Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE

Bc. Josef Čech

Spojování segmentů v českých souvětích

ÚFAL - Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: doc. RNDr. Vladislav Kuboň Ph.D.
Studijní program: Matematická lingvistika
Studijní obor: Matematická lingvistika

Praha 2014

Na tomto místě bych rád poděkoval vedoucímu práce doc. RNDr. Vladislavu Kuboňovi Ph.D. za ochotu, trpělivost a cenné rady při řešení problémů, které se během psaní práce objevily. Zároveň bych chtěl poděkovat svým blízkým za podporu, zejména své manželce.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 11. 4. 2014 Josef Čech

Název diplomové práce: Spojování segmentů v českých souvětích

Autor: Bc. Josef Čech

Katedra (ústav): ÚFAL – Ústav formální a aplikované lingvistiky

Vedoucí diplomové práce: doc. RNDr. Vladislav Kuboň Ph.D.

e-mail: vk@ufal.mff.cuni.cz

Abstrakt: Práce se zabývá studiem lingvisticky motivovaných částí vět – segmenty – a jejich vzájemnými vztahy. Vztahy mezi segmenty popisují jejich slučování do větších větných celků – klauzí. Detekovat segmenty je možné pomocí definice lingvistických oddělovačů založené na pravidlovém přístupu. Tento přístup se osvědčil i při určování vztahů mezi sousedními segmenty. Práce má za úkol zjistit, zda je možné pomocí pravidlového přístupu získat ze segmentů i klauze. Zároveň při práci vznikl návrh pozičního tagování segmentů. Tag segmentu popisuje jednotlivé vlastnosti segmentu jako celek. V metodách analýzy vztahů segmentů je plně zaměnitelný za celý segment.

Klíčová slova: segment, klauze, tag, spojování segmentů, syntaktická analýza

Title: Joining segments in Czech sentences

Author: Bc. Josef Čech

Department: Institute of Formal and Applied Linguistics

Supervisor: doc. RNDr. Vladislav Kuboň Ph.D.

e-mail: vk@ufal.mff.cuni.cz

Abstract: This thesis follows up segmentation of complex sentences to linguistic motivated objects – segments – and their mutual relations. These relations can be used for next work with segments. Main purpose for mapping relations is their joining into next level unit – clause. Theoretically should be possible to analyse each clause of complex sentence separately. Analysis of set of clauses should be quicker than of analysis whole complex sentence. Segments should be found thanks to linguistic separators and rule approach. Rule approach proves in problem relations between neighbouring segments. This thesis should attest that rule approach is best solution for joining segments into clauses. Position tag of segment was part of this thesis. This tag should be used in methods dealing with segments instead of custom segment.

Keyword: segment, clause, tag, joining segments, syntactic analysis

Obsah

1 Úvod

Syntaktická analýza je jedna ze základních a také nejnáročnějších procedur při automatickém zpracování textu. Na jejích kvalitních výsledcích stojí mnoho aplikovaných úkolů z počítačové lingvistiky. Zároveň však patří mezi ty nesložitější úkoly. Díky práci na parserech (syntaktických analyzátorech) je dnes možné analyzovat krátké věty s poměrně velkou spolehlivostí. Ovšem u všech algoritmů nastává problém s mírou spolehlivosti při analýze rozvitějších vět (více v článku [8]).

Algoritmus syntaktické analýzy musí zpracovat velké množství variant, kterými lze větu analyzovat. V takovém případě se zvyšuje nejen pravděpodobnost chyby, ale zároveň čas, který je potřeba analýze věty věnovat. Jednou z možností, jak by bylo možné zvýšit úspěšnost, je předzpracovat složitá souvětí do malých celků, které parser dokáže zpracovat rychleji a s vyšší úspěšností.

Teorie segmentů a klauzí umožňuje rozložení vět do menších celků. Tyto celky mají lingvistickou motivaci. Zjednodušeně lze klauze přirovnat k větám v souvětí a segmenty k části klauze, která je rozdělená. Klauze by pak mohly tvořit nový předzpracovaný vstup pro syntaktickou analýzu, který by mohl výsledky zlepšit.

Motivací pro tento způsob předzpracování jsou výsledky procedury, která byla aplikována na syntakticky analyzovaných větách v PDT (Pražském závislostním korpusu). Úspěšnost této procedury a zároveň rozložení výsledků ukazuje, že pokud by se podařilo analyzovat pouze jednotlivé klauze, které by bylo možné sloučit do celkového výsledku, mohlo by to přinést zlepšení.

Otázkou v takovém případě je, zda je možné postupy otočit, tedy nejdříve zpracovat věty z pohledu segmentů a klauzí – sloučit segmenty do klauzí bez znalostí analytických funkcí a vztahů mezi slovy. Segmenty jsou podle článků [1] i [2] dobře definovatelné a jejich nalezení není příliš složité. Klauze jsou však dobře definované pouze pro syntakticky analyzované věty (viz článek [7]) a v povrchové struktuře musíme zatím pracovat pouze s její intuitivní představou (viz kap. 4).

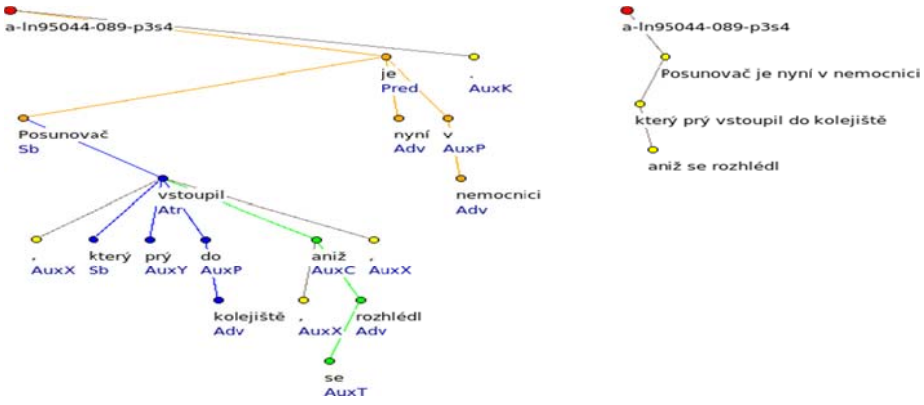
V době, kdy jsem začal pracovat na tématu segmentů a klauzí, existovala již sada pravidel pro segmentaci věty na povrchové struktuře, několik kvantitativních analýz i malý korpus několika tisíc vět, které byly anotovány právě pro práci se segmenty.

Ve své práci bych chtěl ukázat, že i přes různé problémy s detekcí klauzí na povrchové rovině a přes neznalost syntaktických a sémantických vlastností, lze s určitou mírou úspěšnosti nalézt klauze ve větě a jejich vztahy. Vytvořením takovéto struktury by byl připraven vstup pro syntaktickou analýzu. Tento předzpracovaný vstup by měl být zpracován rychleji a snad i úspěšněji. Obsahem této práce by však mělo být pouhé zpracování textu z pohledu segmentů a klauzí. Tedy rozložení vět do klauzí a segmentů a určení jejich vztahů bez využití analytických informací v textu. Základními vstupními daty budou texty, které jsou rozdělené na věty a zároveň již prošly morfologickou analýzou.

