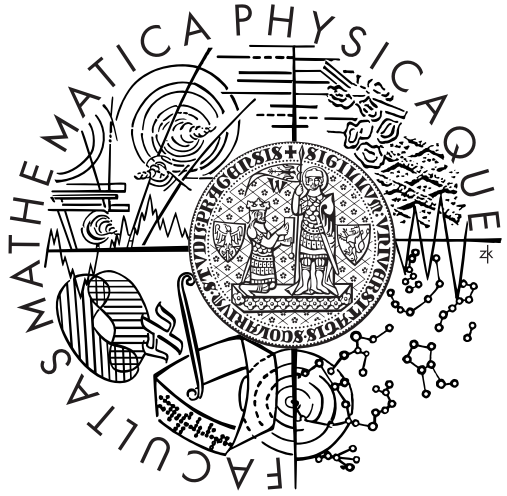
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Miroslav Týnovský

Využití lingvistických informací při EBMT

Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce práce: RNDr. Vladislav Kuboň, Ph.D.
Studijní program: Informatika, počítačová a formální lingvistika

2007

Děkuji především RNDr. Vladislavu Kuboňovi, Ph.D. za laskavé a podnětné vedení práce.

Poděkování si zaslouží také všichni ti, kteří pomohli cennými radami. Jsou to Eelco Mossel a Michael Baum (parsing němčiny), Cristina Vertan, Natalia Elita, Monica Gavrila (konzultace EBMT metodiky), Zdeněk Žabokrtský (parsing češtiny), Oldřich Krůza (Perl), Ondrej Šterbák (svn, vim). Zvláštní dík patří také Martině Špakové a Martinu Tkáčovi, kteří pomohli s hodnocením překladů.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 10. srpna 2007 Miroslav Týnovský

Obsah

Název práce: Využití lingvistických informací při EBMT
Autor: Miroslav Týnovský
Katedra (ústav): Ústav formální a aplikované lingvistiky
Vedoucí diplomové práce: RNDr. Vladislav Kuboň, Ph.D.
E-mail vedoucího: Vladislav.Kubon@mff.cuni.cz

Abstrakt: Metoda strojového překladu založená na příkladech (EBMT) je korpusová metoda strojového překladu, která se pokouší získat překlad vstupního textu pomocí analogie s překladem textu podobného již hotového.

Tato práce zkoumá význam využití lingvistické informace v této metodě překladu, a to konkrétně na dvou jazykových párech: čeština-angličtina a čeština-němčina. Zahrnuje shromáždění anotovaných paralelních dat pro jazykový pár čeština-němčina, návrh experimentálního EBMT systému, jeho implementaci a vylepšování jeho části s použitím lingvistických informací. Práce také obsahuje podrobné vyhodnocení jak výchozího systému, tak systému využívajícího informace o morfologii a syntaxi a jejich porovnání. Vyhodnocení systému bylo provedeno jednak automatickými metodami BLEU, NIST a METEOR a jednak ručně za pomoci anotátorů. Lingvistické informace aplikované na experimentální EBMT systém zahrnují morfologické a syntaktické porovnávání vstupní věty s příklady v překladové paměti.
Klíčová slova: Strojový překlad založený na příkladech, paralelní korpusy, evaluace strojového překladu.

Title: The Exploitation of Linguistic Information in EBMT
Author: Miroslav Týnovský
Department: Institute of Formal and Applied Linguistics
Supervisor: RNDr. Vladislav Kuboň, Ph.D.
Supervisor's e-mail address: Vladislav.Kubon@mff.cuni.cz

Abstract: Example-based machine translation (EBMT) is a corpus-driven method of machine translation. It builds the translation using analogy of the input text with a translation already made.

The benefit of using linguistic knowledge within EBMT is the subject of this thesis. Two language pairs are covered: Czech-English and Czech-German. The thesis covers gathering annotated parallel Czech-German data, design and implementation process of an experimental EBMT system, and the effort to improve it using linguistic knowledge. Detailed evaluation and comparison of both the baseline EBMT and the linguistically enhanced system are described. Evaluation has been done using machine and human evaluation methods. The three automatic evaluation methods are BLEU, NIST and METEOR. The linguistic enhancement of the baseline EBMT system includes comparisons of the input sentence with the examples in the translation memory based on morphology and syntax.
Keywords: Example based machine translation, parallel corpora, machine translation evaluation.

Kapitola 1

Úvod

Strojový překlad založený na příkladech (Example-based machine translation, EBMT) je alternativní metodou k dvěma hlavním směrům strojového překladu, k progresivnímu strojovému překladu statistickému a k tradičnímu strojovému překladu založenému na pravidlech. Cílem práce je prozkoumání možností využití lingvistických informací při překladu právě metodou EBMT. K jeho dosažení vede cesta spočívající v provádění dílčích úkonů, jejichž popis obsahují další strany této práce.

Prvním z nich je představení problematiky EBMT a strojového překladu obecně, které zahrnuje popis historického vývoje této disciplíny, poznání aktuální fáze vývoje a současných systémů.

Metoda EBMT je *data-driven*, což znamená, že pro překlad čerpá znalosti z lingvistických studnic dat, korpusů. A protože překlad probíhá mezi aspoň dvěma jazyky, musí jít o korpusy dvoj- či vícejazyčné. Tato práce se zabývá překladem z češtiny do angličtiny a z češtiny do němčiny, zapotřebí jsou tedy korpus česko-anglický a česko-německý. Jejich získáním se zabýváme v kapitole 3.

Konečně dalším nezbytným úkonem je samotný návrh a implementace prototypu EBMT systému. Kapitola 4 popisuje výchozí systém bez lingvistiky, kapitola 5 pak obsahuje možnosti využití lingvistických informací a jejich začlenění do výchozího systému. V obou těchto kapitolách jsou uvedena také hodnocení kvality překladu těmito systémy.

V závěrečné kapitole 6 shrnujeme dosažené výsledky a diskutujeme o možných vylepšeních výsledného systému.

Práci doplňuje pět příloh

- příloha A obsahuje ukázku anotovaných dat z použitých korpusů,
- příloha B obsahuje ukázky překladů jednotlivých variací implementovaného systému,
- příloha C stručný návod k použití implementovaného experimentálního systému,
- příloha D seznam a popis souborů na přiloženém CD

- a přílohou E je samotné CD s implementovaným systémem, nástroji pro konverzi datových formátů a sestaveným česko-německým korpusem s morfosyntaktickou anotací.

Kapitola 2

Vymezení strojového překladu na základě příkladů

2.1 Strojový překlad

2.1.1 Historie

Strojový překlad zůstává již více než 50 let obtížnou výzvou pro informatiky a počítačové lingvisty. Úplně první myšlenky strojového překladu sahají až do roku 1933. Francouz Georges Artsrouni a Rus Petr Trojanskij si nezávisle na sobě patentovali návrhy překladových strojů. Trojanského návrh již dokonce prognózoval jakýsi univerzální jazyk jako mezistupeň při překladu mezi více jazyky.

První pokusy o překlad jednoho přirozeného jazyka do druhého přišly v 50. letech 20. století – v lednu 1954 firma IBM představila tzv. Georgetown-IBM experiment, první relativně úspěšný systém strojového překladu. Tento systém překládal sice jen přibližně 60 vybraných vět z ruštiny do angličtiny, ale jeho význam je velký, protože vzbudil velké nadšení a byl počátečním motorem výzkumu v oblasti strojového překladu.

Následovalo dvanáct optimistických let. Výzkum strojového překladu probíhal na mnoha amerických univerzitách a byl také vládou štědře dotován. Byly představeny různé systémy, které typicky sestávaly z velkých slovníků a pravidel pro generování správného slovosledu v cílovém jazyce (např. Mark II z dílny IBM a Univerzity ve Washingtonu nebo systém Univerzity v Georgetownu). Kvalita výstupu těchto systémů však byla nízká a při snaze ji zvýšit se začaly objevovat komplikované lingvistické problémy, například nejednoznačnost nebo závislost významu na znalosti okolního světa. Tyto problémy přinesly zklamání, to vyvrcholilo negativním vyjádřením výboru ALPAC (Automatic Language Processing Advisory Commitee) v roce 1966.

Důsledkem tohoto vyjádření bylo zastavení výzkumu strojového překladu v USA na více než deset let a jeho omezení i v dalších zemích, především v Sovětském svazu a ve Velké Británii. Jednou z mála výjimek byl projekt SYSTRAN realizující překlad z ruštiny do angličtiny, který byl dokončen v roce 1970 skupinou vědců pod vedením Petera Tomy z bývalého georgetownského týmu. Tento systém byl

později rozšířen o spoustu jazykových párů a používá se dodnes v online překladačích a je oficiálním automatickým nástrojem pro překlad dokumentů Evropské unie.

Výzkum pokračoval v ostatních zemích, především v Kanadě, Francii a Německu. V Saarbrückenu se v roce 1967 začal vyvíjet systém pro překlad z ruštiny do němčiny nazvaný SUSY. Ten byl později rozšířen o angličtinu a francouzštinu. V Montrealu to byl zase systém Météo pro překlad meteorologického zpravodajství z angličtiny do francouzštiny. Tento systém byl dokončen v roce 1976 a používá se dodnes. Navíc pro něj byl současně vyvinut formalismus Systémy Q, který je ideovým předchůdcem programovacího jazyka Prolog.

V sedmdesátých letech se tedy projevil i pozitivní dopad vyjádření ALPAC, týmy opustily myšlenku přímého překladu a začaly vyvíjet sofistikovanější metody, které využívaly nějakých mezistupňů mezi zdrojovým a přeloženým textem, které vznikly syntaktickou nebo sémantickou analýzou vstupního textu a překladem na úrovni této analýzy. Některé systémy využívaly také tzv. interlingvu, jakýsi univerzální jazyk, do něhož se vstupní text převedl, aby z něj byl následně přímo generován výstupní text. V roce 1978 zahájila Komise Evropských společenství velmi ambiciózní projekt Eurotra, který měl zajistit strojový překlad mezi jazyky všech členských zemí.

V osmdesátých letech pokračuje výzkum pokročilých technik nepřímého překladu, objevuje se také myšlenka strojového překladu na základě příkladů. Dochází k opětovnému „probuzení“ disciplíny strojového překladu po odeznění negativního dopadu zprávy ALPAC. Objevuje se mnoho nových systémů, problematikou strojového překladu se začínají zabývat další pracoviště. Výzkum strojového překladu začínají podporovat i velké společnosti (např. Philips – projekt Rosetta, Hitachi – HICATS, Fujitsu – projekt ATLAS, Toshiba – podpora projektu Mu kjótské univerzity, BSO – projekt DLT a další). Pozadu však nezůstávají ani akademické projekty (např. skupina GETA v jihofrancouzském Grenoble – projekt Ariane, Univerzita v Saarbrückenu – projekt SUSY, pokračující projekt Eurotra a další). Systémů vzniká velké množství, jejich popis je nad rámec tohoto stručného shrnutí.

Zlomovým rokem ve vývoji strojového překladu byl rok 1990. Společnost IBM přišla se svým modelem statistického překladu, jenž se stal následně dominantním směrem. Současně japonská skupina začala realizovat ideje překladu na základě příkladů z 80. let.

Devadesátá léta tedy přinesla dynamický rozvoj statistického překladu, ale také přesun od teoretického výzkumu k praktickému využití dosažených poznatků. Příkladem je překlad mluvené řeči, což bylo také ambiciózním cílem německého projektu Verbmobil, nástupce neúspěšného projektu EUROTRA.

V posledních letech se dále vylepšují stejné metody, jejich kombinací se vytvářejí hybridní systémy. Do popředí zájmu se také dostávají systémy pro strojový překlad v reálném čase (především pro překlad webových stránek, kde se v posledních letech stala významným hráčem společnost Google). Významnou doménou jsou také nástroje, jež nepřekládají samy o sobě, ale pomáhají lidskému překladateli.

Tento historický souhrn vychází z práce [Hutchins06], kde lze najít více podrobností o zmiňovaných systémech. V dalších odstavcích ještě stručně shrneme historii strojového překladu na Matematicko-fyzikální fakultě v Praze.

Úplně prvním českým pokusem o strojový překlad bylo přeložení jedné věty na počítači SAPO v roce 1957. Prvním významným projektem byl systém překladu z angličtiny do češtiny APAČ. Ten byl vyvíjen na Matematicko-fyzikální fakultě v 80. letech Zdeňkem Kirschnerem. Pro překlad používal slovník s 1500 výrazy a transdukční slovník pro překládání koncovek převzatých slov.

V druhé polovině 80. let byl na podobných principech vyvíjen systém překladu z češtiny do ruštiny RUSLAN. Pro překlad používal slovník o velikosti 8500 slov, transdukční slovník a gramatiku ve formalismu Systémy Q. Výzkum byl zastaven těsně před provozními zkouškami v roce 1990.

V roce 1998 se začal vyvíjet systém Česílko pro překlad mezi velmi podobnými jazyky. Jeho doménou je lokalizace velkých softwarových systémů. Tato lokalizace má probíhat ve dvou krocích, prvním krokem je lidský překlad ze zdrojového, typově odlišného jazyka do tzv. pivotního jazyka a druhým krokem je překlad strojový z pivotního jazyka do jazyků velmi podobných. Systém Česílko bere češtinu jako pivotní jazyk pro podobné (slovanské) jazyky. Analýza se provádí pouze na úrovni morfologie. Původně byl systém Česílko nástrojem určeným pro překlad z češtiny do slovenštiny, v současné době je implementován překlad do několika slovanských jazyků a stále pokračují práce na přidávání dalších cílových jazyků.

Další současný výzkum strojového překladu se utváří kolem dvojjazyčného Pražského česko-anglického korpusu (PCEDT). Kromě klasických statistických metod (např. balík SMT Quick Run, který je součástí distribuce PCEDT) se zkoumá také využití provedené anotace na těchto datech – morfologie, syntax, tektogramatika (např. systém DBMT [Čmejrek03] nebo teprve připravovaný systém TectoMT Zdeňka Žabokrtského).

2.1.2 Metody strojového překladu

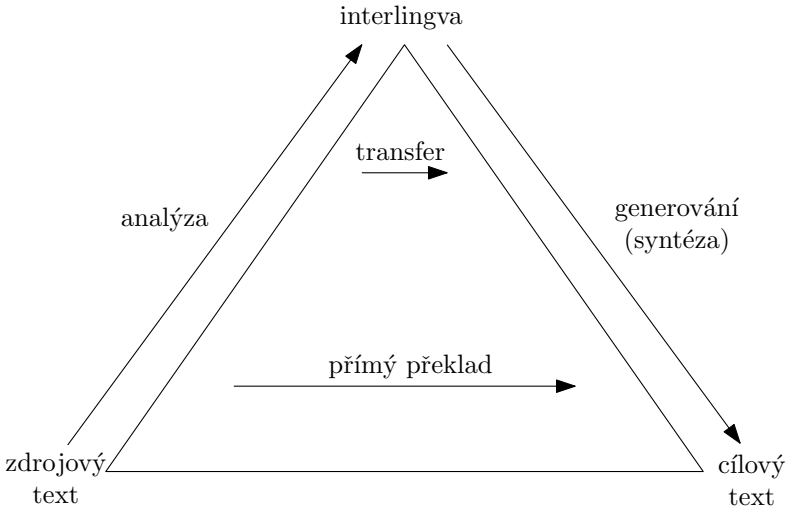
Metody strojového překladu můžeme rozdělit do dvou hlavních skupin:

- Strojový překlad na základě pravidel (rule-based machine translation, dále RBMT)
- Strojový překlad na základě korpusu (data-driven machine translation, corpus-based machine translation, dále CBMT)

Strojový překlad na základě pravidel

RBMT systém používá pro překlad lingvistická pravidla. Sestává ze tří procesů:

- Analýzy textu ve zdrojovém jazyce (např. morfologické, syntaktické)
- Přeložení struktur vzniklých analýzou do odpovídajících struktur pro cílový jazyk (tzv. transferu).



