ESCA Eurospeech 1997
- Cuřín J.: Statistical Machine Translation Quick Run Package (version 1.2). http://ufal.mff.cuni.cz/pcedt/tools/SMT_QuickRun/Doc/SMT_QuickRun.html
- Čerešňa M.: http://www.dbai.tuwien.ac.at/staff/ceresna/
- Francis, W. M. and H. Kučera (1964). Brown Corpus Manual of Information. Department of Linguistics, Brown University. Also available at http://khnt.hit.uib.no/icame/manuals/brown/
- Gale, W. and K.W.Church (1991). A program for aligning sentences in bilingual corpora. Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics: 177-184.

- Germann, U., Jahr, M., Knight, K., Marcu, D., and Yamada, K. (2001) Fast Decoding and Optimal Decoding for Machine Translation. http://www.isi.edu/natural-language/software/decoder/manual.html Proceedings of ACL-01. Toulouse, France.
- Gilleland Michael: Levenshtein Distance, in Three Flavors. http://www.merriampark.com/ld.htm
- Hajíč J., Kuboň V., Hric J. (2000). Machine Translation of Very Close Languages. http://acl.ldc.upenn.edu/A/A00/A00-1002.pdf. In 6th ANLP Conference / 1st NAACL Meeting. Proceedings. Seattle, Washington
- Hiemstra, D. (1998). Multilingual domain modeling in Twenty-One: Automatic creation of a bi-directional translation lexicon from a parallel corpus. In Peter-Arno Coppen, Hans van Halteren, and Lisanne Teunissen, editors, Proceedings of the 8th Meeting of Computational Linguistics in the Netherlands (CLIN), number 25 in Language and Computers: Studies in Practical Linguistics, pages 41-58, Nijmegen The Netherlands. Rodopi, Amsterdam, Atlanta.
- Isabelle, P. (1992). Bi-textual aids for translators. In Proceedings of the 8th Annual Conference of the UW Centre for the New OED and Text Research, pages 76-89, University of Waterloo, Waterloo, Canada.
- Kirschner, Z. (1987). APAC3-2: An English-to-Czech Machine Translation System; Explizite Beschreibung der Sprache und automatische Textbearbeitung XII1, MFF UK Prague
- Knight K. et al. (1999a). The EGYPT Statistical Machine Translation Toolkit. http://www.clsp.jhu.edu/ws99/projects/mt/toolkit/. Center for Language and Speech Processing at Johns-Hopkins University (CLSP/JHU).
- Knight K. (1999b). A Statistical MT Tutorial Workbook. http://www.clsp.jhu.edu/ws99/projects/mt/wkbk.rtf Center for Language and Speech Processing at Johns-Hopkins University (CLSP/JHU).
- Koehn P. (2002). Europarl: A Multilingual Corpus for Evaluation of Machine Translation. http://people.csail.mit.edu/~koehn/publications/europarl.ps. Information Sciences Institute, University of Southern California
- Korpus prim0.2: Slovenský národný korpus. Bratislava: Jazykovedný ústav Ľ. Štúra SAV 2004. Dostupný z WWW: http://korpus.juls.savba.sk.

- Korpus SYN2000: Český národní korpus - SYN2000. Ústav Českého národního korpusu FF UK, Praha 2000. Dostupný z WWW: <http://ucnk.ff.cuni.cz>.
- Kuenning G.: Dictionaries for International Ispell. http://fmg-www.cs.ucla.edu/geoff/ispell.html
- Melamed, I.D. (1995). Automatic Evaluation and Uniform Filter Cascades for Inducing N-Best Translation Lexicons. Proceedings of the Third Workshop on Very Large Corpora. Cambridge: 184-198.
- NIST (2002). Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. http://www.nist.gov/speech/tests/mt/
- Och, F.J. and H. Ney (2000). A comparison of alignment models for statistical machine translation. In Proceedings of the 18th International Conference on Computational Linguistics (COLING), pages 1086-1090, Saarbrücken, Germany.
- Och, F.J. (2001). mkcls: Training of word classes. http://www.fjoch.com/mkcls.html
- Och, F.J. and H. Ney (2003). A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics 29(1) March 2003: 19-52.
- Och, F.J. (2003). GIZA++: Training of statistical translation models http://www.fjoch.com/GIZA++.html
- Papineni K., Roukos S., Ward T., Zhu W-J (2001). Bleu: a Method for Automatic Evaluation of Machine Translation. http://www1.cs.columbia.edu/nlp/sgd/bleu.pdf IBM Research Report 2001
- Sedláček R. (1999). Morfologický analyzátor češtiny ajka. http://nlp.fi.muni.cz/projekty/ajka/ajkacz.htm
- Sedláček R. (2005). ajka tagset. http://nlp.fi.muni.cz/projekty/ajka/tags.pdf
- Teos Trenčín (2004). PC Translator 2004. http://www.teos.sk/translat.htm
- Tiedemann, J. (2003). Recycling Translations - Extraction of Lexical Data from Parallel Corpora and their Application in Natural Language Processing. Doctoral Thesis. Uppsala University. Department of Linguistics. ISSN 1652-1366, ISBN 91-554-5815-7.
- Van der Eijk, P. (1993). Automating the acquisition of bilingual terminology. In Proceedings of the 6th Conference of the European Chapter of the

Association for Computational Linguistics (EACL), pages 113-119, Utrecht/The Netherlands.

- Wang X. (2004). Evaluation of Two Word Alignment Systems. http://www.ep.liu.se/exjobb/ida/2004/dd-d/019/exjobb.pdf. Linköpings Universitet. ISRN: LITH-IDA-EX--04/019--SE