1.1 Rozložení věty do klauzí a PDT

Segmenty jsou přesně definované části vět, které se nachází mezi dvěma oddělovači segmentů (viz kap. 4). Klauze jsou celky složené ze segmentů podle různých pravidel (viz kap. 7.1). Podle [10] je možné v PDT na analytické rovině určit klauze a segmenty velmi jednoduše díky syntaktické analýze. Procedura, která umožňuje na tomto pracovat, je velmi přímočará a dosahuje úspěšnosti 97.51 %.

Procedura, která určuje klauze na analytické rovině, má tak vysokou úspěšnost, protože klauze si lze pro jednoduchost představit v terminologii PDT jako jednotlivé úrovně mezi slovesy, jak zobrazuje následující schéma.



Obr. 1: Grafické zobrazení klauzí

2 Teoretický úvod

Jak bylo řečeno výše, syntaktická analýza je sama o sobě velmi náročný proces, a zejména pro jazyky s poměrně volným slovosledem, jakým je například čeština. Při automatické syntaktické analýze se pro každou větu generuje několik možností analýzy. Na základě naučených znalostí je poté vybrána jediná možnost. Čím složitější věta je analyzována, tím je počet variant větší. Proto se složitostí vět klesá spolehlivost jejich automatické analýzy.

Kroky, které by mohly zlepšit spolehlivost i rychlost, můžeme rozřadit do několika skupin. V prvním případě lze zvyšovat počet a kvalitu vstupních dat, ale historie ukázala, že spolehlivost zlepšovaná na základě dat stoupá pouze logaritmicky a pro znatelné zlepšení je nutno více a více dat. Navíc příprava kvalitních dat je časově i finančně náročný způsob zvyšování spolehlivosti. Dalšími způsoby jsou změna algoritmu učení, uchování nových informací nebo zlepšení pomocí heuristik. Tyto přístupy jsou jistě relevantními technikami, ale jejich hlavní problém spočívá v neustálé kontrole výsledků při každé změně. Zároveň bychom se měli vyhnout přizpůsobení algoritmu na testovací množinu dat. Spojení nutnosti neustále testovat a mít dostatečně rozmanitá data vede k nutnosti mít poměrně rozsáhlou množinu dat, na které je možné algoritmy testovat. Další prostor pro zlepšení se skrývá v možnosti předzpracování vstupních a zpracování výstupních dat. Tento přístup je možné v širším kontextu chápat jako určitý typ heuristiky.

Tato práce se zabývá právě předzpracováním vstupních dat způsobem, který by mohl pomoci zrychlit celý proces a snad i zlepšit výsledky. Vstupními daty pro tuto práci jsou věty opatřené pouze morfologickými informacemi. Výsledkem předzpracování je rozdělení složitého souvětí na menší celky, které lze samostatně analyzovat. Tyto celky byly nazvány klauze. Klauze by v ideálním případě měly reprezentovat jednu větu v souvětí. Čeština je jazyk velmi tvárný a mezi klauzemi byly při analýzách často nalezeny i takové, které nereprezentovaly přímo větu, ale pouze sousloví, které je do věty vloženo jako doplňující informace. Zároveň klauze vkládané do jiných klauzí zapříčiní roztříštění těchto klauzí v povrchové struktuře. Rozdělení klauze znamená oddělení segmentů klauze, segment jako jednotka věty je nerozdělitelná. Na těchto popsaných případech je ukázána hlavní překážka, která

brání v přímém vytvoření klauze ze segmentů. Problémy skrývají hned dvě poměrně časté situace:

1. Existence klauze, která nereprezentuje celou větu (vsuvka, výpustka).

2. Existence segmentů, které patří do jedné klauze, ale mezi nimi se nachází další segmenty v rámci povrchové struktury.

Segmenty, které podle prvního bodu tvoří vlastní klauzi, je těžké rozlišit od segmentů, které jsou pouze v členské koordinaci s jinými segmenty předchozí klauze. Tento problém je náročný vzhledem k nedostatečnému porozumění textu. Pro řešení druhého bodu lze použít intuitivní představu o klauzi. Podle této představy by se měly všechny její segmenty nacházet na jedné úrovni zanoření segmentů. Toto je jeden z důvodů, proč je v rámci této práce potřeba provést celý proces segmentace a určení správného indexu segmentu.

Získání klauzí přímo z povrchové struktury by znamenalo identifikaci oddělovačů pro klauze a zároveň identifikaci jednotek, které umožňují sloučit některé části klauze do sebe (těch, které byly odděleny závislou klauzí). Tento úkol není reálný v případě, že nemáme žádnou další informaci, ať syntaktickou, nebo sémantickou. Předzpracování věty na segmenty odfiltruje méně důležité informace a zdůrazní ty, které by mohly být důležité. Při tvorbě klauzí je vybíráno právě z těchto vybraných informací.

Na základě teoretických předpokladů by mělo dojít ke zlepšení spolehlivosti a k zrychlení běhu analýzy na složitějších větách. Ale rychlejší syntaktická analýza není jediná aplikace, pro kterou lze segmenty, popř. klauze využít. Klauze a segmenty by mohly být použity také při automatickém překladu. I v této oblasti se ukazuje, že většina algoritmů pracuje lépe s krátkými a jednoduchými větami, a proto by zjednodušení vstupů mohlo pomoci jak v rychlosti, tak v kvalitě překladu.

Jedním z důvodů proč se předpokládá, že segmentace a následně vytvořené klauze mohou významně pomoci při syntaktické analýze, je fakt, že při tvorbě klauzí omezíme části věty pro syntaktickou analýzu. V terminologii analytické roviny v PDT označíme izolací klauze v souvětí podstrom, který je možné analyzovat.

3 Data

Původním záměrem bylo využití již anotovaného zdroje PDT, ze kterého by se data získala transformací pomocí několika pravidel. Analytická rovina uchovává informace především o vztazích mezi slovy, neobsahuje informace o vztazích mezi skupinami slov. Snaha získat z analytické roviny PDT správně indexované segmenty nebyla příliš úspěšná (viz článek [2]).

Pro další práci byla připravena sada zlatých dat z PDT. Jedná se o sadu 2500 vět, které se nachází v PDT, k nimž byla přidána informace o počtu segmentů a jejich indexu. Zároveň byla přidána informace o vztazích mezi sousedícími segmenty. Anotace uchovává informaci o tom, které ze sousedících segmentů jsou natolik provázané, že pokud bychom změnili index úrovně jednoho z nich, museli bychom změnit index celé skupině segmentů.

Tato zlatá data jsou velmi pečlivě připravena, ale nezachycují informaci o vztazích segmentů, které tvoří dohromady klauzi, i když jsou od sebe vzdálené přes několik dalších segmentů.