Obrázek 2.1: Vauquoisova pyramida – znázornění strojového překladu

- Následné syntézy textu v cílovém jazyce.

Tyto fáze se zpravidla vyobrazují na tzv. Vauquoisově pyramidě (viz Obrázek 2.1). V ní jsou také vyznačeny dva extrémní případy. První z nich je překlad přímý, ve kterém chybí analýza a syntéza a překládá se na úrovni slovních tvarů beze snahy vstupnímu textu (částečně) porozumět. Druhým extrémem je překlad s interlingvou, ve kterém zase zcela chybí transfer – vstupní text se analýzou převede do reprezentace nezávislé na jazyce a z ní se přímo generuje překlad.

RBMT je tradiční metoda, metody využívající korpus se poprvé objevily až na konci 80. let. Její hlavní nevýhodou je to, že pro vývoj RBMT systému je potřeba mnoho roků práce týmu lingvistů, kteří sestavují jednotlivá pravidla pro jednotlivé fáze. Výhodou je zase přesný popis fungování systému, který umožňuje provádět úpravy cílené na konkrétní nedostatky.

Strojový překlad na základě korpusu

Alternativním směrem jsou metody CBMT, které se snaží odstranit hlavní nedostatek RBMT, spočívající v dlouhém shromažďování informací (sestavování pravidel). Předpokládají existenci dvojjazyčného paralelního korpusu (databázi přeložených vět), využívají ho právě pro získání potřebných znalostí pro překlad. Práce lingvistů je tedy nahrazena automatickou extrakcí znalostí z dat. V rámci CBMT jsou dva základní směry:

- Statistický strojový překlad (statistical machine translation, dále SMT)
- Strojový překlad na základě příkladů (example-based machine translation, dále EBMT)

Statistický strojový překlad převádí problém překladu na problém maximalizace pravděpodobnosti. Z korpusu jsou natrénovány dva modely:

- jazykový model cílového jazyka – přiřazuje větě v cílovém jazyce tím vyšší pravděpodobnost, čím je v nějakém smyslu bližší plynulé gramatické větě tohoto jazyka. Na jeho natrénování stačí jednojazyčný korpus.

- překladový model – přiřazuje dvojici vět ze zdrojového a cílového jazyka tím vyšší pravděpodobnost, čím bližší je jejich význam. K jeho natrénování je třeba dvojjazyčného korpusu.

Systém vybere takový překlad zdrojového textu, pro který je součin pravděpodobností podle těchto modelů největší. Statistický strojový překlad je v současné době dominujícím směrem.

Druhý uvedený směr v rámci CBMT – metoda strojového překladu na základě příkladů – je založen na předpokladu, že jednotlivé části vstupního textu, který chceme překládat, už přeloženy byly. Stačí tyto jednotlivé kousky najít v databázi přeložených příkladů a ekvivalenty v cílovém jazyce umně poskládat dohromady.

Tato diplomová práce má ověřit, zda je přínosné do databáze příkladů přidat lingvistickou informaci a využívat ji při překladu. Strojový překlad na základě příkladů tedy představíme podrobněji než předchozí metody v následující kapitole.

2.2 Překlad na základě příkladů (EBMT)

2.2.1 Historie

S myšlenkou této metody přišel japonský profesor Makoto Nagao v roce 1981 a popsal ji o tři roky později ve své práci [Nagao84]. Popisuje v ní motivaci tohoto přístupu: člověk při překladu nepoužívá lingvistickou analýzu, nýbrž rozdělí překládaný text do určitých menších částí, ty přeloží, a poté je zase poskládá do jedné nebo více gramatických vět v cílovém jazyce. EBMT je tedy automatická metoda, která se snaží uspět při překládání právě imitací tohoto lidského přístupu. K tomu používá databázi příkladů, tj. nějakých úseků textu, které jsou ve zdrojovém i cílovém jazyce. Překládaný text rozdělí a snaží se jednotlivé části přeložit na základě analogie s některým z těchto příkladů.

Nagao již také popsal rozdělení tohoto procesu do tří fází:

- rozdělení vstupní věty na menší kousky (fragmenty) a nalezení jim podobných příkladů v databázi příkladů (*matching*),

- přeložení těchto fragmentů do cílového jazyka analogicky s překladem podobného příkladu (*alignment*) a

- správné poskládání těchto přeložených fragmentů do celé věty v cílovém jazyce (*recombination*).

Do stejného nebo jen krátce pozdějšího období patří i výzkum skupiny DLT v Utrechtu, který se věnuje i použití míry podobnosti na základě sémantické podobnosti.

Rozsáhlejšího vývoje se však tato metoda dočkala až v devadesátých letech zpočátku především v Japonsku. Sato a Nagao rozpracovali metodu pro matching nepřesné shody a publikovali ve své práci [Sato90]. Ve stejném roce přišel Eiichiro Sumita s praktickým uplatněním této metody tím, že navrhl systém pro překlad japonské předložkové fráze „N_1 no N_2“ do různých anglických předložkových frází. Obě tyto práce však používají pro porovnávání míru založenou na sémantické podobnosti slov a narážejí tedy na problém získání thesauru s ohodnocenou blízkostí synonym.

V 90. letech také vzniklo několik systémů obecného strojového překladu založeného čistě na příkladech. Výzkum se rozdělil do různých větví, některé systémy se snaží vyhledávat podobnost na podstromech syntaktických stromů, jiné se pokoušejí nacházet podobné celé věty a překlad konstruovat jejich pozměněním. Pokračují také pokusy s přesnou lexikální shodou.

Po roce 2000 diverzita výzkumu EBMT pokračuje, objevují se aplikace EBMT přístupu v různých specializovaných systémech, například při překladu mluvené řeči. Metoda EBMT se také využívá jako součást hybridních systémů.

2.2.2 Hlavní myšlenky EBMT

Ze stručného souhrnu historie vývoje EBMT jsou přibližně jasné principy a charakteristiky této metody. EBMT je korpusová metoda překladu, která používá databázi příkladů a princip analogie pro přeložení vstupní věty. Tato (nebo nějaká podobná) vágní definice dostačuje pro obecnou představu o EBMT. Absence exaktní definice ostatně nemusí při rozvoji určité disciplíny vadit. Příkladem je třeba umělá inteligence, která také postrádá exaktní definici, a přitom prochází bouřlivým rozvojem. Nutnost exaktní definice se však ukáže při pokusu důsledně vymezit EBMT v kontextu ostatních metod strojového překladu. Toho jsou si vědci obzvláště v posledních letech vědomi, a tak se objevují pokusy nějakým způsobem EBMT víceméně přesně definovat[1].

Abychom rozebrali definice jednotlivých autorů, pokusíme se vyjmenovat všechny dosud zmíněné rysy EBMT a následně popíšeme, které z nich jednotliví autoři považují za důležité pro charakteristiku EBMT.

1. EBMT je metoda překladu imitující lidského překladatele – pokouší se o překlad jednotlivých jednoduchých frází ze zdrojového textu analogií s již provedeným překladem podobných frází.

2. EBMT je metoda navazující na RBMT a snaží se vyřešit jeho nedostatky, jako jsou špatné překlady kolokací a chyby v pořádku slov.

3. EBMT nahrazuje složité procedury lingvistické analýzy databází příkladů.

4. EBMT má třífázovou architekturu – matching, alignment, recombination.

[1] Například v článcích [Somers03], [Turcato03] a [Hutchins05], které budu níže citovat.

5. EBMT má potenciál zlepšit kvalitu výstupních vět, protože překlad je založen na příkladech lidského překladu, namísto aby byl generován gramatikou nebo modelem cílového jazyka.

6. EBMT pracuje s frázemi namísto se slovy.

Harold Somers ve své práci zdůrazňuje hlavně použití databáze příkladů. Upíná se tedy především na bod (3) z našeho seznamu. Píše doslova, že báze znalostí je v příkladech, jako doplňující kritérium uvádí použití databáze příkladů za běhu systému jako kontrast vůči statistickým metodám, které používají dvojjazyčný korpus dopředu na natrénování překladového modelu. Toto kritérium je však příliš restriktivní – vyloučí i některé systémy považované za systémy EBMT.

Davide Turcato a Fred Popowich Somersovi oponují. Podle nich je irelevantní, jak je uchovávaná báze znalostí, klasifikaci překladových systémů by měla sloužit informace o tom, jaké znalosti se uchovávají a jakým způsobem se využívají. V závěru označují jako pravou metodu EBMT takovou, která se opírá o původní myšlenku překladu na základě analogie, tedy zdůrazňují především bod (1).

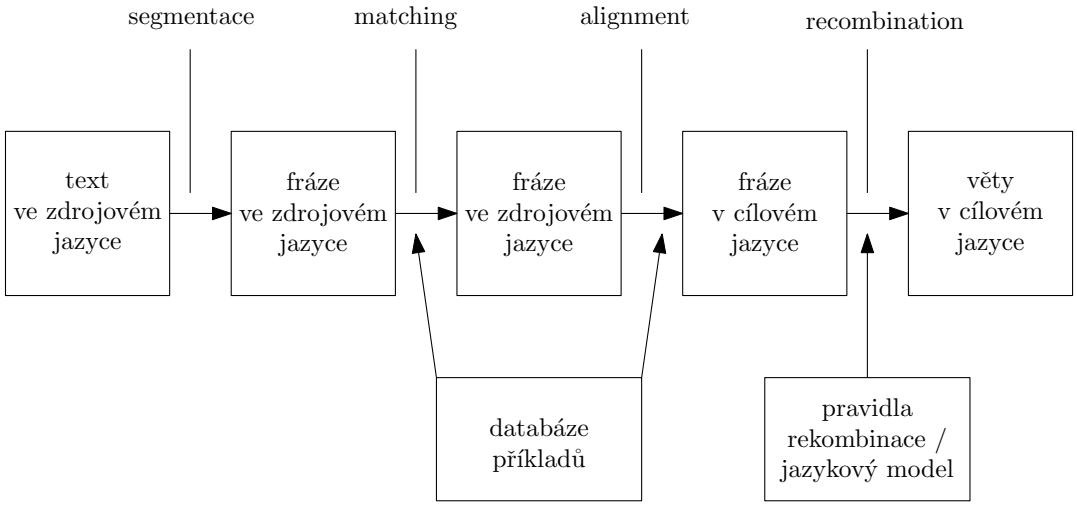
John Hutchins ve svém článku zmiňuje obě jmenované práce, ale nabízí ještě jiný pohled. Za důležitý považuje u každého systému především proces konverze ze zdrojového jazyka do cílového. Ostatní procesy, jako předzpracování vstupu, vyhlazení výstupu a předzpracování báze znalostí odsouvá do pozadí, stěžejním procesem pro klasifikaci MT systému je právě konverze.

V případě RBMT je tento stěžejní proces zprostředkován dvojjazyčným slovníkem a množinou pravidel pro převedení struktur zdrojového jazyka do cílového jazyka. V SMT je stěžejní proces zajištěn překladovým modelem, jehož vstupem jsou slova či fráze zdrojového jazyka. V EBMT je stěžejní proces nalezení fragmentů cílového jazyka, které odpovídají fragmentům ve zdrojovém jazyce. Ten je dle Hutchinse hlavním charakteristickým znakem EBMT. Předzpracováním vstupu je rozdělení zdrojových vět do těchto fragmentů a nalezení jim podobných v databázi příkladů a vyhlazením výstupu je poskládání přeložených fragmentů do gramatických vět. Tyto znaky jsou pro EBMT však až druhotné. Hutchins se tedy ve své (asi nejpropracovanější) definici odkazuje na body (1), (3), (4).

2.2.3 Vymezení EBMT oproti ostatním metodám

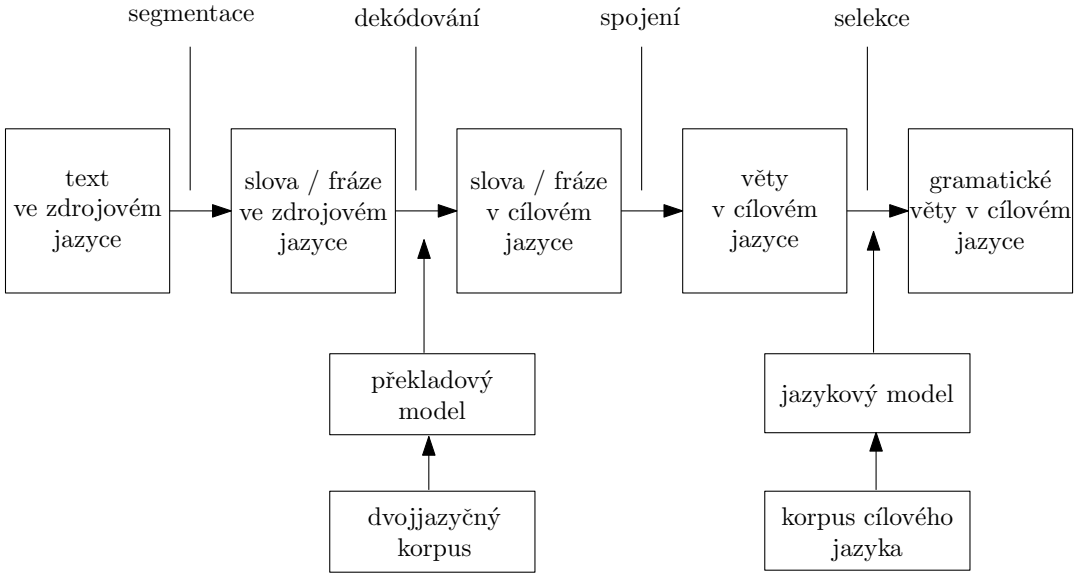
Pro dokončení obecného popisu pojmu Example-based machine translation uvedeme v této podkapitole porovnání této metody s ostatními dvěma, RBMT a SMT. U obou uvedeme společné rysy a rozdíly. K tomu pomohou názorné diagramy popisující každou ze tří porovnávaných metod. Tyto diagramy vycházejí z podobných, jež byly uvedeny v prezentaci k práci [Hutchins06].

Zřejmým společným rysem EBMT a SMT je zdroj znalostí potřebných pro překlad. Tím je u obou z nich dvoj- nebo vícejazyčný korpus. Společná také může být poslední fáze generování výstupu (u EBMT rekombinace, u SMT selekce, viz Obrázky 2.2 a 2.3), která může být i v případě EBMT založena na jazykovém modelu cílového jazyka.



Obrázek 2.2: Schéma systému překladu založeného na příkladech

Hlavním rozdílem EBMT vůči SMT byla velikost jednotek (fragmentů), na které se vstupní věta rozděluje. U EBMT jsou to typicky jednotky delší než slova (právě užití kontextu je výhodou této metody), zatímco u klasického SMT to byla právě slova. Toto rozlišení se však změnilo s nástupem frázových SMT metod, u kterých už ostatně odlišení EBMT není příliš jasné. Použití překladového modelu v SMT totiž můžeme chápat jako složení fází matchingu a alignmentu u EBMT.

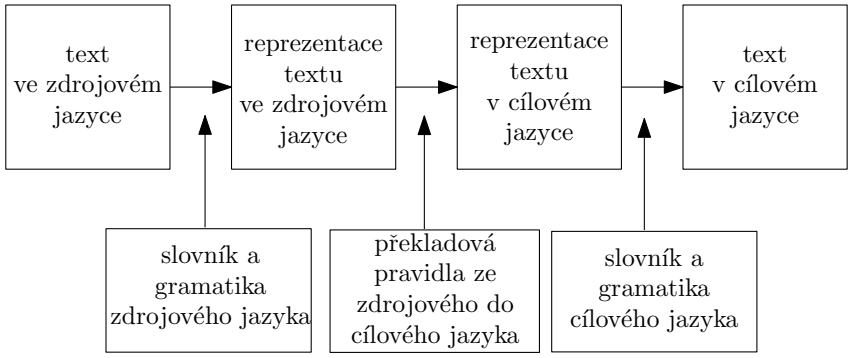


Obrázek 2.3: Schéma statistického systému překladu

Obecné společné rysy EBMT a RBMT je obtížné vyjmenovat, EBMT systémy však docela často využívají mechanismy RBMT, příkladem mohou být systémy se vzorky aj., které využívají pravidla pro alignment, nebo systémy s příklady

v podobě závislostních stromů, které používají stejné metody jako RBMT pro lingvistickou analýzu.

Naopak zase některé RBMT systémy využívají pravidla extrahovaná z databáze příkladů. Takové systémy bychom ve smyslu uvedených definic EBMT spíše označili jako example-based.



Obrázek 2.4: Schéma systému překladu založeného na pravidlech

Nakonec ještě zmiňme rozlišení EBMT systémů a systémů s překladovou pamětí (Translation Memory, TM), které trochu splývají kvůli příbuznosti problémů a podobné terminologii. Aspoň zde je však rozlišení naprosto jednoznačné: EBMT systémy jsou překladové systémy, zatímco systémy s překladovou pamětí jsou nástroje pro pomoc lidským překladatelům – Computer Aided Translation (CAT) systémy. TM systémy pomáhají překladateli tím, že se mu ukazují překlady podobných vět. S EBMT systémy tedy sdílejí pouze problematiku matchingu.

Vidíme, že vymezení EBMT není vůbec přímočaré. S trochou nadsázky se dá říci, že jako EBMT je možné označit snad všechny metody založené na korpusu, které nejsou statistické.

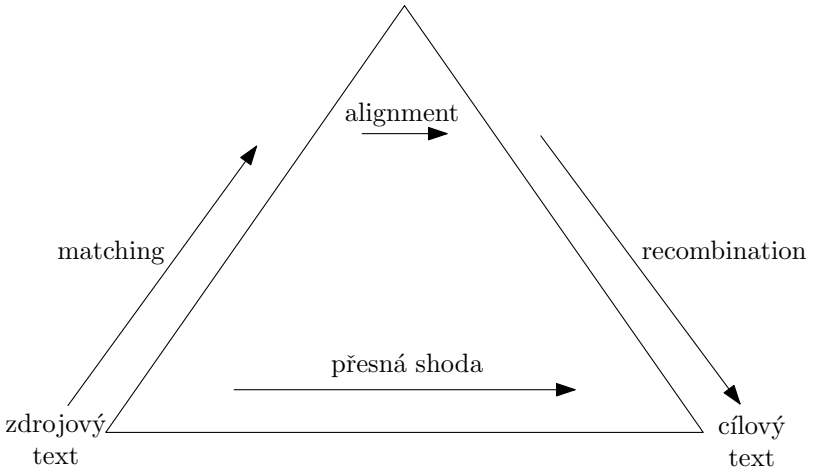
2.2.4 Jednotlivé fáze EBMT

Jak již bylo uvedeno v kapitole 2.2.1, proces překladu se dá rozdělit do tří podúkolů:

- Nalezení co nejpodobnějšího příkladu či více příkladů v databázi (matching).
- Překlad jednotlivých částí vstupu na základě nalezených příkladů (alignment).
- Poskládání přeložených částí zpět do souvislého textu (rekombinace, recombination).

Toto rozdělení lze také vyobrazit na analogii Vauquoisovy pyramidy pro EBMT[2] (viz Obrázek 2.5).

[2]Toto znázornění je též použito v [Somers03]



Obrázek 2.5: Vauquoisova pyramida pro example-based machine translation

Tomuto procesu ovšem musí předcházet zkonstruování databáze příkladů, popř. dalších bází znalostí, nad nimiž bude systém fungovat. Než rozebereme jednotlivé fáze, popišme ještě stručně problémy, které je třeba při přípravě systému rozhodnout.

Vlastní databáze příkladů mají v různých systémech různou podobu. Liší se počtem příkladů, jejich granularitou, obecností a jaké lingvistické informace jsou v příkladech obsaženy.

Problém počtu příkladů je zřejmý, čím méně příkladů, tím rychlejší prohledávání databáze, naopak do jisté míry platí, že s přibývajícími příklady roste kvalita produkovaného překladu.

Granularita určuje, jak „dlouhé“ příklady se použijí, zda budou tvořeny větami anebo většími či menšími jednotkami. Čím delší jednotky se použijí, tím menší je pravděpodobnost úplné shody, avšak čím kratší jednotky se použijí, tím větší je riziko, že výsledný překlad bude složen z nesouvisejících úseků a tudíž nízké kvality.

Konkrétní podoba příkladů může být prakticky libovolná – od čistého textu bez lingvistické informace, po anotované stromové struktury.

Metody matchingu

Prvním úkolem EBMT systému poté, co dostane na vstup větu k přeložení, je najít v databázi co nejpodobnější příklady k této větě. Pro splnění tohoto úkolu musí být systém schopen nějak určit, které příklady jsou si podobné a které ne. K tomu slouží funkce (similarity measure), která pro vstupní větu a nějaký příklad vrátí číslo, vyjadřující míru jejich podobnosti. Hledání co nejpodobnějších příkladů je ekvivalentní maximalizaci této funkce pro vstupní větu přes množinu všech příkladů.

Podobnost může být posuzována na různých úrovních od podobnosti řetězců znaků po podobnost na syntaktických stromech, samozřejmě v závislosti na tom,

jaké informace obsahuje databáze příkladů a jaké informace můžeme získat o vstupní větě. Podle tohoto rozdělení rozlišujeme:

- podobnost na znacích (character based similarity)
- podobnost na slovních tvarech (word based similarity, lexical similarity)
- podobnost na morfologické anotaci (annotated word based similarity)
- podobnost na syntaktických strukturách (structure based similarity)

První a nejjednodušší způsob, podobnost na znacích, pohlíží na vstup a příklady jako na řetězce znaků a zcela rezignuje na částečné porozumění. Klasickou mírou spadající do této kategorie je Levenshteinova vzdálenost (též editační vzdálenost). Ta udává počet editačních operací (Insert, Delete, Replace) nutných k přechodu od jednoho řetězce k druhému. Existují i jiné metody, sdílejí však bezzubost v podobě povrchního porovnávání řetězců. Mnoho těchto metod je implementováno v knihovně Simmetrics Sama Chapmmana z sheffieldské univerzity[3]. Ta je dostupná i se zdrojovými kódy a popisy jednotlivých metod.

Nejjednodušší metody v rámci určování podobnosti na slovech vycházejí ze znakových porovnání s tím rozdílem, že za jednotku berou slovní tvar. Spíše je však tato kategorie zastoupena metodami s využitím thesauru, což bylo ostatně i v původním popisu EBMT profesora Nagaa. Použití thesauru pomůže například při výběru správného příkladu pro anglickou vstupní větu
He eats potatoes.
při výběru z těchto dvou možností:
(1) *A man eats vegetables.* (2) *Acid eats metal.*

Samozřejmě je žádoucí, aby byla vybrána věta (1), protože sloveso *eats* má v tomto příkladě stejný význam jako ve vstupní větě. Kdyby se pro překlad použil příklad (2) třeba do češtiny, slovo *eats* by mohlo být přeloženo nesprávně jako *rozkládá*. K výběru správného příkladu (1) použití thesauru pomůže, protože ten vrátí vyšší sémantickou podobnost pro dvojici (*He*, *A man*), než pro (*He*, *Acid*) a taktéž i pro dvojici (*potatoes*, *vegetables*) než pro (*potatoes*, *metal*)[4].

Jinou klasickou metodou pro určování podobnosti je tzv. Carollův úhel podobnosti, který je užitečný, když příklad zahrnuje vstupní větu, ale obsahuje i něco navíc. V takovém případě by editační vzdálenost byla vysoká, avšak překládat na základě tohoto příkladu je vhodné. Carollův úhel podobnosti v této situaci poslouží, protože podobnost vyhodnotí jako velmi vysokou.

Všechny dosud zmíněné metody matchingu měly jedno společné, porovnávala se podobnost celého vstupu a příkladů. Oproti nim ještě stojí metoda matchingu jednotlivých částí vstupu pro dosažení pokrytí vstupní věty, tzv. „partial matching for coverage“.

Určováním podobnosti na syntaktických strukturách se v tomto textu zabývat nebudeme, abychom mohli ukázat rozdíl kvality překladu systémem bez užití lin-

[3] `http://www.dcs.shef.ac.uk/~sam/simmetrics.html`
[4] Tento příklad vychází právě z [Nagao84]

gvistické informace a systémem s užitím lingvistické informace, navrhneme v dalším lexikální systém, který bude lingvistickou informaci využívat jako doplněk. Podobný přístup popisuje článek [Brown99], avšak využití lingvistické informace se liší.

Popis matchingu na syntaktických strukturách je obsažen v článku [Somers03], ze kterého vychází i popis uvedených metod.

Metody alignmentu

Alignment v EBMT probíhá na dvou úrovních. První úrovní je tzv. větný alignment. Ten se využívá vždy při konstrukci databáze příkladů. Pomocí něj se určuje, který příklad ve zdrojovém jazyce odpovídá kterému příkladu v cílovém jazyce. Ovšem za předpokladu, že příklady jsou celé věty. Pokud jsou příklady tvořeny kratšími úseky, uplatňuje se ihned i alignment druhé úrovně popsaný níže.

Druhou úrovní je alignment jemnější než větný, tj. slovní nebo frázový. Ten určuje, která část příkladu v cílovém jazyce je překladem které části odpovídajícího příkladu ve zdrojovém jazyce. Může být konstruován taktéž předem při vytváření databáze příkladů anebo až v průběhu samotného překladu. Je nutný v okamžiku, kdy databáze neobsahuje příklad přesně se shodující se vstupem. Potom je třeba překlad zkonstruovat nějakou kombinací částí různých příkladů a překlady těchto částí určuje právě slovní nebo frázový alignment.

Metody větného alignmentu typicky předpokládají, že pořadí vět je ve zdrojovém a cílovém jazyce totožné. Také mají nějak omezenou korespondenci mezi větami v paralelních textech, například očekávají, že jedné větě ve zdrojovém textu mohou odpovídat maximálně dvě věty v cílovém textu a naopak. Pro nalezení této korespondence se používají převážně statistické metody založené na počtu slov a znaků a na výskytu podobných slov. Metody slovního alignmentu nemohou využít počtu slov či znaků, na druhou stranu však už hledají ekvivalenty jen v rámci vět stanovených větným alignmentem.

Metody rekombinace

Rekombinace je fází EBMT v literatuře nejvíce odbývanou. Jejím cílem je z přeložených příkladů do cílového jazyka vytvořit plynulý srozumitelný výstup. Prakticky se rozděluje na dva směry, jedním je nějaká forma pravidlového postprocessingu, druhým generování výsledného textu za pomoci statistického jazykového modelu.

2.3 Výhody a úskalí EBMT

Na závěr této kapitoly o vymezení systémů EBMT jen stručně uvedeme jejich výhody a nevýhody. Hlavní výhoda systémů EBMT plyne z toho, že jsou založeny na korpusu. Díky tomu stejně jako systémy SMT nahrazují časově náročné vytváření pravidel zbudováním paralelních korpusů. Přináší to ještě výhodu jazy-

kové nezávislosti, stejné principy a algoritmy pro překlad mohou být při záměně bilinguálního korpusu pro překlad mezi jiným jazykovým párem.

Oproti systémům SMT mají výhodu v transparentnosti fungování. To přináší možnost přímočaré opravy chyb. Pokud statistický systém generuje nějaké konkrétní chybné překlady, je poměrně obtížné upravit překladový model tak, aby byly eliminovány. Systém EBMT pro překlad používá přímo konkrétní příklady, tudíž chybný překlad může být eliminován prostým přidáním příkladů, které přinesou přesnou shodu.

EBMT má však i své neduhy. Nejčastěji zmiňovanými problémy EBMT jsou výpočetní náročnost, možnost sestavení výstupu z vzájemně nesouvisejících příkladů a tzv. *boundary friction*.

Výpočetní náročnost plyne samozřejmě z nutnosti prohledávání typicky rozsáhlé databáze příkladů. Řešeními jsou použití vhodných datových struktur, používání indexace a také omezení počtu příkladů jejich generalizací.

Problém *boundary friction*[5] nastává, když cílový jazyk má jemnější rozlišení nějaké gramatické kategorie. Dobře jej ilustruje tento příklad: Mějme v databázi příkladů tyto větné páry:

(1) The handsome boy entered the room.
Ten hezký hoch vešel do pokoje.

(2) I saw the handsome boy.
Viděl jsem toho hezkého hocha.

Pokud překládáme větu obsahující „the handsome boy“ do češtiny, kvalita překladu záleží na výběru příkladu, přestože v angličtině jde o přesně stejnou frázi.

Boundary friction je příklad problému EBMT, který by mohla lingvistická znalost vyřešit. Matching, který by bral v potaz syntaktickou kategorii, by zajistil výběr vhodnějšího příkladu.

[5] uvedené vysvětlení i příklad vycházejí z [Somers03]

Kapitola 3

Paralelní texty

Základem jakéhokoliv EBMT systému jsou samozřejmě paralelní texty, které buď přímo anebo po nějakém předzpracování slouží jako databáze příkladů. Pro lingvistikou obohacený systém však holé paralelní texty nestačí, jsou potřeba texty anotované. V této podkapitole popíšeme použité paralelní texty a způsoby jejich získání; česko-anglické texty byly i s anotací snadno dostupné v Pražském česko-anglickém závislostním treebanku, česko-německé texty jsem získal z JRC-Acquis korpusu a automaticky anotoval dostupnými nástroji.

3.1 Česko-anglické texty

3.1.1 Pražský česko-anglický závislostní treebank

Jako česko-anglická data jsem použil Pražský česko-anglický závislostní treebank 1.0[1] (dále PCEDT). Tento paralelní korpus sestává ze dvou částí, dvou zdrojů dat:

- překlad 21 656 vět z Penn Treebanku z angličtiny do češtiny
- Reader's digest korpus – překlad 450 článků z Reader's Digestu (z let 1993–1996) z angličtiny do češtiny

Data z Penn Treebanku jsou texty z ekonomického deníku Wall Street Journal, které byly přeloženy přímo pro PCEDT, a překlad je proto přizpůsoben potřebám paralelního korpusu. Díky tomu je jedna věta v originále přeložena vždy jednou větou v češtině (alignment s korespondencí 1 : 1). Navíc Penn Treebank je ručně syntakticky anotovaný, takže tato lidská anotace je v PCEDT přejata, je pouze automaticky převedena ze složkových stromů do závislostních. Česká část dat je anotována automaticky ve čtyřech fázích:

[1]Poznámka: Tento stručný popis Pražského česko-anglického závislostního treebanku není úplný, popisuje jej s ohledem na další využití dat v této práci. Není například vůbec zmíněna tektogramatická anotace části dat. Podrobnější popis je v [Čmejrek04], úplný v oficiální dokumentaci PCEDT (`http://ufal.mff.cuni.cz/pcedt/doc/PCEDT_main.html`)

- automatická tokenizace
- morfologické značkování (morfologické značkování přiřazuje každému slovu lemma a morfologickou značku, je popsáno v práci Jana Hajiče a Barbory Hladké [Hajič98])
- zbudování syntaktických stromů (syntaktické stromy jsou generovány dvěma parsery, Collinsovým a Charniakovým. Collinsův parser, jehož výstupy dále používám, má podle testu na testovacích datech Pražského závislostního korpusu 2.0 úspěšnost 82,43% [Collins99]. (úspěšnost odpovídá počtu správně rozpoznaných hran ku počtu všech hran).)
- přiřazení analytických funkcí uzlům [Žabokrtský02].

Důležitou součástí dat jsou také referenční překlady. Český překlad 515 vět byl čtyřmi nezávislými překladateli přeložen nazpět do angličtiny. Tyto překlady lze dobře využít při evaluaci překladového systému standardními evaluačními technikami.

Alignment vět je v těchto datech uchován implicitně, odpovídající věty mají (až na sufix) stejné identifikátory. To je možné díky již zmíněné korespondenci 1 : 1.