Podle článku [7] lze celkem přesně izolovat klauze na základě informací z analytické roviny, a proto je možné rekonstruovat klauze věty. „Zlatá data“ jsou vybranou kolekcí z PDT, a je tedy možné izolovat klauze z anotovaných vět. Na výše zmíněném základě byly izolovány klauze v celém PDT s F-score více jak 97 %.

3.1 Tvorba „zlatých dat“

Data, která jsou k dispozici: ručně označkovaná kolekce segmentů (celkem 2500 vět) a PDT 2.5 (morfologické informace a číslo klauze, které se nachází na analytické rovině). Prvním krokem bylo načtení vět z morfologické roviny a jejich rozdělení na segmenty. Naštěstí se nepotvrdily mé obavy – počty mnou nalezených segmentů plně odpovídaly počtům segmentů v anotovaných větách. Dalším krokem bylo načíst věty z analytické roviny a každému slovu přiřadit číslo klauze. Tato úprava zajistí snadnější a přímější porovnání výsledků mých algoritmů a ruční anotace.

Sada 2500 vět anotovaná z pohledu segmentů obsahuje věty náhodně vybrané

přes celý soubor dat v PDT. Kvůli tomuto způsobu volby vět bylo nutné při úpravě „zlatých dat“ projít celou kolekci PDT. Naopak díky proceduře v PDT, která je schopná na základě analytických stromů s 98% schopností určit klauzi, jsem byl schopen připravit i data, která umožní otestování úspěšnosti tvorby klauzí přes celé PDT, ovšem bez znalosti segmentů a jejich zanoření.

Výše zmíněná úprava „zlatých dat“ byla poměrně přímočarým krokem, během kterého nedošlo k významnějším problémům. Bylo pouze nutné upravit některé drobnosti v závislosti na zpracování segmentů a klauzí v PDT (viz kap. 4.7). Příkladem takového drobného rozporu v PDT může být oddělení hranic klauzí (např. interpunkčních znamének) do samostatné klauze s číslem 0, ale v původní teorii jsou hranice duplikovány a část patří do předchozí klauze a část do následující klauze, pokud takové klauze existují.

4 Slovníček pojmů

V práci je použita především terminologie zavedená v článcích o segmentaci a klauzích. Články používají většinou stejné termíny, ale v jejich používání nejsou příliš jednotné. Pro potřeby této práce bylo nutné některé z termínů rozšířit nebo upravit jejich definici.

a) Segment

Segment je lingvisticky motivovaná jednotka. Intuitivně může být segmentem nazývána část věty, která obsahuje pouze autosémantická slova a která je ve větě oddělena jinými jednotkami. Předpokladem pro segmentaci vět jsou věty obohacené o morfologické informace. Segmenty, jak jsou definovány v článcích, jsou definovány v povrchové struktuře jako větné jednotky obsahující maximální neprázdnou množinu tokenů mezi dvěma hranicemi (viz níže). Tato maximální množina neobsahuje jinou hranici.

b) Hranice (oddělovač, separátor)

Hranice je jednotka věty, která odděluje dva segmenty. Realizace konkrétních slov a znaků reprezentující hranice je jazykově závislá. Součástí teorie segmentů je definice primitivních hranic. Primitivní hranice se skládají z jednoho tokenu. Protože vstupem pro segmentaci jsou data obohacená morfologickou analýzou, jsou primitivní hranice definovány pomocí morfologických značek i pomocí výčtu jednotlivých konkrétních tokenů. Tento token může být spojka, interpunkce ve větě, závorky, pomlčky a uvozovky. Hranicí definovanou tagem jsou spojky podřadicí („J,“). Výčtem definované hranice tvoří většinou nealfanumerické znaky, které se mohou vyskytnout ve větách. Jsou to dvojtečka, čárka, vykřičník, otazník, středník, závorky, pomlčky, uvozovky označující přímou řeč a dvě speciální hranice - začátek a konec věty.

V případě výskytu dvou a více oddělovačů jsou tyto oddělovače sloučené v jednu hranici, se kterou dále pracujeme jako s jediným objektem. Tento typ hranice je nazýván složenou (komplexní) hranicí.

V článcích se pro hranici volí různá označení např. oddělovač, separátor, hranice segmentu. Definice termínu hranice sémanticky označuje právě hranici mezi segmenty. Pro další práci je však nutné rozlišit i termín pro hranici mezi klauzemi.

Hranice je objekt, který nese především informaci o hloubce zanoření, a každá jeho realizace může napovědět, jakým způsobem ovlivňuje okolní segmenty. Hranice jsou objekty, jejichž vlastnosti splývají s vlastnostmi segmentů, proto se v této práci s oběma objekty zachází velmi podobně. Hlavním rozdílem jsou tokeny, ze kterých se segmenty a hranice skládají.

c) Hranice klauze

Hranice segmentů rozdělují větu na segmenty. Hranici segmentu, která zároveň uvozuje novou klauzi nebo ukončuje předchozí, nazýváme hranicí klauze. Realizace hranice klauze je shodná s realizací hranice segmentů. Rozdíl se skrývá v chápání objektu hranice. Pouze o dvou hranicích segmentů je možné prohlásit, že jsou vždy zároveň hranicemi pro klauze. Jsou to speciální hranice – začátek a konec věty.

d) Příznak (flag)

Segmenty mohou obsahovat speciální konstrukce nebo slova, která mohou napovědět, jak s nimi v dalších analýzách pokračovat. V současné době se používá pouze příznak podřízenosti (subflag), který vyjadřuje závislost jednoho segmentu na druhém. Očekává se, že s dalším studiem jich bude přibývat.

e) Index segmentu

Jde o přirozené číslo, které je přiřazeno každému segmentu. Číslo vyjadřuje hloubku zanoření. Nejvyšší úroveň má index 0 a hloubka je teoreticky neomezená. Omezení je ovšem dáno myšlenkovými schopnostmi mluvčího. Maximální hloubka zanoření je rovna hloubce, do které je mluvčí (uživatel jazyka) schopen udržovat v paměti kontext, aby mohl celé větě bez problémů porozumět.

f) Úroveň segmentů

Je to imaginární rozdělení segmentů podle jejich indexu.

g) Klauze

Klauze je neprázdná množina segmentů na jedné úrovni zanoření mezi dvěma hranicemi klauzí. Hranice mezi segmenty jsou součástí klauze. Pokud jako příklad použijeme Obr. 2, poté segmenty {1, 5} mohou tvořit jednu klauzi a segmenty {2, 4} mohou tvořit druhou klauzi, která je závislá na první klauzi. Předchozí definice předpokládá výsledek segmentace, tedy rozdělení věty na segmenty a hranice, a zároveň přiřazení indexu zanoření každému z objektů.