Data z Reader's digestu jsou naopak docela volné překlady. Proto je jen 43 969 z 54 091 zarovnaných segmentů překlad jedné věty na jednu větu. Navíc je anotace na obou stranách automatická. Z těchto důvodů jsem pro vývoj překladového systému použil pouze data založená na Penn Treebanku s Collinsovou anotací češtiny.

3.1.2 Formát CSTS a jazyk PML

PCEDT je distribuován ve formátu CSTS, což je hlavní datový formát pro Pražský závislostní treebank (PDT) 1.0. Je to formát založený na SGML a reprezentuje morfologickou, analytickou a částečně tektogramatickou rovinu anotace.

Pro PDT 2.0 byl formát CSTS nahrazen modernějším a lépe navrženým jazykem PML (Prague Markup Language) založeným na XML. Tento jazyk je vlastně jakýmsi obecnějším „nadformátem"; umožňuje definici formátů dat, schémat (nejen) pro reprezentaci lingvisticky anotovaných textů. Pro PDT 2.0 jsou v tomto jazyce nadefinována čtyři schémata:

- schéma pro neanotovaná (holá) data (`wdata_schema.xml`)
- schéma pro morfologickou rovinu anotace (`mdata_schema.xml`)
- schéma pro syntaktickou rovinu anotace (`adata_schema.xml`)
- schéma pro tektogramatickou rovinu anotace (`tdata_schema.xml`)

Na webové stránce PDT 2.0 je k dispozici nástroj pro konverzi CSTS do PML. Pomocí něj a jsem převedl vybranou část PCEDT do PML. Pro syntaxi anglických dat jsem vytvořil PML schéma analogické se schématem pro česká data, pouze s jinými možnostmi pro analytické funkce uzlů (soubor `adata_en_schema.xml` v adresáři `tred-things/resources` na přiloženém CD).

Výsledná česko-anglická data jsou tedy věty z Penn Treebanku s překlady na větách s korespondencí 1 : 1. V obou jazycích je na datech provedena morfologická a syntaktická (závislostní) anotace. Vše je uloženo v moderním formátu PML (přesněji v prvních třech ze čtyř výše zmíněných schématech nad jazykem PML – holá data, morfologická rovina, analytická rovina).

3.2 Česko-německé texty

3.2.1 Korpus JRC-Acquis

Příprava česko-německých dat byla obtížnější. Paralelní texty jsem získal z korpusu dostupného na webu. V úvahu připadaly dvě možnosti. První z nich byl korpus OPUS[2], který obsahuje jednak paralelní texty Evropské ústavy a jednak paralelní texty z dokumentací a lokalizací open-source projektů. Původně jsem chtěl použít hlavně texty z dokumentací, protože mají typicky omezený slovník a používají relativně jednoduché věty. Avšak ukázalo se, že kvalita těchto dat je nízká. Údajně české věty často obsahují nepřeložené části a některé jsou celé v originálním jazyce.

Z tohoto důvodu jsem použil druhou alternativu, a to korpus JRC-Acquis, který je postaven na textech Evropské legislativy v jednadvaceti oficiálních jazycích EU. Jedná se o obrovský objem textů z let 1958 až 2005 (rozsah celých dat podrobně ukazuje tabulka 3.1). Automatický alignment systémem Hunalign[3] těchto textů na větách je součástí dat.

Pro vybudování mého experimentálního korpusu jsem použil samozřejmě pouze texty české a německé o omezil jsem se na dokumenty z roku 2004. Pro toto omezení jsem se rozhodl především kvůli časové náročnosti německé anotace. Navíc jsem pro jednoduchost vybral pouze takové překlady, kde jedna česká věta odpovídá jedné německé větě. Po této selekci zůstalo přesně 10 937 německých a 10 937 českých vět (podrobnější statistiky jsou vyobrazeny v tabulce 3.2).

Na získaných datech jsem provedl automatickou syntaktickou anotaci na obou jazycích. Výstupy těchto anotací jsem poté převedl do jednotného formátu PML. Postupy anotace, konverze a použité nástroje podrobněji popíšu v následujících dvou podkapitolách.

[2] `http://logos.uio.no/opus/`

[3] `http://mokk.bme.hu/resources/hunalign`

Jazyk	Počet dokumentů	Počet slov	Počet znaků
čeština	7 983	5 979 261	38 479 314
dánština	7 939	6 548 461	44 444 011
němčina	7 914	6 576 633	47 047 334
řečtina	7 782	7 377 316	47 715 936
angličtina	7 972	7 512 013	45 150 120
španělština	7 809	7 964 255	48 281 455
estonština	7 944	4 925 361	38 603 952
finština	7 735	5 134 294	43 705 813
francouzština	7 862	7 812 577	45 609 935
maďarština	7 489	5 391 810	40 601 868
italština	7 872	7 264 126	46 792 286
litevština	7 966	5 386 359	39 936 370
lotyština	7 980	5 656 335	39 290 110
maltština	7 639	7 230 538	43 919 981
nizozemština	7 882	7 339 465	47 699 598
polština	7968	5 974 605	43 160 945
portugalština	7 848	7 851 904	47 225 710
rumunština	5 792	5 122 354	33 681 450
slovenština	5 278	3 911 895	26 077 956
slovinština	7 984	5 989 322	37 844 883
švédština	7 731	6 472 717	42 990 411
průměrně	7 636	6 353 410	42 340 925

Tabulka 3.1: Rozsah dat JRC-Acquis. Počty slov a znaků v tělech dokumentů.

3.2.2 Anotace němčiny

Pro anotaci německého textu jsem použil parser vyvíjený skupinou CDG-team na katedře NatS (Natürlichsprachliche Systeme) Univerzity v Hamburku, konkrétně tyto nástroje:

- CDG Tokenizer
- CDG Parser
- Postprocessing pro disambiguaci morfologických značek

Popis CDG parseru

CDG v názvu parseru je zkratkou pro Constraint Dependency Grammar. Jedná se o robustní parser založený na omezujících pravidlech, jehož výstupem jsou závislostní stromy. Omezující pravidla, která explicitně definují popisovaný jazyk (němčinu), jsou ohodnocená čísly z intervalu $(0, 1)$. Problém nalezení správného závislostního stromu pro vstupní větu je převeden na problém nalezení takového

Rozsah	Jazyk	Počet vět	Počet slov
Dokumenty z roku 2004	čeština	26 100	411 044
	němčina	22 030	519 117
Pouze věty 1 : 1	čeština	9 715	231 814
	němčina	9 715	223 355

Tabulka 3.2: Výběr česko-německých dat z korpusu JRC-Acquis.

stromu, který co nejméně porušuje daná omezení, tj. takový, jehož součin vah porušených omezujících pravidel je co největší. Platí totiž, že čím vyšší je ohodnocení omezujícího pravidla, tím menší postih je za jeho porušení.

Omezující pravidla německé gramatiky jsou psána ručně, pokusy o jejich automatickou extrakci dat nepřinesly navýšení úspěšnost parsingu. Parser je robustní, tzn. že pro každou větu vrací nějaký výsledek. Navíc v každém okamžiku parsingu má kandidáta na nejlepší výsledek. Díky tomu lze dobu jeho výpočtu omezit za cenu snížení úspěšnosti (v případě, že celý výpočet trvá delší dobu, než je stanovené maximum). Úspěšnost parseru v závislosti na maximální době výpočtu ukazuje tabulka 3.3. Test byl proveden na 1000 větách z německého NEGRA korpusu [Foth2000].

časový limit pro jednu větu	průměrný použitý čas pro jednu větu	úspěšnost
neomezený		90,9 %
600 s	68,0 s	89,0 %
400 s	59,3 s	88,7 %
200 s	44,9 s	88,2 %
100 s	31,8 s	87,1 %
50 s	21,6 s	84,6 %

Tabulka 3.3: Úspěšnost CDG parseru v závislosti na maximální době výpočtu.

Pro anotaci německých vět jsem použil časový limit 500 s, takže úspěšnost by měla být okolo 89 %, pokud by byla anotována stejně kvalitní data, jako jsou v NEGRA korpusu. Bohužel jsou v datech velmi často dlouhé věty a také segmentace na věty není dokonalá (v některých segmentech jsou jen části vět a naopak některé segmenty obsahují více vět; přesto v dalším budeme tyto segmenty pro jednoduchost nazývat „věty“). Proto je pravděpodobně úspěšnost provedené anotace nižší.

Použití CDG parseru, zpracování vstupů a výstupů

Vstupem pro CDG tokenizer je čistý text, takže jeho příprava byla jednoduchá. Stačilo zdrojová data ve formátu XML převést do textových souborů s jednou větou v každém odstavci, díky čemuž tokenizer zachoval rozdělení na věty.

Vstupem pro sadu skriptů, která spouští parser, je taktéž jednoduchý textový formát *lattice* (.lat, .cdg), pro jehož generování z výstupu tokenizeru jsem použil nástroje připravené CDG teamem. Tyto skripty jsou na přiloženém CD v adresáři `tools/cdg-scripts`. Samotný parser jsem spustil přímo v Hamburku na clusteru NatS, takže jsem měl k dispozici výpočetní sílu deseti strojů.

Výstupem CDG parseru je formát *annotation* (.ann) založený na XML, ten jsem, abych dosáhl jednotného zpracování převedl skriptem `ann2amw.pl` (v adresáři `tools/my-scripts` do formátu PML. Pro syntaxi německých dat jsem vytvořil PML schéma `adata_schema_de.xml`, které je stejně jako u anglické varianty analogické s českým schématem, pouze zachycuje analytické funkce, které přiřazuje CDG parser.

Nevýhodou anotace CDG parserem jsou některé odlišné konvence, například interpunkční znaménka jsou typicky zavěšena přímo na kořen stromu aj. Úprava stromů tak, aby odpovídaly konvencím Pražského závislostního treebanku, je mimo rámec této práce, avšak pro další využití těchto paralelních dat by byla velmi užitečná.

3.2.3 Anotace češtiny

Pro anotaci českého textu jsem použil metody projektu TectoMT Zdeňka Žabokrtského pro překlad mezi angličtinou a češtinou s transferem na úrovni tektogramatické roviny. Tento projekt používá pro anotaci češtiny tyto nástroje:

- Tokenizace (původní)
- Morfologický slovník
- Disambiguace morfologie [Hajič98]
- McDonaldův parser

3.3 Shrnutí

Výsledkem přípravy dat jsou velmi kvalitní data pro jazykový pár čeština-angličtina, což je dáno vysokou kvalitou dat z Penn Treebanku v PCEDT. Obsahují doslovné překlady vět, které jsou v korespondenci 1 : 1, s ručním alignmentem, ruční syntaktickou anotací na anglické straně a automatickou syntaktickou anotací na české straně.

Česko-německá data jsou méně kvalitní, což je dáno kvalitou původních textů a automatickým alignmentem. Byly vybrány taktéž jen věty v korespondenci 1 : 1. Syntaktická anotace je provedena na obou jazycích automaticky kvalitními parsery s vysokou úspěšností. Jde nejspíše o první paralelní česko-německá data se syntaktickou anotací.

Česko-anglická i česko-německá data jsou uložena jednotně ve formátech pro morfologickou a analytickou anotaci definovaných jazykem PML. Alignment je uchován implicitně shodou identifikátorů vzájemně ekvivalentních vět.

Kapitola 4

Vytvoření EBMT systému

Cílem této práce je vytvořit jednoduchý systém strojového překladu založeného na příkladech a ukázat, nakolik pomůže využití lingvistických informací v tomto systému, především ve fázi matchingu. V této kapitole popíšeme vytvořený výchozí systém („baseline“) bez použití lingvistických informací a metody jeho hodnocení.

4.1 Metody jednotlivých fází překladu

Vytvoření EBMT systému spočívá v předzpracování dat a navržení a implementování jednotlivých fází popsaných v kapitole 2. Níže tedy popíši použité algoritmy a naznačím jejich implementaci. Kompletní zdrojové kódy výchozího systému lze nalézt na přiloženém CD v adresáři `baseline`.

4.1.1 Předzpracování dat

Před napsáním samotného systému bylo třeba ještě provést dvojí předzpracování dat:

- vytvoření indexu výskytu lemmat ve větách
- extrakce slovníků

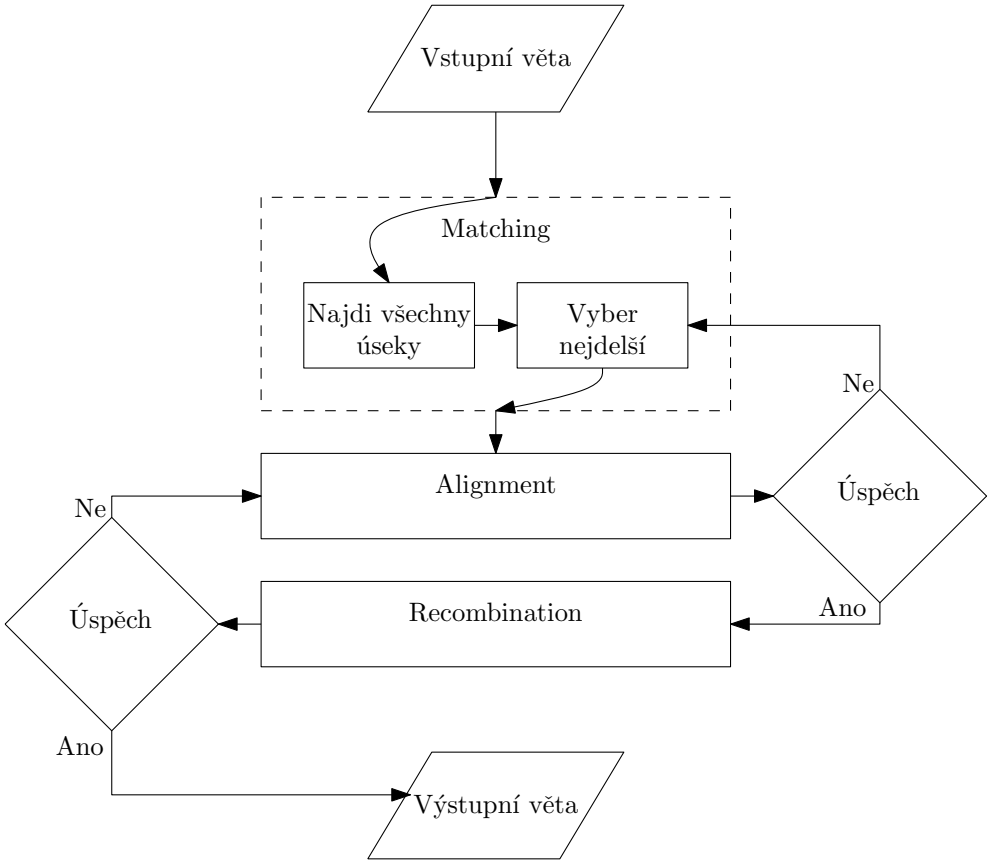
První z nich slouží při fázi matchingu. I ve výchozím systému totiž pracuji s lemmaty a ne se slovními tvary. Kvůli bohaté morfologii češtiny a relativně malé překladové paměti by systém pracující na jednotlivých slovních tvarech jistě dosahoval velmi slabých výsledků, a byl by proto nevhodný pro porovnání. Vytvořený index je velmi jednoduchý, pro každé lemma je v něm uložen seznam identifikátorů vět, ve kterých se dané lemma vyskytuje, a pozice v těchto větách. Jeho využití vyložím níže v rámci popisu implementace matchingu. Skript pro jeho generování (`lemma_index.btred`) lze nalézt na přiloženém CD v adresáři `tools/my-scripts`.

Extrakce slovníku zase slouží ve fázi alignmentu. Pro angličtinu jsou sice dostupné slovníky, avšak potíž je s němčinou. Aby měly překlady na obou jazykových párech co nejpodobnější podmínky, provedl jsem extrakci i pro angličtinu. Téma extrakce slovníku přesahuje rozsah této práce, proto jsem ji provedl velmi

jednoduchým způsobem založeném na absolutní frekvenci slov v textu a na frekvenci společného výskytu páru slov v ekvivalentním páru vět. Skript pro extrakci slovníku lze taktéž nalézt na přiloženém CD (soubor `extract_lemmas_dict.pl` v adresáři `tools/my-scripts`). Použití nějaké sofistikované metody by jistě vylepšilo výstupy celého systému a je tedy na předním místě seznamu úkonů pro budoucí vylepšení.

4.1.2 Schéma systému

Přehledné schéma celého systému ukazuje obrázek 4.1. Je z něj patrno, že fáze alignment a recombination mají možnost vrátit výpočet do předchozího kroku, tedy do matchingu, respektive alignmentu. Tato možnost je využívána pouze v alignmentu v případě, že se nepodaří sestavit dobře ohodnocené ekvivalenty vstupních úseků. Podrobněji se k tomu vrátíme níže při popisu alignmentu. Recombination uspěje vždy, takže k návratu výpočtu do fáze alignmentu nedojde nikdy. Avšak tato možnost může pomoci při budoucím vylepšování systému.



Obrázek 4.1: Schéma výchozího systému

4.1.3 Matching

Pro matching používám výše popsaný postup – tzv. „partial matching for coverage“. Vstupní větu se tedy snažím celou pokrýt podobnými úseky vět z překladové paměti. Jako dostatečně podobné úseky beru takové, jež mají shodná lemmata slov, z nichž se skládají. Vidíme tedy, že i výchozí systém používá nějakou lingvistickou informaci, to však považuji kvůli bohaté české morfologii za nezbytné pro dosažení přijatelné kvality překladu.

Samotné pokrytí probíhá ve dvou fázích. Nejprve algoritmus nalezne v překladové paměti všechny podobné úseky, které překladová paměť poskytuje. V druhé fázi z nich hladově vybírá pokrytí vstupní věty s preferencí delších úseků před kratšími.

Zastavme se ještě u první fáze, tedy nalezení všech podobných úseků. Pro něj jsem navrhl a implementoval tento efektivní algoritmus:

1. Pro každé lemma vstupní věty najdi všechny jeho výskyty v překladové paměti. Pro tento krok se právě použije výše zmíněný předem generovaný index lemmat.

2. Projdi všechny nalezené výskyty všech lemmat a u každého označ, zda následník ve větě v překladové paměti má shodné lemma jako následník odpovídajícího slova ve vstupní větě. U každého výskytu tedy přibyla informace, zda má (1) či nemá (0) následníka.

3. Nyní změníme číslo určující, zda má následníka, na počet následníků. Znovu projdi všechny nalezené výskyty, tentokrát v pořadí od konce vstupní věty. Pokud má výskyt V následníka N, přičti k počtu následníků výskytu V počet následníků následníka N.

Tím dostáváme všechny úseky v podobě dvojice (lemma v překladové paměti, počet následníků). Pro každý nejdelší možný úsek množina obsahuje také všechny jeho podúseky vzniklé odtržením libovolného počtu slov zleva. To není na škodu, jen je třeba na to pamatovat při dalším zpracování, tedy vybírání pokrytí.

Vybrat pokrytí z této množiny je už snadné. Vždy vybereme nejdelší možný úsek a patřičně ořežeme úseky, se kterými se překrývá. Výsledné pokrytí můžeme předat další fázi – alignmentu. Ještě pro přehlednost shrneme vstup, výstup a datové zdroje fáze matchingu:

- Vstup: Věta ve zdrojovém jazyce určená pro překlad.
- Výstup: Množina úseků – čtveřic (i_{IN}, V_{SL}, i_V, n), kde

  i_{IN} je pozice prvního slova úseku ve vstupní větě,
  V_{SL} je věta ve zdrojovém jazyce z překladové paměti v níž byl úsek nalezen,
  i_V je pozice prvního slova úseku ve větě v překladové paměti a
  n je délka (počet slov) nalezeného úseku

- Datové zdroje: věty z překladové paměti ve zdrojovém jazyce, index výskytů lemmat v překladové paměti

4.1.4 Alignment

Alignment je založen na extrahovaném slovníku. Pro každý úsek (i_{IN}, V_{SL}, i_V, n) vrácený matchingem se hledá překlad pro všechna jeho překládají se slova – pro každé slovo úseku se hledá nejlepší překlad v rámci ekvivalentní věty k větě V_{SL} v cílovém jazyce (V_{TL}). Pro nalezení nejlepšího překladu se používá ohodnocení, které k danému překladu vrací slovník. Toto ohodnocení se provede pro všechny možnosti překladu úseku v rámci věty V_{TL}, zvýhodní se překlady složené ze souvislých sekvencí slov a vybere se překlad s nejvyšším takovým ohodnocením.

Pokud je ohodnocení nižší než určená mez (pro tento úsek program nedokáže najít uspokojivý překlad), tak se výpočet vrátí do matchingu, konkrétně do hledání pokrytí s tím, že se tento úsek vypustí z množiny úseků, z nichž se pokrytí vybírá. Pokud jsou všechny úseky přeloženy s ohodnocením vyšším než mez, pokračuje se rekombinací.

Významnou slabinou alignmentu je, že počet slov přeloženého úseku nemůže být vyšší než počet slov originálu. Ke snížení počtu slov dojde také jen poměrně vzácně – a to v případě, že se pro dvě různá slova úseku najde jako překlad jedno slovo.

Nakonec opět stručně shrňme vstup, výstup a datové zdroje alignmentu:

- Vstup: výstup matchingu
- Výstup: čtveřice z výstupu matchingu obohacené o překlady úseků
- Datové zdroje: slovník

4.1.5 Recombination

Recombination je v tomto experimentálním systému nejjednodušší fází a je také fází, na které je pro budoucí vylepšení systému hodně co zlepšovat. Implementace nějaké pokročilé metody rekombinace by však dalece překročila rámec této práce a navíc je rekombinace literaturou dosti opomíjená část EBMT překladu. Z těchto důvodů rekombinaci v systému představuje prosté poskládání výstupních úseků alignmentu za sebe stejně, jako byly jejich ekvivalenty naskládány ve vstupní větě.

- Vstup: výstup alignmentu
- Výstup: věta v cílovém jazyce, úseky přeložené alignmentem naskládány za sebou

Jednou možností, jak tuto slabinu odstranit, by bylo natrénovat jazykový model cílového jazyka a maximalizovat pravděpodobnost výsledné věty přes různá pořadí přeložených úseků nebo přímo jejich jednotlivých slov.

4.2 Metody hodnocení kvality překladu

Abychom ukázali, zda lingvistická informace zvyšuje kvalitu překladového systému, potřebujeme nějaké měřítko této kvality. Překladový systém je samozřejmě tím lepší, čím lepší překlad generuje. Jako měřítko kvality systému nám tedy poslouží hodnocení kvality překladu, který generuje. Metody hodnocení kvality překladu můžeme rozdělit do dvou hlavních skupin:

- hodnocení lidmi
- automatické hodnocení

Lidským ohodnocením se budeme zabývat v následující podkapitole, různými automatickými metodami v dalších podkapitolách této části.

4.2.1 Lidské ohodnocení

Lidské hodnocení bývá co do kvality nadřazováno hodnocení automatickému. To je pochopitelné z toho důvodu, že výstupy strojového překladu mají sloužit lidem a právě lidé mají nejvyšší kompetenci posoudit, co je pro ně dobré (užitečné) a co ne.

Hodnocení strojového překladu lidskými anotátory se zpravidla provádí po větách. Anotátor ohodnotí každou větu dvěma hodnotami, typicky v rozsahu 1–5. První z nich určuje srozumitelnost věty (*fluency*, *intelligibility*), druhá hodnotí věrnost překladu (*adequacy*, *fidelity*). Srozumitelnost vyjadřuje, nakolik je věta plynulá, gramatická, věrnost vyjadřuje, nakolik má přeložená věta stejný význam jako věta původní. Anotátoři dostanou podrobné informace, jak mají tyto hodnoty přiřazovat.

V této práci však tento tradiční způsob lidského hodnocení nepoužijeme. Důvodem jsou jeho vysoké nároky na práci anotátorů. Naším cílem je určit, zda přidání lingvistické informace do systému pomáhá, a k tomu nám poslouží i jednodušší metodika hodnocení – binární porovnání. Ta je důkladně popsána v článku [Vilar07]. Myšlenka této metody je velmi jednoduchá, anotátor dostane původní větu a její překlady dvěma porovnávanými systémy. Z těchto dvou překladů vybere ten lepší anebo je označí jako rovnocenné. Počty výher každého systému tvoří jejich skóre. Zbývá určit, jaký výsledek vypovídá o tom, že je jeden ze systémů významně lepší než druhý. Výše uvedený článek obsahuje odvození, z něhož vyplývá, že pokud platí nerovnost

$$\left|\frac{x-y}{m}\right| > \frac{1,96}{m-1}\sqrt{x+y-\frac{(x-y)^2}{m}},$$

kde
m je počet hodnocených vět,
x je počet „výher“ systému X a
y je počet „výher“ systému Y,
pak s 95% pravděpodobností je systém s lepším skóre významně lepší než druhý.

Tento způsob hodnocení jsem použil pro vyhodnocení chování systémů na obou jazykových párech. Anotační práce ochotně odvedli Bc. Martina Špaková (němčina i angličtina) a Martin Tkáč (angličtina), pro které to byla první anotátorská zkušenost. Jejich znalost jazyka je však na velmi dobré úrovni (Martina Špaková – odborná státní jazyková zkouška z němčiny, německé maturitní vysvědčení, praktická zkušenost s překladem odborného textu, First Certificate of English; Martin Tkáč — Certificate of Advanced English). Z toho důvodu byla použitá metodika hodnocení velmi vhodná, jednak naplňuje požadavek porovnání výchozího a obohaceného systému a jednak mohla být provedena bez podrobných anotačních instrukcí.

4.2.2 BLEU

Hlavním reprezentantem automatických metod hodnocení kvality překladu je metrika BLEU (Bilingual Evaluation Understudy). BLEU byla jedna z prvních automatických metrik, které vykazují vysokou korelaci s lidským hodnocením kvality. Byla vyvinuta v roce 2001 výzkumným centrem IBM a představena v roce 2002 [Papineni02]. V současnosti je nejpopulárnější metrikou pro hodnocení strojového překladu. Je založena na myšlence, že strojový překlad je tím lepší, čím je bližší profesionálnímu lidskému překladu.

Metrika počítá skóre pro jednotlivé segmenty (typicky věty) a následně průměruje tato skóre přes celý korpus. Bylo ukázáno, že vysoce koreluje s lidským hodnocením, avšak pouze na úrovni celého korpusu, tedy řádově na tisících vět. Při ohodnocování jednotlivých vět je korelace podstatně nižší. Kvalitu překladu v BLEU reprezentuje číslo od nuly do jedné a toto číslo vyjadřuje statistickou podobnost k sadě kvalitních lidských referenčních překladů. Nebere tedy přímo v potaz význam, srozumitelnost ani gramatickou správnost.

BLEU funguje na principu měření počtu společných výskytů n-gramů v ohodnocovaném překladu a sadě referenčních překladů. Doporučuje se, aby byla použita sada alespoň čtyř různých nezávislých referenčních překladů. Zjednodušeně lze popsat algoritmus následovně. Nejprve se spočítá počet unigramů v přeloženém segmentu, které se vyskytují alespoň v jednom z referenčních překladů, ku počtu všech unigramů v přeloženém segmentu (p_1). Poté spočte analogické poměry pro bigramy (p_2), trigramy (p_3) a tetragramy (p_4), ze všech čtyř hodnot spočte geometrický průměr a ten vrátí jako výsledek.

Popis algoritmu v předchozím odstavci však není zcela přesný, BLEU řeší ještě dva jeho nedostatky. Prvním z nich je nesprávné opakování správných n-gramů v přeloženém segmentu. Tento nedostatek si ukážeme na příkladu. Mějme dva referenční překlady

(1) the cat is on the mat

(2) there is a cat on the mat

a přeložený segment (výsledek strojového překladu)

(3) the the the the the the the

Počet společných unigramů přeloženého segmentu a referenčních překladů je sedm, počet všech unigramů v přeloženém segmentu je sedm, tedy platí, že $p_1 = \frac{7}{7} = 1$. Intuitivně je však zřejmé, že dát výslednému překladu plné ohodnocení ze shody na unigramech je v tomto případě nežádoucí. BLEU nabízí následující řešení: pro každý n-gram si určí maximální počet výskytů v přeloženém segmentu, kolik se jich ještě započítá jako shodných. Toto číslo se stanoví jako maximální počet výskytů v jednotlivých referenčních překladech. V našem příkladu se unigram „the“ vyskytuje nejvíckrát v referenčním překladu (1), a to dvakrát (tedy $p_1 = \frac{2}{7}$, nikoliv 1).

Dalším problémem je, že samotný počet společných n-gramů (precision) nestačí. Ukážeme si jej opět na příkladu. Při referenčních překladech (1) a (2) hodnoťme překlad

(4) on the

Tento překlad má $p_1 = p_2 = 1$ i přesto, že neobsahuje vlastně žádnou informaci. BLEU se snaží tento problém řešit znevýhodněním stručnosti. Pokud je přeložený segment kratší než referenční překlad, který se mu délkou nejvíce blíží, pak se aplikuje toto znevýhodnění – spočte se rozdíl jejich délek r a výsledný geometrický průměr se vynásobí číslem $e^{(1-r)}$.

Hodnocení BLEU je implementováno v skriptu, který vydává NIST (National Institute for Standards and Technology). Hodnoty BLEU i NIST, které dále uvádím jsou hodnoty získané verzí 11b tohoto skriptu. Použité příklady v popisu algoritmu jsou z učebního materiálu Michaela Collinse dostupného na jeho webové stránce[1].

4.2.3 NIST

Metrika NIST je založena na stejné myšlence jako BLEU, tedy na porovnávání n-gramů. Od BLEU se odlišuje v tom, že neoceňuje všechny shodné výskyty stejně, n-gramy, které se v textu často vyskytují, mají na výsledek nižší vliv, než n-gramy, které se v textu vyskytují málo často. Účelem je, aby málo informativní n-gramy (jako například bigram „on the“ z minulého příkladu) lacině nevylepšovaly skóre.

Druhou odlišností je výpočet znevýhodnění za stručnost, drobné odlišnosti v délce segmentu nejsou tak silně postiženy.

4.2.4 METEOR

Metoda METEOR [Lavie05] je mladší a funguje jinak. Je spočtena jako harmonický průměr hodnot precision a recall na unigramech. Vyžaduje jeden nebo více referenčních překladů. Pokud je k dispozici referenčních překladů více než jeden, spočte se hodnota nezávisle pro každý z nich a vybere se z nich maximum.

[1] `http://people.csail.mit.edu/mcollins/teaching.html`

Algoritmus tedy dostane jako vstup dvě věty, strojový překlad a referenci. Prvním krokem je vytvoření tzv. alignmentu – vzájemného provázání unigramů vstupních vět. Toto provázání si můžeme představit jako bipartitní graf (V_1, V_2, E), kde množiny V_1 a V_2 tvoří unigramy přeložené, respektive referenční věty. Každý unigram může být spojen hranou s nejvýše jedním unigramem v protější větě, tedy hodnost každého vrcholu je nejvýše 1.

Provázání se vytváří v několika fázích, účelem je spojit slova, která si vzájemně odpovídají. Tyto fáze jsou tedy různé algoritmy matchingu. V první fázi se nejprve spojí hranou všechny vzájemně odpovídající unigramy a poté se vybere podgraf s maximálním počtem hran tak, aby hodnost každého vrcholu byla nejvýše 1. Pokud existují dva různé grafy se stejným počtem hran, vybere se ten s nižším počtem zkřížených hran. Další fáze jsou analogické, avšak každá další fáze už pracuje jen s takovými vrcholy, jejichž hodnost je 0. Jednotlivé fáze – algoritmy matchingu jsou proto seřazeny od nejpřísnějších k benevolentnějším.