Přesný způsob, jakým lze definovat klauzi v povrchové struktuře věty, neexistuje. Existuje však intuitivní popis, jakým by klauze měly vznikat. Klauze je možné přirovnat k jedné samostatné větě uvnitř souvětí. Některé části souvětí mohou obsahovat výpověď, ale nemusí být určené jako plnohodnotné věty v rámci souvětí (např. vsuvky, apozice, výpustky).

V článku [7] definují autoři klauzi na základě analytické roviny. Základní pravidlo je myšlenkově nenáročné, ale tato definice nevychází z žádné lingvistické intuice. Definice z článku [7] říká, že klauze je podstrom predikátu, včetně predikátu samotného. Do tohoto pravidla musíme zahrnout dvě výjimky:

1. Pokud predikátu, který uvozuje klauzi, předchází podřadicí spojka, je tato spojka součástí uvozované klauze.

2. Pokud je predikát, který uvozuje klauzi, v podstromu jiné klauze, není klauze uvozená tímto predikátem součástí nadřazené klauze.

Snadná identifikace predikátu na analytické rovině PDT se provádí na základě analytických funkcí, morfologických značek či pozice ve stromu (uzel, který je koordinován s jiným predikátem).

h) Segmentační schéma (segmentation chart)

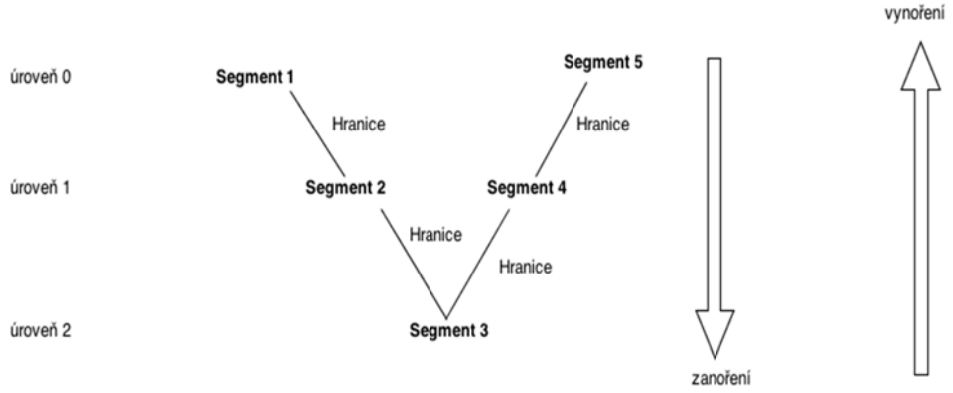
Jde o množinu uspořádaných dvojic čísel (*i, j*), kde *i* je index segmentu a *j* je pozice v povrchové struktuře. Tato množina nám dává představu o způsobu, jakým je věta segmentována. Pro představu vztahů mezi segmenty, které segmentační schéma představuje, lze segmentační schéma použít jako množinu vrcholů *V* do grafu *G* = (*V, E*), kde *E* je množina hran mezi vrcholy, pro které platí $(e_1, e_2) = ((i_1, j_1),(i_2, j_1+1))$. Pro tento termín se vyskytují ve článcích různé definice (viz níže), ale tyto definice příliš nevyhovují. Přesto jsem popsal alespoň jejich základní myšlenky, abych ukázal, že předchozí definice termínu neztrácí žádnou informaci.

V článku [2] je segmentační schéma definováno jako řetězec indexů segmentu. Pozice indexu v řetězci odpovídá pozici segmentu v povrchové struktuře. Tato definice podle mne nebere v úvahu segmenty s indexem, který má vyšší hodnotu než 9. (010) je příkladem segmentačního schématu pro větu, která obsahuje tři segmenty. – první a třetí segment s indexem 0 a druhý segment s indexem 1. Není to schéma věty, která má dva segmenty – první s indexem 0 a druhý s indexem 10.

V článku [5] je za segmentační schéma považován graf, kde jsou hranice zaneseny jako dvojice uzlů, které jsou propojeny dvěma typy hran. Horizontální hrany mezi dvěma hranicemi reprezentují segment a druhý typ hran mezi kopiemi jedné hranice reprezentuje zanoření o úroveň níže nebo návrat na vyšší úrovně.

4.1 Terminologie

Vzhledem ke snaze popisovat vztahy a průběh segmentace tak, aby se shodovaly s grafickým znázorněním (např. Obr. 2), může dojít k mírnému zmatení. Segment, který je považován za hlavní a je na nejvyšší úrovni zanoření, má hodnotu indexu 0. Čím hlouběji je segment zanořen (segment tedy závisí na předchozím segmentu), tím vyšší je i hodnota indexu segmentu. Při zpětném procesu dochází k vynořování, tedy ke snižování indexu zpět k 0. Situaci lze přirovnat k udávání hloubky ponoru.



Obr. 2: Grafické znázornění versus terminologie

5 Segmentace

5.1 Motivace

Čeština je jazyk s volným slovosledem. Tato vlastnost umožňuje uživatelům jazyka skládat jednotlivá slova téměř libovolným způsobem. Volný slovosled je jedním z důvodů, proč je syntaktická analýza češtiny velmi složitá. Závislostní syntaktická analýza může vygenerovat mnoho variant, odstranit špatné typy závislostních stromů a na konci analýzy vybrat nejpravděpodobnější variantu. Redukční fáze by měla odstranit všechny špatné analýzy, v ideálním případě by měla zbýt pouze jedna. Český jazyk má poměrně volný slovosled, proto pravidla pro oddělování jednotlivých částí musí být pevná. Těchto pravidel využívá segmentace k rozdělení věty na menší celky.

Návrh segmentace původně sloužil hlavně k ukázce slabin syntaktických analyzátorů pro češtinu. S dalším studiem vznikaly návrhy na využití této metody i pro předzpracování dat ke zjednodušení a zlepšení analýz.

Většina prací dělí text na věty a věty na slova, která bere jako základní jednotku členění pro všechny další analýzy. V některých případech se ukazuje, že takové členění je nedostatečné a že mohou existovat skupiny slov, které se chovají jako samostatné jednotky (např. přísudky jmenné se sponou). Segmentace se snaží rozdělit větu na větší, lingvisticky motivované části, se kterými by bylo možné dále pracovat nebo je analyzovat jako samostatné jednotky.

5.2 Seznam oddělovačů

Základem seznamu se staly přirozené oddělovače textu, jako jsou interpunkce, spojky, závorky (všech typů), pomlčky a uvozovky označující přímou řeč. Dalšími dvěma speciálními oddělovači jsou začátek a konec věty. Oddělovač pro označení začátku věty je imaginární oddělovač. Oddělovač pro konec věty je reprezentován ukončovacími interpunkčními znaménky, jako jsou tečka, vykřičník, otazník atd. Při analýze se vychází z předpokladu, že jsou k dispozici morfologické informace a jednoznačný konec věty. Proto tečka za řadovou číslovkou nebo tečka pro označení

zkratky není při tvorbě segmentů považována za oddělovač. Další výjimku tvoří uvozovky, které dodávají slovu nebo sousloví ironický význam. V takovém případě bychom uvozovky neměli počítat do seznamu oddělovačů, ale bohužel tento jev nelze poznat bez znalosti vyšších rovin jazyka. Některé nealfanumerické separátory lze označit za párové. Jedná se především o závorky, uvozovky označující přímou řeč a pomlčky, které oddělují vsuvky. Části mezi těmito párovými oddělovači mají k větě velmi vzdálený vztah. Analýzu částí uvnitř těchto párových oddělovačů je možné oddělit od zbytku věty a provést ji zcela samostatně. Části mezi párovými oddělovači mohou sloužit jako nový vstup pro samostatný podproces segmentace. Po určení segmentů a jejich indexů lze tuto část zapojit pod předchozí segment. Indexy připojovaných segmentů se pak zvýší o i+1, kde i je hloubka segmentu, ke kterému je skupina připojena.

Dalším přirozeným typem oddělovačů jsou souřadicí spojky. V PDT je souřadicí spojka anotována tagem, který začíná *J^*. Za základní je možné považovat jednoslovné spojky, kombinací těchto jednoslovných spojek mohou vzniknout spojky víceslovné. Víceslovné spojky mohou být kombinací i jednoslovných spojek a jiného slovního druhu nebo násobným použitím jiného slovního druhu.

V českém jazyce existují i spojky podřadicí. Tyto spojky fungují také jako přirozené oddělovače, ale pro teorii segmentů jim byla přiřazena jiná role. Každé podřadicí spojce předchází čárka (s výjimkou skupin podřadicích spojek nebo kombinace podřadicí spojky a jiného tokenu, který může fungovat jako podřadicí spojka), a proto je její funkce co by hranice redundantní, ale z těchto spojek lze získat jinou důležitou informaci pro segmentaci. Této vlastnosti se věnuji v kapitole níže.

5.3 Příznaky

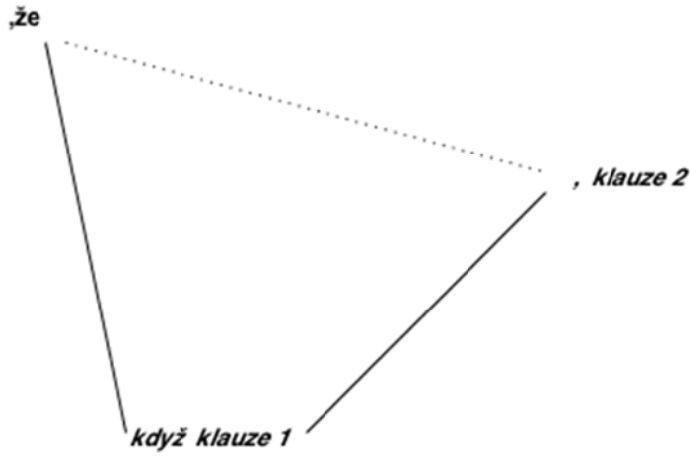
Dalším prvkem v teorii segmentů jsou tzv. příznaky. Příznak má popisovat elementární vlastnost, která je lehce automaticky rozpoznatelná. V současné chvíli teorie pracuje pouze s příznakem podřízenosti, který indikuje závislost segmentu, ve kterém je obsažen, na předchozím segmentu. Předpokládá se, že dalším studiem této teorie se množina příznaků rozroste o další typy.

Podobně jako oddělovače i seznam příznaků podřízenosti vychází z lingvistických vlastností. Přirozenými jednotkami, které zajišťují závislost jedné části věty na druhé, jsou podřadicí spojky. Dalšími kvantitativními analýzami byl tento seznam rozšířen o vztažná a tázací zájmena, zájmenná příslovce a speciální tvary číslovek. Většinu tohoto seznamu je možné jednoduše identifikovat na základě morfologických značek. Pouze pro zájmenná příslovce neexistuje konkrétní tag a jejich identifikace musí probíhat pouze na základě přidruženého seznamu (viz Tab. 1).

Tag(y)	Slovní popis	Výčet možných položek
J,	podřadicí spojka	že, aby, než, až…
P4, PE, PJ, PK, PQ, PY	vztažná a tázací zájmena	což, již, kdo, co, oč…
C?, Cu, Cz	číslovky	kolik, kolikrát, kolikátý
D	zájmenná příslovce	jak, kam, kde, kdy, proč

Tab. 1: Definice příznaků podřízenosti

V češtině je také možné spojení více podřadicích spojek za sebou. Tyto konstrukce mohou podobně jako v případě spojek souřadicích splynout v jeden příznak pro jeden segment. Další možností je, že podřízenou větu k hlavní klauzi předchází její podřízená věta (viz Obr. 3).



Obr. 3: Schéma výskytu dvou subflagů v souvětí

Podle článku [3] se při výskytu dvou podřadicích spojek z různých vět čárka píše pouze před první z podřadicích spojek. Avšak z krátkého nahlédnutí do ČNK (Českého národního korpusu) zjistíme, že pokud se setkají dva podřadicí výrazy, jsou možné obě varianty (viz Tab. 2). Bohužel ČNK obsahuje nejen formálně správné věty, proto poměr výskytů mezi prvním a druhým typem výskytů skupin podřadicích spojek zadává příčinu k úvaze, zda první typ netvoří pravopisně chybné věty.

Sousloví / prefix sousloví	čárka	jiné slovo
že, když	8	2
že když	5655	250
kter[.*] když	93	4
kter[.*], když	115	19

Tab. 2: Aplikace pravidla pro výskyt dvou podřadicích spojek

První sloupec ukazuje počet výskytů daného sousloví s prefixem v podobě interpunkce (čárka). Druhý sloupec obsahuje počet výskytů sousloví, u kterého je prefixem jiné slovo než interpunkce. Z tabulky jasně vidíme, že v prvním případě lze variantu s čárkou mezi spojovacími výrazy považovat za chybu. V druhém případě je vidět, že používání obou variant je téměř srovnatelné.

5.4 Algoritmus segmentace

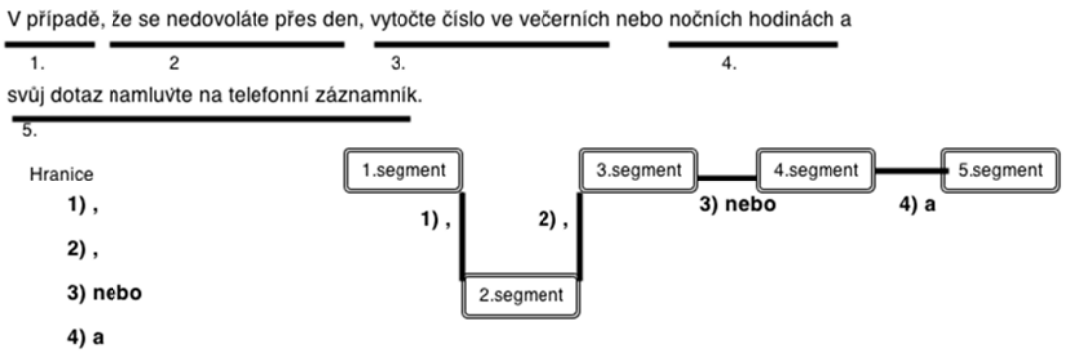
Minimálním vstupem pro aplikování algoritmu je jedna věta obohacená o morfologické informace. Jako výstup se očekává rozdělení na segmenty a přiřazení indexů jednotlivým segmentům. Z tohoto rozdělení indexů by mělo být rekonstruovatelné segmentační schéma tak, jak bylo popsáno na začátku této práce.