Po konstrukci provázání se spočte precision (P), recall (R) a jejich harmonický průměr (F_{mean}) podle následujících vzorců:

$$P = \frac{e}{v_t} \qquad R = \frac{e}{v_r} \qquad F_{mean} = \frac{10PR}{R + 9P}$$

kde e je počet hran a v_t a v_r jsou počty unigramů (vrcholů) v hodnoceném překladu resp. v referenci.

Spočtený harmonický průměr je kandidátem na výsledné skóre. Avšak do výsledného skóre se ještě podepíše informace o tom, jak je na tom překlad s delšími segmenty než unigramy. Konkrétně se skóre penalizuje vynásobením číslem $(1-p)$, spočteném následujícím způsobem.

Nejprve se jednotlivé unigramy sloučí do delších fragmentů, tzv. chunků. Jeden chunk obsahuje takové unigramy, které jsou sousední v překladu i v referenci. Potom penalizaci p určuje tento vzorec:

$$p = 0,5(\frac{c}{e})$$

kde c je počet chunků a e je počet hran.

Výsledné skóre METEOR je tedy

$$METEOR = (1 - p)F_{mean}$$

Míra METEOR vykazuje dle autorů vyšší míru korelace s lidským ohodnocením překladu. Zatímco BLEU údajně na úrovni korpusu dosahuje korelace 81,7 %, METEOR dosahuje až 96,4 %.

Skript počítající METEOR je volně dostupný na webu[2]. Při hodnocení systému v této práci budu uvádět hodnoty BLEU, NIST i METEOR, ale pro odladění parametrů systému používám pouze populární hodnotu BLEU.

[2]http://www.cs.cmu.edu/~alavie/METEOR/

4.3 Vyladďování parametrů systému

Kvalitu překladu výchozího systému můžeme ovlivnit dvěma parametry. Oba ovlivňují alignment, protože matching i recombination jsou jednoznačné algoritmy bez parametrizace.

Prvním z nich je minimální výška ohodnocení překladu úseku, který je ještě uspokojivý, který tedy nevrátí výpočet do fáze matchingu. Druhým parametrem je koeficient, jímž se vynásobí (a tím zvýhodní) ohodnocení souvislého překladu úseku. Tyto dva parametry jsem se pokusil nastavit tak, aby hodnocení kvality překladu bylo co nejvyšší.

Jelikož překlad vět vyvinutým systémem je dosti časově náročný, vybral jsem pro vyladění těchto parametrů jen část z development dat, a to konkrétně soubory `wsj_2332`, `wsj_2393`, `wsj_2435` a `wsj_2436`. Ty dohromady obsahují přibližně 40 vět. Jak je uvedeno výše, hodnocení BLEU je relevantní až u daleko vyššího počtu vět, proto jsem ho bral jen jako orientační, a tudíž jsem ani parametry nenastavoval s neadekvátně velkou přesností. Pokusů jsem provedl celkem 16 – pro čtyři různé hodnoty prvního parametru a čtyři různé hodnoty druhého parametru.

Hodnoty prvního parametru, tedy minimálního ohodnocení uspokojivého překladu, jsem zvolil následujícím způsobem. Nejprve jsem nechal proběhnout systém bez omezení tímto parametrem a nechal jsem vypsat ohodnocení všech úseků použitých pro překlad. Z těchto ohodnocení jsem vybral horní kvartil, medián (prostřední kvartil) a dolní kvartil a použil jsem je spolu s nulou jako hodnoty parametru. Podle hodnoty parametru tedy projde buď jen čtvrtina nejlépe hodnocených překladů úseků, nebo polovina, tři čtvrtiny, nebo konečně úplně všechny překlady bez omezení.

Hodnoty druhého parametru jsem určil odhadem. Nastavil jsem je na 1 (bez zvýhodnění souvislého překladu úseku), 1.5, 2 a 3. Protože první a druhý parametr zajisté nejsou nezávislé, nemohl jsem je podle maximalizace ohodnocení nastavovat postupně, ale musel jsem hodnocení maximalizovat přes všechny kombinace hodnot těchto parametrů. Protože hodnoty jsem pro oba dva zvolil čtyři spustil jsem tedy 16 běhů systému s různými kombinacemi parametrů. Tabulka 4.1 ukazuje označení jednotlivých běhů a tabulka 4.2 jejich hodnocení BLEU a NIST.

Výsledky naznačují, že odhad nastavení druhého parametru byl správný. U jeho nejvyšší hodnoty (3) dojde při libovolném prvním parametru k poklesu hodnocení, dá se tedy předpokládat, že pro vyšší zvýhodnění souvislého překladu úseku by pokles hodnocení kvality překladu pokračoval.

4.4 Vyhodnocení základního systému

Běh Baseline 11 dosáhl nejvyššího hodnocení BLEU, jeho nastavení parametrů jsem tedy vybral jako nejlepší. Tabulka 4.3 ukazuje vyhodnocení systému Baseline 11 pro celou sadu development dat a evaluation dat.

Minimální uspokojivé ohodnocení	Koeficient zvýhodnění souvislého překladu: 1	1,5	2	3
0	Baseline 1	Baseline 2	Baseline 3	Baseline 4
Dolní kvartil	Baseline 5	Baseline 6	Baseline 7	Baseline 8
Medián	Baseline 9	Baseline 10	Baseline 11	Baseline 12
Horní kvartil	Baseline 13	Baseline 14	Baseline 15	Baseline 16

Tabulka 4.1: Označení výchozího systému pro jednotlivá nastavení parametrů

	BLEU	**NIST**	**METEOR**
Baseline 1	0.1444	5.1143	0.4461
Baseline 2	0.1667	5.0970	0.4323
Baseline 3	0.1645	5.0192	0.4278
Baseline 4	0.1612	4.9504	0.4208
Baseline 5	0.1497	5.3330	0.4539
Baseline 6	0.1672	5.3260	0.4537
Baseline 7	0.1650	5.2562	0.4460
Baseline 8	0.1558	5.0395	0.4281
Baseline 9	0.1272	5.0823	0.4455
Baseline 10	0.1745	5.2492	0.4562
Baseline 11	0.1781	5.2481	0.4547
Baseline 12	0.1733	5.2177	0.4447
Baseline 13	0.1260	4.6013	0.4218
Baseline 14	0.1437	4.8719	0.4311
Baseline 15	0.1593	5.0438	0.4500
Baseline 16	0.1755	5.0784	0.4461

Tabulka 4.2: Hodnoty BLEU a NIST pro jednotlivé hodnoty parametrů

	BLEU	**NIST**	**METEOR**
Development data	0.1667	5.9655	0.4305
Evaluation data	0.1487	5.7307	0.4327

Tabulka 4.3: Vyhodnocení Baseline 11 na celých sadách dat.

Kapitola 5

Vylepšování lingvistikou

Tato kapitola se zabývá obohacením výchozího systému o využití lingvistických informací obsažených v připravených datech. Tato obohacení rozdělíme pro přehlednost do skupin podle fáze překladu, ve kterém se informace využije, a podle typu lingvistické informace, která se v dané fázi použije. Celkem tedy dostaneme šest skupin

- využití morfologie v matchingu
- využití syntaxe v matchingu
- využití morfologie v alignmentu
- využití syntaxe v alignmentu
- využití morfologie v rekombinaci
- využití syntaxe v rekombinaci

5.1 Využití morfologie v matchingu

Při vyhledávání nejlépe odpovídajících úseků pokrývajících vstupní větu vezmeme po délce shody na lemmatech jako druhé hodnotící kritérium jejich morfologickou shodu. Tu zjistíme porovnáním morfologických značek jednotlivých slov úseku ve vstupní větě a v překladové paměti. V systému je implementována velmi jednoduchá metoda. Předpokládá totiž, že všechny znaky tvořící morfologickou značku libovolného slova mají pro morfologii stejný význam. Z úseků dané délky (kandidátů) se tedy vybere takový, jenž se od odpovídajícího úseku vstupní věty liší v nejmenším počtu znaků morfologických značek všech jeho slov.

Tato jednoduchá metoda by samozřejmě mohla být vylepšena diferenciací relevance jednotlivých znaků v morfologických značkách. Například slovní druh je pro význam věty (a tedy správný překlad) zajisté důležitější, než například pád.

Na druhou stranu, pokud se slova odlišují ve slovním druhu, pak se to velmi pravděpodobně podepíše i na ostatních znacích v morfologické značce, takže i tato jednoduchá metoda penalizuje tento rozdíl více. Navíc hodnocení relevance

jednotlivých znaků morfologických značek by do systému vneslo obrovský počet parametrů, jejichž optimalizace by si vyžádala neúnosné množství výpočetního času. Spokojíme se tedy s jednoduchou implementací, ostatně hodnocení na konci kapitoly ukáže, že pro zlepšení kvality překladu poslouží i ona.

5.2 Využití syntaxe v matchingu

Zabýváme se vylepšováním lexikálního výchozího systému lingvistickou informací; z toho důvodu nebudeme rozebírat metody matchingu na syntaktických stromech. Použití těchto metod by totiž nebylo vylepšením, nýbrž kompletním pozměněním principu systému.

Postupujme tedy obdobně jako při užití morfologie. Nechme jako výchozí kritérium délku shody na lemmatech a nalezené úseky zvýhodňujme nebo penalizujme na základě syntaktických vlastností.

Zvýhodnění je žádoucí v případě, že mezi jednotlivými slovy úseku ve vstupní větě jsou přímé syntaktické vazby a tyto vazby jsou shodné i v odpovídajícím úseku nalezeném v překladové paměti. Penalizace je naopak vhodná tehdy, když slova úseku ve vstupní větě nemají žádnou syntaktickou spojitost. V takovém případě dokonce vyvstává otázka, zda je daný úsek vůbec vhodné použít pro překlad. Jednotlivá slova jsou v tomto úseku víceméně shodou okolností. Kratší úseky, které by třeba pro některá z těchto slov mohly zachytit syntaktický kontext, by byly pro použití při překladu adekvátnější.

Výchozí systém tedy obohatíme následovně. Z nalezených úseků pro překlad se použijí jen takové, které tvoří ve vstupní větě i ve větě v překladové paměti souvislý podgraf syntaktického stromu.

Využití lingvistické informace v alignmentu a v rekombinaci nejsou součástí zadání této práce. Proto se nestanou součástí implementace systému, ale v dalším textu alespoň stručně naznačíme, jaké možnosti v těchto fázích lingvistické informace nabízí.

5.3 Využití morfologie v alignmentu

Morfologická informace by mohla pomoci jako doplněk k výše navržené slovníkové metodě. Skóre dvojice slov ve zdrojovém a v cílovém jazyce by tedy záviselo nejen na pravděpodobnosti vrácené slovníkem, nýbrž i na morfologické shodě. Protože dvojice jazyků, na nichž se provádí překlad (čeština-angličtina, čeština-němčina) jsou dvojice jazyků typologicky odlišných a parsery vrací popis lingvistických informací v odlišných konvencích, bylo by třeba vytvořit nějaký druh mapování shodných morfologických vlastností.

Velkým přínosem by mohlo být přinejmenším užití informace o slovním druhu (ač spíše sémantická kategorie, je slovní druh součástí morfologické značky, technicky je jeho využití tedy využití morfologie).

5.4 Využití syntaxe v alignmentu

Využití syntaxe v alignmentu by mohlo kvalitu překladu významně pozvednout. Navržená a implementovaná slovníková metoda je velmi jednoduchá a má mnoho nedostatků. Ideální by bylo doplnit použité korpusy informací o frázovém alignmentu a pro jeho extrakci využít syntaktické informace dostupné pro všechny jazyky.

To by přineslo obohacení zdrojových dat, které by mohlo být využito i v jiných projektech založených na těchto datech.

5.5 Využití morfologie v rekombinaci

Morfologie by šla v rekombinaci využít definováním a použitím pravidel pro mapování morfologie mezi zdrojovým a cílovým jazykem. Kdyby se v použitém segmentu morfologie neshodovala, mohla by se zpětně změnit v cílovém jazyce při rekombinaci.

Tímto zásahem by se systém přiblížil systémům pravidlovým, úprava by si s sebou také nesla nevýhody takových systémů – zbudování univerzální databáze těchto pravidel by bylo jistě velmi časově náročné a také nejisté.

5.6 Využití syntaxe v rekombinaci

Syntaxi by šlo v rekombinaci použít analogickým způsobem jako morfologii. Lišil by se účel tohoto využití, zatímco morfologie by mohla pomoci najít a použít správné tvary slov, využití informace o syntaxi by mohlo vylepšit slovosled.

5.7 Implementace navržených úprav

Úpravy navržené v podkapitolách o využití morfologie a syntaxe v matchingu jsou v systému implementovány. Používání lingvistických informací je však velmi náročné na výpočtový čas, a proto implementace vyžaduje optimalizace.

Především je důležité zamezit zbytečnému opakování již provedených výpočtů, ke kterému by mohlo dojít při iterování po odmítnutí alignmentu, jehož skóre nepřekročí stanovenou mez. Toho je dosaženo uchováváním informací o úsecích v persistentní struktuře; vždy předtím, než se provádí nějaký výpočet týkající se konkrétního úseku, ověří se, zda výsledek už není v této struktuře obsažen.

Druhým faktorem ovlivňujícím dobu výpočtu je stanovení, kdy se mají počítat skóre určující morfologickou a syntaktickou diferenci. V systému je implementována kombinovaná varianta, která se jeví jako nejvhodnější: u všech úseků délky alespoň 2 se tato skóre spočítají ihned v proceduře nalezení všech úseků pokrývajících vstupní větu. Pole všech úseků se pak seřadí podle tří kritérií, nejprve podle délky úseku, pak podle syntaktické diference a na závěr podle diference morfologické.

V samotném vybírání pokrytí se pak toto pole prochází sekvenčně a vybírají se všechny takové úseky, které jsou disjunktní s již pokrytou částí věty (s aktuálním pokrytím). Když při tomto procházení dojde na úseky délky jedna, dopočítávají se jejich morfologické diference v této fázi (syntaktická diference u úseku délky jedna postrádá význam, jedná se vždy pouze o jeden uzel). Tento výpočet se však provádí pouze u těch úseků, které jsou disjunktní s aktuálním pokrytím. Navíc je použito prořezávání, pokud se najde úsek s nulovou morfologickou diferencí, použije se ihned bez počítání morfologické diference zbývajících úseků.

Systém navíc používá data v rychlém formátu Perl Storable, do kterého jsem je převedl skriptem dostupným na webu PDT.

I přes tyto optimalizace je výpočtová doba velmi vysoká, překlad jedné věty se pohybuje ve velmi nejistém rozmezí od několika sekund do několika hodin, v závislosti na počtu iterací způsobených odmítnutím alignmentu pro nízké skóre.

5.8 Vyhodnocení vylepšeného systému

Pro vyhodnocení vylepšeného systému na angličtině uvádím všechny popsané automatické metriky pro sadu development i evaluation dat. Uvedeny jsou v tabulce 5.1, pro porovnání jsou tamtéž znovu uvedena ohodnocení výchozího systému.

		BLEU	**NIST**	**METEOR**
Vylepšený systém	Development data	0.1758	6.0299	0.4353
	Evaluation data	0.1571	5.8585	0.4366
Výchozí systém	Development data	0.1667	5.9655	0.4305
	Evaluation data	0.1487	5.7307	0.4327

Tabulka 5.1: Ohodnocení systému automatickými metrikami

Navíc jsem provedl díky pomoci anotátorů binární porovnání systémů způsobem popsaným v kapitole 4.2.1. Anglickou sadu evalution dat ohodnotili dva anotátoři, jejich hodnocení ukazuje tabulka 5.2. Z ní je patrno, že rozdíl mezi systémy je minimální; nerovnost popsaná v kapitole 4.2.1 neplatí pro výsledek anotace žádného z nich, a tudíž není potvrzen významný rozdíl mezi systémy. To potvrzuje dojem samotných anotátorů, dle nichž byl rozdíl mezi posuzovanými překlady nepatrný. Detailní výsledky hodnocení jsou na přiloženém CD v adresáři `outputs`.