Algoritmus se skládá ze dvou obsáhlejších kroků:

1) rozdělení věty na segment,

2) označení segmentů indexem hloubky.

Následující schéma (Obr. 4) ukazuje, jak by měl algoritmus segmentace správně postupovat.



Obr. 4: Ukázka postupu algoritmu segmentace

První krok se zdá být velice přímý, přesto v sobě může skrývat jistý problém. Základní myšlenkou je rozdělení na menší jednotky podle jasně určených oddělovačů. Jak bylo výše ukázáno, v jazyce existují konstrukce, které plně neodpovídají pravidlům segmentace (viz Obr. 4).

V případě, kdy podobné konstrukce nebudou součástí analýzy, dojde ke dvěma typům chyb. První chybou je nesprávný počet segmentů a druhou chybou je chybné přiřazení indexu zanoření, který získá nižší hodnotu, než by odpovídalo skutečnému rozdělení. V takovém případě chyba ovlivní i rozložení klauzí.

Zpracování podobných konstrukcí je možné řešit dvěma způsoby. První způsob rozšíří pravidla pro přiřazení indexu segmentům o nový bod, který by zahrnoval tuto situaci a ovlivnil následující dva segmenty. U problému dojde k zapracování na úrovni příznaků, která vychází přímo z povrchové realizace věty, ale kromě přiřazení indexů by takové pravidlo muselo ovlivnit i počet segmentů. Druhý způsob znamená zapracovat hlubší analýzu a chápat podobnou konstrukci jako příznak virtuálního oddělovače. Dojde k obohacení seznamu hranic o novou virtuální hranici. V takovém případě již není třeba žádných dalších úprav a získáme správnou analýzu celé věty.

Po průběhu prvního kroku algoritmu je věta rozdělena na dvě množiny objektů, prvními jsou segmenty a druhými hranice. Oba typy objektů sdílejí stejný tag, a tak lze analyzovat data jen na základě těchto tagů segmentů. Krok, který vytváří tyto objekty (rozložení segmentů a hranic), je z větší části jasně definován. Není tedy problém automaticky získat z většiny vět správnou strukturu hranic a segmentů.

V druhém kroku se každému segmentu přiřadí přirozené číslo i od 0 do ∞. Pro

i z (1, ∞) platí, že segment s indexem *i* je závislý na prvním předchozím segmentu s indexem *i*-1. Zde dochází k prvním problémům automatického zpracování. Přiřazení indexu segmentu probíhá dle následujících pravidel, která vycházejí z [2]:

1) První segment věty má úroveň 0. Pokud však první segment obsahuje příznak podřízenosti, získá úroveň 1.

2) Pokud je hranicí segmentu čárka a následující segment neobsahuje příznak podřízenosti, dostává následující segment stejnou úroveň zanoření. V případě, že se jedná o již zanořený segment, může získat úroveň zanoření (0, *i*), kde *i* je úroveň předchozího segmentu.

3) Pokud je hranicí segmentu čárka a následující segment obsahuje příznak podřízenosti, získá následující segment index zanoření o 1 vyšší, než má předchozí segment.

4) Pokud je hranicí segmentu souřadně spojující výraz, získává následující segment stejnou úroveň, jakou má předchozí segment. Toto pravidlo platí i v případě, že oba segmenty (předchozí i následující) obsahují příznaky podřízenosti.

5) Pokud je hranicí tečka, otazník nebo vykřičník, jedná se o konec věty. Tato hranice je vždy poslední a zároveň získá index zanoření 0.

6) Pokud věta obsahuje přímou řeč (uvozovky nahoře jsou považovány za hranici, pokud začínají větu nebo následují po čárce, a uvozovky dole, pokud je věta uvozena uvozovkami přímé řeči). Segmenty ohraničené těmito uvozovkami mají index zanoření o 1 vyšší vzhledem k použití ostatních pravidel.

7) Pro segmenty, které se nacházejí mezi jinými párovými oddělovači, než jsou

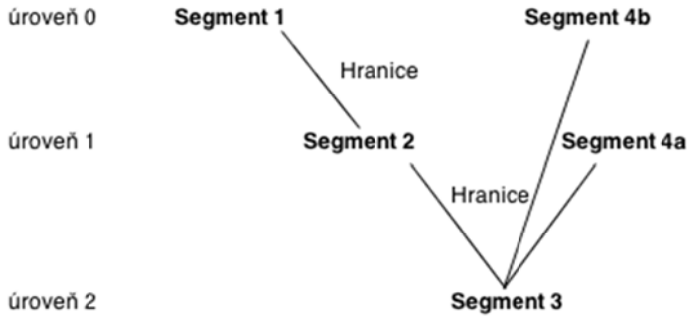
uvozovky pro přímou řeč, platí stejná pravidla jako v případě uvozovek pro přímou řeč.

Při analýze pro automatické zpracování jsou body 1, 3 a 5 snadno proveditelné. Detekce počátečního segmentu a závěrečné hranice je velmi triviální. Zároveň je možné omezit identifikaci segmentů s příznaky podřízenosti pouze na porovnání dvou množin objektů, jak bylo popsáno v předchozích kapitolách.

Body 6 a 7 pracují navíc s oddělovači, které se vyskytují v páru. Při identifikaci párových oddělovačů je třeba provést jedno prohledání omezené množiny párových oddělovačů. Tokeny mezi párovými oddělovači je možné zpracovat samostatně, mohou tak tvořit vstup pro novou instanci analýzy.

Bod 4 je konkretizací bodu 2. V tomto případě je obecná hranice nahrazena konkrétní souřadicí spojkou. Tyto podmínky umožňují přiřazení indexu následujícímu segmentu.

Na konci tohoto kroku by měl být každé podstruktuře věty přiřazen index zanoření. Většina těchto pravidel se zdá být velmi přímá a automaticky jednoduše proveditelná. Největší problém ale způsobuje bod 2, který dává velmi velkou volnost v přiřazení indexu následujícímu segmentu, pokud nejsou zapracovány další omezující podmínky pro přiřazování indexu, které vycházejí z hlubšího porozumění textu nebo z kvantitativních analýz některých jevů.



Obr. 5: Ukázka možnosti

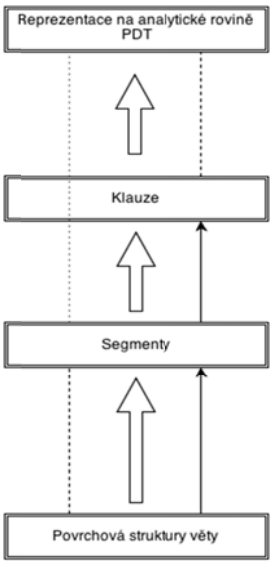
Za podmínek, které tento bod nastavuje, neexistuje možnost, jak správně rozhodnout, zda jde o koordinaci segmentů, která reprezentuje koordinaci větnou nebo členskou, nebo o segmenty patřící ke klauzím s indexem zanoření nižším než má předchozí segment. Segmenty, které vyhovují bodu 2, mohou získat celý rozptyl od indexu *i* předchozího segmentu až po hlavní (nultou) úroveň. Jistotu můžeme mít pouze ve speciálních případech. Příkladem je interpunkční znaménko na konci věty – koncová hranice získá vždy index zanoření 0. Pravděpodobnost, že bude v bodu 2 správně rozhodnuto, je tím vyšší, čím hlubší je naše porozumění samotnému textu. Tato práce však nepracuje s žádným hlubším porozuměním textu. Jediné informace, které v této práci máme k dispozici, jsou z morfologické analýzy. Pracujeme se syntaktickými vlastnostmi jednotlivých slov.

5.5 Vztah mezi segmenty, klauzemi a analytickou rovinou

V současné době se pro syntaktickou analýzu používá povrchová struktura obohacená o morfologickou analýzu. Využití klauzí pro syntaktickou analýzu je proces, který pro ověření použití potřebuje několik kroků. Některé z těchto kroků již byly provedeny.

Následující schéma (Obr. 6) přibližuje zařazení analýzy klauzí a segmentů z pohledu celkové analýzy jazyka. Bílé šipky znázorňují směr k syntaktické analýze věty. Tenké přerušované linie znázorňují vztahy, které již byly studovány. Přerušované čáry reprezentují směr k syntaktické analýze a tečkovaná čára směr od analytické roviny do nižších úrovní. Černé šipky znázorňují směr, kterým se zabývá tato práce.

Z Obr. 6 je patrné, že proces segmentace věty z holého textu již byl proveden v [9]. Proces segmentace se však skládá z vytvoření segmentů a přiřazení indexu zanoření. Bohužel pro získání segmentů a správného zanoření nebyly ve zmíněné práci použity pouze morfologické informace, ale také analytické funkce. Pokud mé výsledky mají být využity jako krok před syntaktickou analýzou, nesmí se odkazovat na data z hlubších rovin jazykové analýzy.



Obr. 6: Schéma vztahů mezi povrchovou strukturou věty, segmanety, klauzemi a analytickými stromy v PDT

5.6 Současný stav segmentace

Návrh segmentace se poprvé objevil v práci [5], kde byl použit pro znázornění složitosti analýzy českých vět. První navazující práce [4] se snažila připravit pravidla a ověřit je proti zdroji PDT. Ukázala, že navržená pravidla jasně identifikují segmenty. Problém však nastal při získání segmentů z analytické roviny PDT. Analytická rovina z pohledu segmentů obsahuje příliš detailní anotaci, která je zaměřena především na jednotlivá slova ve větě. Různé jevy ve větě byly navíc anotovány více způsoby. Bohatost anotace slov zapříčinila ztrátu informací o částech mezi lingvistickými oddělovači. Získat data z PDT, u kterých by byla 100% úspěšnost, a porovnávat je s výsledky na textem obohaceným o morfologickou analýzu, nebylo tak přímočaré, jak autoři zamýšleli. Proto bylo pro potřeby článku speciálně anotováno několik vět. Na těchto větách se mimo jiné ukázalo, že i když pravidla dovolovala poměrně velkou volnost a více způsobů, jakými lze přiřadit indexy segmentům, a proto i existenci více možných schémat, bylo většině vět přiřazeno pouze jedno schéma.

V případě nemožnosti využití analytické roviny z PDT bylo nutné vytvořit nový set testovacích dat. V PDT se anotují především vztahy mezi slovy a použití analytické roviny by vyžadovalo poměrně velké úsilí. Zároveň by takto získaná

anotovaná data nemohla být považována za dostatečně kvalitní vzorek. Opět dochází ke správnému rozdělení na segmenty, ale přiřazení indexů k segmentům není na analytické rovině implicitně anotováno. Jak ukázaly pokusy v článku [2], úspěšnost segmentace na analytických stromech PDT je nižší než 60 %. Proto byla vybrána množina vět, která byla pečlivě anotována pro potřeby pokusů se segmenty a klauzemi. Na základě těchto zlatých dat byla vytvořena kvantitativní analýza a připraveno několik návrhů pro další automatické zpracovávání segmentů a klauzí. Tyto návrhy budou blíže analyzovány v kapitole 7, příp. je možno detailnější informace získat z článku [1].

V článku [7] byly porovnávány výsledky, které byly získány z prostého textu a z navrženého způsobu, jak získat segmenty z PDT. Metoda získávání segmentů z PDT nebyla příliš úspěšná, i přes velmi obsáhlou a detailní anotaci. Výsledky jsou způsobeny především zaměřením anotace na slova a vztahy mezi nimi.

Dalším krokem byla definice klauze pomocí analytické roviny v rámci PDT. Tento krok se ukázal poněkud jednodušším než hledat jednotlivé segmenty. Tímto krokem bylo dokázáno, že i přes anotaci orientovanou na jednotlivá slova a vztahy mezi nimi, lze rekonstruovat část informace o skupinách slov, které nebyly přímo anotovány.

5.7 Problémy při tvorbě „zlatých dat“

V úvodu bylo naznačeno, jak vznikala data, která v další fázi práce slouží jako referenční vzorky. Jde o kombinaci tří (v případě započtení rozdělení vět na segmenty čtyř) zdrojů. Aplikovat tyto zdroje z oddělených dat přímo při analýze by jistě nebylo myšlenkově náročné, ale protože by byla tato data opakovaně využívána, značně by takový postup zkomplikoval a zpomalil samotnou analýzu. První část práce byla zaměřena především na sloučení důležitých informací ze všech zdrojů.

Zjednodušený postup slučování informací:

1. načtení morfologické věty ze souboru (soubory s příponou .m),
2. provedení segmentace na této větě,
3. načtení analytického souboru (soubory s příponou .a),

4. sloučení načtených informací,

5. nalezení věty v souboru ručně anotovaných segmentů,

6. přiřazení informace o úrovni zanoření k segmentům.

Předpokladem pro tento způsob sloučení bylo jasné identifikování hranic bez sebemenších odchylek, které se ukázalo jako bezproblémové. Při dokončení sloučení vět jsem hledal různé inkonzistence mezi úrovní zanoření a příslušnosti ke klauzi.