Anotátor	**Počet „výher“ výchozího systému**	**Počet „výher“ vylepšeného systému**	**Počet „remíz“**
Martina	124	97	35
Martin	101	114	41

Tabulka 5.2: Ohodnocení systému anotátory (angličtina)

Pro německá data nejsou dostupné sady referenčních překladů, takže bylo provedeno pouze lidské ohodnocení. Anotátorka porovnala dvojice překladů vět ze sady development dat, jimiž nebylo nijak ovlivněno chování systému. Výsledky anotace ukazuje tabulka 5.3. Systém obohacený lingvistikou má paradoxně nepatrně horší výsledek než výchozí systém, nerovnost z kapitoly 4.2.1 ani v tomto případě nepotvrzuje statisticky významný rozdíl mezi systémy. Zajímavou odlišností je velký počet „remíz“ na německých datech, ten je dán pravděpodobně jejich menším rozsahem – to zvyšuje pravděpodobnost, že obě varianty systému použijí stejné pokrytí věty.

Anotátor	**Počet „výher“ výchozího systému**	**Počet „výher“ vylepšeného systému**	**Počet „remíz“**
Martina	29	40	50

Tabulka 5.3: Ohodnocení systému anotátorkou (němčina)

Kapitola 6

Závěr

6.1 Výsledky a přínosy

Práce ukázala, že využití lingvistiky vede k malému zlepšení výsledků automatických metrik kvality překladu lexikálním EBMT systémem. Toto zlepšení se však nepotvrdilo při lidském ohodnocení překladu. Tam je rozdílnost v hodnocení výchozího a vylepšeného systému tak malá, že není statisticky průkazná. Automatické metriky se použily při hodnocení kvality překladu z češtiny do angličtiny, lidské hodnocení bylo využito u překladů mezi oběma jazykovými páry, tedy z češtiny do angličtiny i z češtiny do němčiny. Výsledky lidského hodnocení jsou na obou jazykových párech stejně neprůkazné, kvalita německých překladů je výrazně nižší, což je způsobeno nízkou kvalitou segmentace německých dat a neúplnou lemmatizací.

Přestože práce neukázala jasný přínos přidání lingvistické informace, myslím si, že může být užitečná. Její přínosy jsou především:

- Vytvořený překladový systém. Kvalita překladu je sice poměrně nízká, ale otevírá prostor pro další vylepšení, některé z možností nastíníme v příští podkapitole.
- Sběr a anotace německých dat; paralelní korpus, který v rámci této práce vznikl, je jediný dostupný česko-německý paralelní korpus se syntaktickou anotací. Přináší možnost vyzkoušet i jiné již vyvinuté metody překladu (například statistické) i na němčině.
- Podrobné vysvětlení obtížné problematiky vymezení EBMT v rámci ostatních metod překladu.
- Vysvětlení principů evaluačních technik včetně moderní automatické metriky METEOR a jednoduché techniky lidského ohodnocení pro vzájemné porovnání dvou systémů.
- Důležitým negativním výsledkem je zjištění, že kvalita českých dat v současné verzi korpusu OPUS je velmi nízká, doufám, že práce někomu pomůže vyvarovat se použití těchto dat.

6.2 Co zlepšovat?

Práce také poskytuje podněty pro další výzkum. Jsou to především tyto dva:

- další práce na EBMT systému
- úpravy pro zvýšení použitelnosti česko-německých dat
- použití kvalitních slovníků

Slabina vyvinutého systému je hlavně ve fázích alignment a rekombinace, které nebyly hlavním předmětem této práce. Kvalitě překladu by jistě významně pomohlo použití nějaké standardní techniky pro alignment na úrovni slov nebo frází, případně vyvinutí nějaké nové techniky na míru, která by brala v úvahu celou větu obsahující nalezený úsek a nikoliv jen slova tohoto úseku. Pokud matching nalezne nějaký úsek vstupní věty ve více příkladech, měla by také technika alignmentu vzít v úvahu všechny tyto příklady. Pomoci by také mohlo použití lingvistických informací, například pro alignment jmenných frází.

Pro rekombinaci by bylo asi nejúčelnější použití statistického jazykového modelu. Vhodné by bylo též využití mapování morfologie zdrojového a cílového jazyka pro generování správných tvarů slov.

Významné vylepšení výsledků na němčině by jistě přineslo další zpracování vstupních dat. Přestože v práci uvádím pojem „věta“ pro jednotlivé zarovnané segmenty v korpusu, pravdou je, že často se jedná o kratší i delší úseky. Především úseky složené z několika vět byly tvrdým oříškem pro parsery, které na vstupu očekávají pouze jednu větu. Precizní segmentace by data výrazně zkvalitnila.

Dalším problémem je častý výskyt odrážek, číslování, číselně označených entit (např. spisů) apod. Parser němčiny se s nimi vypořádal dobře, ale pro parser češtiny by bylo přínosné tyto anomálie předzpracováním odstranit, protože jistě zapříčiňují vysoké procento chyb parsingu. Příklad této chybné anotace je vidět v příloze A.

Použitý zdroj, JRC-Acquis korpus, je velmi rozsáhlý. Pro účely této práce z něj byla použita jen malá část, jak je popsáno v kapitole 3. Zkvalitnění by jistě přineslo použití většího rozsahu dat, případně s nějakou inteligentní selekcí, která by vyřadila nevhodné páry.

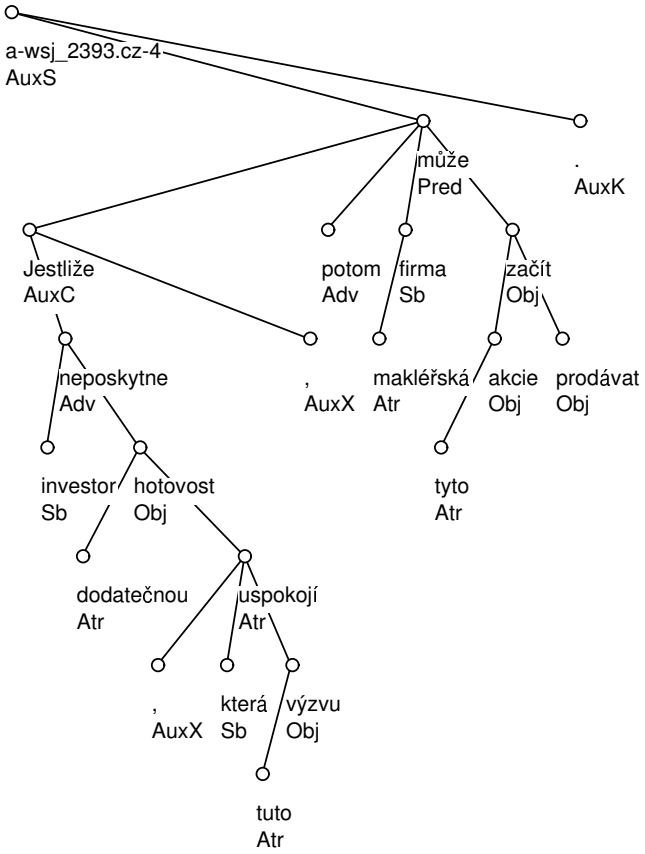
Posledním účinným zpracováním dat by byla úprava výstupu parseru němčiny tak, aby odpovídal (anebo se alespoň podobal) výstupu parseru češtiny. Analogické anotace by se jistě daly dobře využít pro frázový alignment.

Vyšší kvalitu překladu by také zcela jistě přineslo použití kvalitnějších slovníků. V této práci byla navržena a aplikována jednoduchá metoda pro jejich extrakci kvůli jednotnému zpracování dat a tedy následné možnosti porovnání chování systému na jednotlivých jazykových párech.

Všechny uvedené nedostatky systémů i dat a návrhy jejich řešení jsou zajisté dobré podněty pro další výzkum. Ten třeba někdy v budoucnu přinese v rámci možností kvalitní výstupy strojového překladu a snad i efektivní komunikaci bez nedorozumění způsobených jazykovou bariérou.

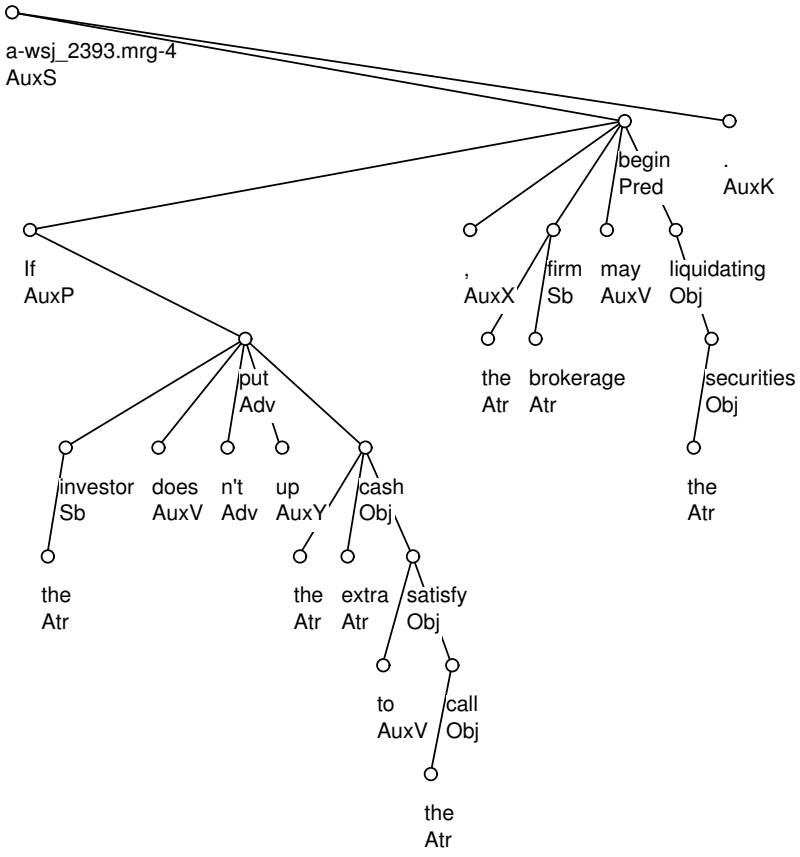
Příloha A

Ukázka anotace v použitých korpusech



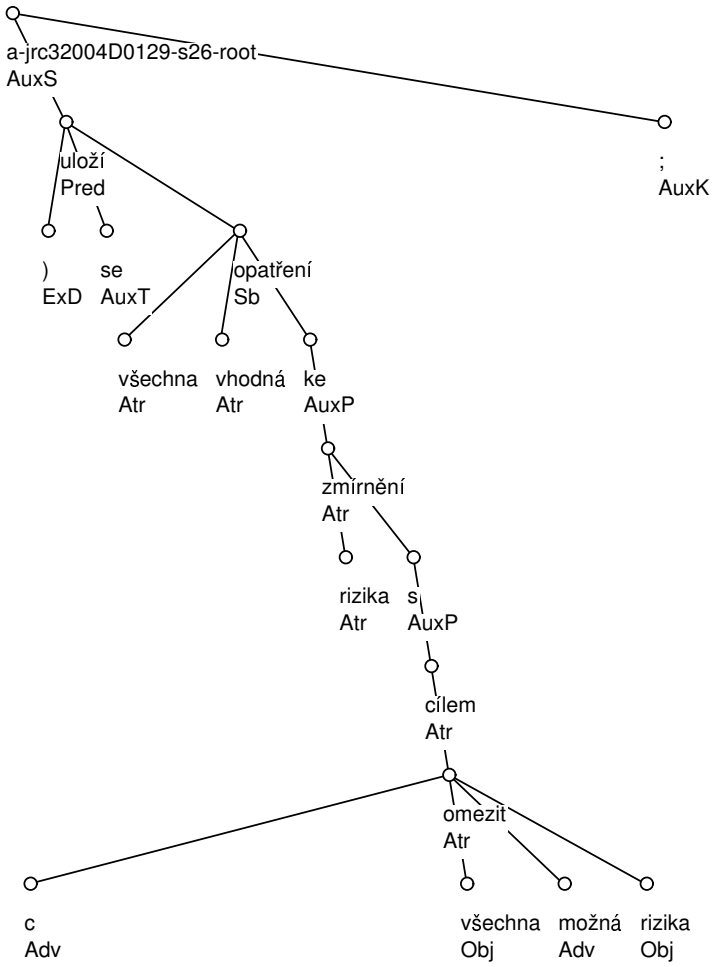
Jestliže investor neposkytne dodatečnou hotovost, která uspokojí tuto výzvu, potom makléřská firma může tyto akcie začít prodávat.

Obrázek A.1: Ukázka syntaktického stromu české věty z PCEDT (automatická anotace Collinsovým parserem)



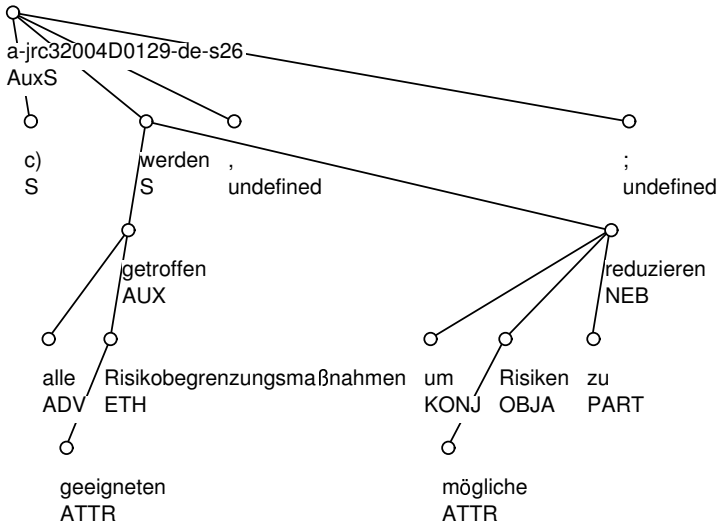
If the investor does n't put up the extra cash to satisfy the call, the brokerage firm may begin liquidating the securities.

Obrázek A.2: Ukázka syntaktického stromu anglické věty z PCEDT



c) uloží se všechna vhodná opatření ke zmírnění rizika s cílem omezit všechna možná rizika;

Obrázek A.3: Ukázka syntaktického stromu české věty (segmentu) z JRC-Acquis



c) alle geeigneten Risikobegrenzungsmaßnahmen getroffen werden, um mögliche Risiken zu reduzieren;

Obrázek A.4: Ukázka syntaktického stromu německé věty (segmentu) z JRC-Acquis

Příloha B

Ukázka výstupu překladových systémů

V této příloze uvedeme některé reálné výstupy výchozího a obohaceného systému pro oba jazykové páry. Pro každou překládanou větu uvedeme její původní znění (dále značené písmenem O – *original*), její překlad tak, jak je uveden ve zdrojových datech (R – *reference*), překlad výchozím systémem (B – *baseline*) a překlad systémem obohaceným lingvistickou informací (E – *enhanced*).

B.1 Překlady z češtiny do angličtiny

Věty jsou ze souborů `wsj_2201` z evaluation dat a `wsj_2303` z development dat.

B.1.1 Věty z evaluation dat

O: Uvidíme , zda reklama funguje .

R: We 're about to see if advertising works .

B: see the whether campaigns work .

E: see whether the campaigns work .

O: Okamžitě po pátečním 190 bodovém propadu akciového trhu a následné nejistotě vypouští několik velkých brokerských firem nové inzeráty vytrubující obvyklé poselství : Pokračujte v investování , trh je v pořádku .

R: Hard on the heels of Friday 's 190-point stock-market plunge and the uncertainty that 's followed , a few big brokerage firms are rolling out new ads trumpeting a familiar message : Keep on investing , the market 's just fine .

B: immediately on Friday 190 Dow Friday fund market and subsequent uncertainty an several big brokerage firms new ad unusual message : continue a investing , market in is fine .

E: immediately on Friday 190 Dow Friday fund market and subsequent uncertainty an several big brokerage firms new ad unusual message : continue a investing the

market in is fine .

O: Jejich úkolem je odradit klienty od útěku z trhu , což jednotliví investoři hromadně činili po propadu v říjnu .

R: Their mission is to keep clients from fleeing the market , as individual investors did in droves after the crash in October

B: its task be discourage client from flight from market , share individual investors en cent after Friday in October

E: its task is discourage client from flight from market the year Individual investors en cent after plunging in October

O: Jen pár dní po propadu v roce 1987 velké brokerské firmy rychle vydaly inzeráty k uklidnění investorů .

R: Just days after the 1987 crash , major brokerage firms rushed out ads to calm investors .

B: only few day after Friday in year 1987 big brokerage firms quickly statement ad “ and investor .

E: only few days after plunging from a 1987 big brokerage firms quickly statement ad “ and investor .

O: Nyní zareagovaly dokonce ještě rychleji .

R: This time around , they ’re moving even faster .

B: now react Yet even quickly

E: now react even quickly .

R: Fidelity Investments placed new ads in newspapers yesterday , and wrote another new ad appearing today .

B: Fidelity Investments new ad into newspaper yesterday and create another new ad , that a appear today .

E: Fidelity Investments new ad into newspaper yesterday creating a another new ad , ” says appear today .

B.1.2 Věty z development dat

O: Konsorcium soukromých investorů fungující jako LJH Funding Co . sdělilo , že dalo nabídku za 409 milionů dolarů v hotovosti na většinu holdingů v oblasti realit a nákupních center firmy L . J . Hooker Corp .

R: A consortium of private investors operating as LJH Funding Co. said it has made a $ 409 million cash bid for most of L.J. Hooker Corp. ’s real-estate and shopping-center holdings .

B: consortium private investors have as Funding Co. , tell said the give offer for 409 a $ million in cash most Holding in area reality and shopping centers firm Hooker Corp. executives .

E: consortium private investors have as Funding Co . tell said the give offer for 409 a $ million in cash most Holding area are reality and shopping centers firm Hooker Corp. executives .

O: Tato 409 milionová nabídka zahrnuje také odhadovaných 300 milionů dolarů v zaručených závazcích na tyto nemovitosti , jak uvádí nabízející strana .

R: The $ 409 million bid includes the assumption of an estimated $ 300 million in secured liabilities on those properties , according to those making the bid .

B: this 409 issue offer also includes estimate $ 300 million in minimum commitment to this estate , how report offer party

E: this 409 issue offer includes also estimate $ 300 million in minimum commitment have to estate , they report offer the other

O: Skupinu vede Jay Shidler , výkonný ředitel Shidler Investment Corp . na Honolulu , a A . Boyd Simpson , výkonný ředitel Simpson Organization Inc . v Atlantě .

R: The group is led by Jay Shidler , chief executive officer of Shidler Investment Corp. in Honolulu , and A. Boyd Simpson , chief executive of the Atlanta-based Simpson Organization Inc .

B: group lead Jay , chief executive Investment Corp. . to , and , L.J. chief executive Simpson Roper Inc. . of Atlanta

E: group lead Jay , chief executive Investment Corp. , for , and , L.J. executive director Simpson Roper of Inc. . Atlanta

O: Firma pana Shidlera se specializuje na investice do obchodních realit a chlubí se majetkem v hodnotě 1 miliardy dolarů ; pan Simpson je developer a bývalý vedoucí pracovník ve firmě L . J . Hooker .

R: Mr. Shidler 's company specializes in commercial real-estate investment and claims to have $ 1 billion in assets ; Mr. Simpson is a developer and a former senior executive of L.J. Hooker .

B: firm Mr. specializes in investment into trade reality and boast with asset of $ 1 billion ; Mr. Simpson be developer and former executive in firm , L.J. Hooker

E: firm Mr. specializes in investment into trade reality and boast with asset of $ 1 billion ; Mr. Simpson be developer and former executive firm in , L.J. Hooker

O: " Aktiva jsou dobrá , ale vyžadují více peněz a řízení " než může L . J . Hooker v současné situaci nabídnout , řekl pan Simpson v jednom rozhovoru . ‘ ‘

R: “ The assets are good , but they require more money and management ” than can be provided in L.J. Hooker 's current situation , said Mr. Simpson in an

interview . “

B: “ asset been better , but require more money and control “ than could , L.J. Hooker in current situation offer , Mr. Simpson in one talk “ .

E: “ asset is good , but require more and money control “ than could Hooker , is currently situation offer “ , Mr. Simpson in one talk “ .

O: Filozofie firmy Hooker byla postavit a prodat .

R: Hooker ’s philosophy was to build and sell .

B: philosophy firm Hooker is set selling and

E: philosophy firm Hooker been built and sold

O: My chceme postavit a ponechat si .

R: We want to build and hold . ”

B: I want build and retain

E: I want build and retain

B.2 Překlady z češtiny do němčiny

Věty jsou ze souboru `jrc32004D0003`

O: Komise

R: der

B: Entscheidung

E: Kommission

O: ze dne 19 . prosince 2003 ,

R: vom 19. Dezember 2003

B: der vom 19. Dezember 2003

E: der vom 19. Dezember 2003

O: kterým se při uvádění sadby brambor na trh na celém území některých členských států nebo na jeho části povoluje používat přísnější opatření proti některým chorobám , než která jsou stanovena v přílohách I a II směrnice Rady 2002 / 56 / ES

R: zur Ermächtigung bestimmter Mitgliedstaaten , für den Verkehr mit Pflanzkartoffeln auf ihrem gesamten oder auf Teilen ihres Hoheitsgebiets strengere als die in den Anlagen i und ii der Richtlinie 2002/56/EG des Rates vorgesehenen Maßnahmen gegen bestimmte Krankheitserreger anzuwenden

B:) bei des Rates Kartoffel Mitgliedstaaten , all bestimmter Hoheitsgebiet Mitgliedstaaten , ihr eines Teil zulassen ab streng Maßnahmen zum bestimmte Krankheiten , als die Anhang der für . , Anhängen Richtlinie mit Pflanzkartoffeln(7) ,

E: und bei des Rates Kartoffel Mitgliedstaaten , all bestimmter Hoheitsgebiet Mitgliedstaaten ist , eines Teil zulassen ab streng Maßnahmen zum bestimmte Krankheiten , als die Anhang der für . und ii die Richtlinie des 2002

O: Dotyčné členské státy zřídí stálý systém pravidelných úředních kontrol určených k tomu , aby podmínek zmocnění byly trvale plněny , a budou vypracovávat zprávy . Komise bude na tento systém dohlížet .

R: um sicherzustellen , dass die Voraussetzungen für die Ermächtigung auf Dauer erfuellt werden , schaffen die betreffenden Mitgliedstaaten ein ständiges System regelmäßiger amtlicher Kontrollen und entsprechender Kontrollberichte , das von der Kommission überwacht wird .

B: betroffenen Mitgliedstaates Ausschuß Ausschuß System Übertragung amtlichen Erhaltungszüchtungskontrollen bestimmen das , in zu Bedingung Ermächtigung die nachhaltig als , die Maßnahme Bericht der Kommission die , System Datum der

E: betroffenen Mitgliedstaaten Ausschuß Ausschuß System Übertragung amtlichen Erhaltungszüchtungskontrollen bestimmen , in das zu Bedingung Ermächtigung die nachhaltig als , die Maßnahme Bericht die Kommission , Informationssystem die Datum der

O: Rozhodnutí 93 / 231 / EHS se zrušuje .

R: die Entscheidung 93/231/EWG wird aufgehoben .

B: Änderung der Entscheidung wird aufgehoben .

E: der Entscheidung 93/197/EWG wird aufgehoben .

O: Odkazy na zrušené rozhodnutí se považují za odkazy na toto rozhodnutí v souladu se srovnávací tabulkou obsaženou v příloze III .

R: Bezugnahmen auf die aufgehobene Entscheidung gelten als Bezugnahmen auf die vorliegende Entscheidung und sind nach Maßgabe der Entsprechungstabelle in Anhang iii zu lesen .

B: Bezugnahmen auf die als Bezugnahmen auf die vorliegende Entscheidung und sind enthalten der Anhang iii

E: auf die aufgehobene dieser Entscheidung als Bezugnahmen auf die vorliegende Entscheidung und sind der iii

Příloha C

Návod k použití překladového systému

Na přiloženém CD je k dispozici experimentální překladový systém, který vznikl v rámci vypracování této diplomové práce. Tento systém postrádá přítulné uživatelské rozhraní, proto je vhodné před jeho spuštěním projít tento stručný návod k použití. Pro nastavení parametrů systému jsou třeba mírné úpravy zdrojového kódu systému.

Systém se spouští jedním z těchto příkazů (spuštěných v Unixovém prostředí v adresáři systému):

- `perl translate.pl` *id_věty*
- `./translate_file.sh` *playlist.txt*, kde *playlist.txt* je textový soubor obsahující seznam identifikátorů vět.

První z nich přeloží jednu větu určenou identifikátorem, druhý umožňuje překlad více vět určených identifikátory v textovém souboru. Spuštění druhého skriptu je jen jednoduché opakování prvního volání, proto v dalším budeme předpokládat, že systém byl spuštěn prvním uvedeným příkazem.

Úspěšný běh systému je podmíněn těmito body:

1. *V systému je nainstalován Perl a potřebné moduly.*
   Požadavky na verzi Perlu a na jeho moduly vychází z toho, že systém používá knihovnu FSLib[1], tudíž má stejné požadavky jako tato knihovna.
2. *Systém má dostupné datové zdroje*
   Datové zdroje potřebné pro běh systému jsou tyto:
   - samotná paralelní data (training)
   - dvojjazyčný slovník (extrahovaný postupně skripty `load_sentences.btred` a `extract_lemmas_dict.pl` z adresáře `tools/my-scripts`)

[1] `http://ufal.mff.cuni.cz/pdt2.0/doc/tools/tred/Fslib.html`

- indexy pro rychlé vyhledání lemmat ve větách zdrojového jazyka (získané skriptem `lemma_index.btred` spuštěným jednorázově na sadu trénovacích dat)

Cesty k těmto souborům musí být uvedeny v souboru `Data_access.pm`.

3. *Identifikátor věty je validní v sadě dat, kterou systém prohledává pro možnost překladu.*
   Prohledávaná sada dat je určena proměnnou `$data_set` v souboru `Data_access.pm`

Chování systému určují následující faktory:

1. *Varianta spuštěného systému (s lingvistikou nebo bez)*
   Na přiloženém CD jsou připravené čtyři varianty systému:
   - výchozí systém pro směr čeština-angličtina (adresář `baseline`)
   - systém obohacený lingvistikou pro směr čeština-angličtina (adresář `enhanced`)
   - výchozí systém pro směr čeština-němčina (adresář `baseline-de`)
   - systém obohacený lingvistikou pro směr čeština-němčina (adresář `enhanced-de`)
2. *Hodnota parametru, který určuje mezní ohodnocení alignmentu*
   Tuto hodnotu určuje proměnná `$treshold` v souboru `translate.pl`.
3. *Hodnota koeficientu, kterým jsou zvýhodněny souvislé úseky*
   Tuto hodnotu určuje proměnná `$continuous_advantage`

Příloha D

Adresářová struktura přiloženého CD

Přiložené CD obsahuje tyto adresáře:

- `baseline` – výchozí systém pro překlad z češtiny do angličtiny
- `baseline-de` – výchozí systém pro překlad z češtiny do němčiny
- `enhanced` – lingvistikou obohacený systém pro překlad z češtiny do angličtiny
- `enhanced-de` – lingvistikou obohacený systém pro překlad z češtiny do němčiny
- `outputs` – anglické výstupy a jejich hodnocení
- `outputs-de` – německé výstupy a jejich hodnocení
- `data1`
  - `JRC-ACQUIS` – německá data v PML a jejich konverze do Perl Storable
  - `JRC-ACQUIS-extractions` – extrakce z německých dat, slovník a index lemmat, uloženo v datových strukturách Perl Storable. Čtení těchto struktur vyžaduje 64-bitovou architekturu, pro běh systému na 32-bitovém stroji je třeba extrakce znovu vygenerovat.
  - `PCEDT` – anglická data v PML (adresář chybí, protože PCEDT je chráněn autorským právem)
  - `PCEDT-extractions` – extrakce z anglických dat (taktéž chybí)
- `perl-things` – moduly perlu potřebné pro běh systému
- `tred-things` – části editoru tred pro načítání PML dat
- `tools`

  - `cdg-scripts` – skripty pro přípravu vstupu CDG parseru
  - `human-evaluation` – velmi jednoduchá PHP aplikace pro použité hodnocení, porovnávání systémů anotátory.
  - `my-scripts` – různé skripty pro konverzi dat z JRC-Acquis korpusu a nakonec nepoužité skripty pro konverzi dat z překladových pamětí a z korpusu OPUS. Navíc některé soubory odkazované z textu.

- `text` – zdrojové texty tohoto dokumentu a použité obrázky

V kořenovém adresáři je navíc elektronická podoba tohoto dokumentu ve formátu PDF.

Literatura

[Brown99] Ralf D. Brown. Adding Linguistic Knowledge to a Lexical Example-Based Translation System.
Proceedings of the TMI-99, Chester, England. 1999.

[Collins99] Michael Collins, Jan Hajič, Lance Ramshaw, and Christoph Tillmann. A statistical parser for czech.
Proceedings of the 37th Annual Meeting of the ACL. College Park, Maryland. 1999.

[Čmejrek03] M. Čmejrek, J. Cuřín, J. Havelka. Czech-English Dependency-based Machine Translation.
Proceedings of the 10th Conference of The European Chapter of the ACL. Budapest, Hungary. 2003.

[Čmejrek04] M. Čmejrek, J. Cuřín, J. Havelka, J. Hajič, V. Kuboň. Prague Czech-English Dependecy Treebank: Syntactically Annotated Resources for Machine Translation.
4th International Conference on Language Resources and Evaluation. Lisbon, Portugal. 2004.

[Foth2000] Kilian Foth, Wolfgang Menzel, and Ingo Schrder. A transformation-based parsing technique with anytime property.
Proceedings of the International Workshop on Parsing Technologies (IWPT-2000). Trento, Italy. 2000.

[Hajič98] J. Hajič, B. Hladká. Tagging Inflective Languages: Prediction of Morphological Categories for a Rich, Structured Tagset.
Proceedings of COLING-ACL Conference. Montreal, Canada. 1998.

[Hutchins05] W. J. Hutchins. Towards a definition of example-based machine translation.
`http://ourworld.compuserve.com/homepages/WJHutchins/`.
2005.

[Hutchins06] W. J. Hutchins. Machine translation: a concise history.
`http://ourworld.compuserve.com/homepages/WJHutchins/`.
2006.

[Lavie05] S. Banerjee, A. Lavie. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. *Proceedings of Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization at the 43rd Annual Meeting of the ACL-2005*. Ann Arbor, Michigan. 2005.

[Nagao84] M. Nagao. A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. A. Elithorn, R. Banerji. *Artifitial and Human Intelligence*, 173–180. 1984.

[Papineni02] K. Papineni, S. Roukos, T. Ward, W. J. and Zhu. BLEU: a method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the ACL*. 2002.

[Sato90] S. Sato, M. Nagao. Toward memory-based translation. *Proceedings of COLING-90*. Helsinki, Finland. 1990.

[Somers03] H. Somers. An overview of EBMT. M. Carl, A. Way *Recent advances in Example-Based Machine Translation*, 3–57, Dordrecht, 2003.

[Turcato03] D. Turcato, F. Popowich. What is example-based machine translation? M. Carl, A. Way *Recent advances in Example-Based Machine Translation*, 59–80. Dordrecht. 2003.

[Vilar07] D. Vilar, G. Leusch, H. Ney, R. E. Banchs. Human Evaluation of Machine Translation Through Binary System Comparisons. *Proceedings of the Second Workshop on Statistical Machine Translation*. Association for Computational Linguistics. Prague, Czech Republic. 2007.

[Žabokrtský02] Petr Sgall, Zdeněk Žabokrtský, Sašo Džeroski. A Machine Learning Approach to Automatic Functor Assignment in the Prague Dependency Treebank. *Proceedings of LREC 2002*. ELRA. 2002.