Při vytváření „zlatých dat“ jsem primárně prováděl pouze tři hlavní kontroly:

1. konzistence segmentů – zda počet nalezených odpovídá počtu segmentů v příslušném souboru obsahující anotaci segmentů,

2. konzistence klauzí a segmentů – zda segment neobsahuje slova z jiné klauze,

3. konzistence klauzí – zda všechny segmenty v jedné klauzi jsou na jedné úrovni.

První dvě kontroly proběhly v pořádku. Třetí typ kontroly již tak bezchybný není. Ukázalo se, že hlavním problémem je především chápání vsuvek, textu v závorkách nebo textu odděleného pomlčkami. Ve výše zmíněném popisu jsou tyto objekty samostatnými klauzemi. V PDT 2.5 jsou však v případě, kdy neobsahují sloveso, chápané jako součást klauze, která vsuvku obklopuje. V souboru „zlatých dat“ (2500 vět) bylo nalezeno celkem 306 vět (cca 12 %) porušujících třetí pravidlo kontroly. Přesto tuto chybovost nepovažuji za tak důležitou, protože jde o spojování segmentů do klauzí a určení úrovně segmentu je pouze pomocný nástroj.

Další drobnou odchylkou od předchozích popisů je přístup ke klauzím a hranicím. V případě, že se jedná pouze o hranici segmentu, je tato hranice považována za součást klauze, do které spadají segmenty z blízkého okolí hranice. V případě, že pracujeme s hranicí klauze, je tomuto objektu přiřazena tzv. nultá klauze. Tato hranice je tedy mimo systém klauzí.

Poslední odchylkou je komplexnost hranic. Teorie říká, že pokud se vyskytnou dvě hranice vedle sebe, jsou spojeny do jedné. Ovšem z implementačních důvodů jsou v anotovaných datech pouze jednoduché hranice. Existují tedy případy výskytu dvou hranic vedle sebe.

Aby bylo možné získat relevantní výsledky, bylo nutné přizpůsobit se těmto odchylkám. Přizpůsobení se nakonec ukázalo jako velmi chytrý krok, který usnadnil další práci. Každá věta je zapsána do samostatného souboru[1], ze kterého je posléze načtena a použita podle potřeby.

[1] Popis struktury rozšířeného .anx souboru v Příloze 2

6 Analýza tvorby klauzí

Přístup, kterým lze získat klauze, se odvíjí od zdroje vstupních dat. Jedna z předchozích prací ukázala, že získávání klauzí a jejich vztahů ze závislostních stromů PDT je poměrně myšlenkově přímočaré. Podle článku [7] stačí ke správnému určení klauze identifikovat přísudek. Klauze je jeho podstrom mimo podstromu jiného přísudku.

Pro tuto práci by měly být vstupem segmenty se správným rozložením indexů zanoření. V současné době neexistuje postup, který by zajišťoval získání segmentů a jejich indexů pouze na základě morfologické analýzy a povrchové struktury. Skutečným vstupem pro testy v této práci je tedy text rozdělený na věty obohacené morfologickou informací. Z tohoto vstupu jsou vytvořeny segmenty, které slouží jako vstup pro tvorbu klauzí.

V první fázi zpracování je nutné provést segmentaci. Většina kroků segmentace je pravidlově velmi dobře popsána, není tedy třeba zlepšovat efektivitu tohoto procesu. Při zpracování těchto kroků je možné dojít k optimalizování procesu.

Ve druhé fázi identifikujeme klauze a vztahy mezi nimi. Předpokladem je správné provedení celého procesu segmentace a v ideálním případě získání jednoho segmentačního schématu pro jednotlivé věty. Následuje rozdělení segmentů podle jednotlivých indexů zanoření.

6.1 Segmentace věty

Výše bylo naznačeno, že kroky segmentace lze rozdělit na dva typy. Prvním typem jsou kroky, které lze jednoduše zachytit podle pravidel, tato pravidla mají velmi vysokou úspěšnost. Úspěšnost se blíží 100 % těchto kroků. Kroky, které lze zachytit pravidly, nebo k jejichž splnění stačí projít předem nadefinovaný seznam prvků, jistě nemá cenu optimalizovat z hlediska zlepšení úspěšnosti. Při aplikaci obecně platných pravidel je možné dosáhnout optimalizace z hlediska rychlosti zpracování. Části procesu segmentace, které lze takto podchytit, jsou:

1) rozdělení věty na segmenty podle hranic segmentů,

2) přiřazení indexu při použití příznaků podřízenosti,

3) určení začátku množiny vnořených segmentů (zrcadlově – ukončování podmnožiny nadřazených segmentů),

4) omezení indexu segmentu při nalezení párové hranice.

Všechny tyto kroky jsou proveditelné pouhým porovnáním obsahu věty oproti dvěma omezeným seznamům, jejichž položky se skládají z označení skupiny slov podle tagů a konkrétních slov, které nelze zachytit pomocí tagů. Oba seznamy jsou definovány v základních pracích zabývajících se segmenty. Seznam oddělovačů vznikl z přirozených hranic mezi slovy a větami. Příznaky podřízenosti byly identifikovány na základě vztahů podřízenosti, které mezi větami vytváří. Podřadicí spojky jsou skupinou, která je založena na vlastnostech příznaků podřízenosti. Dalšími pokusy bylo zjištěno, že k příznakům podřízenosti patří i vztažná a tázací zájmena, speciální číslovky a některá zájmenná příslovce (viz kap. 5.2).

Na základě morfologických značek a přímého porovnání znaků jsou nalezeny hranice segmentů. Výstupem je rozdělení věty na dva typy objektů – segmenty a hranice. Oběma těmto typům objektů je třeba přiřadit index zanoření. Přidělení indexů probíhá z větší části na základě implicitních pravidel zavedených definicí příznaku podřízenosti.

Aplikací předchozích kroků získáme správné rozdělení na segmenty a částečné přiřazení indexů zanoření. Přesné rozdělení indexů zanoření chybí v případech, kdy segment neobsahuje příznak podřízenosti a zároveň je jeho index vyšší než 0. Podle popsaného algoritmu (v kap. 5.4) takový segment může získat plný rozptyl indexů *i* až 0, kde *i* je index předchozí hranice.

Pokud segment získá výše zmíněný rozsah indexů zanoření, má také více segmentačních schémat. Pro zredukování počtu segmentačních schémat je možné omezit tyto rozptyly na základě několika jednoduchých lingvistických pravidel. První pravidlo vychází přímo z teorie a říká, že žádný ze segmentů nesmí získat záporný index zanoření. Pokud by měl záporný index získat, je analýza zcela určitě chybná. Dalším důsledkem teorie je existence segmentů na hlavní (nulté) úrovni. Pokud po analýze věty získají všechny segmenty index zanoření hlubší než 0, je možné prohlásit, že segmentace byla opět chybná.

Je známo, že mezi segmenty jsou jisté vztahy. Zkoumáním těchto vztahů bylo nalezeno několik náznaků v povrchové struktuře, které samy o sobě nedávají plnou informaci, ale pouze její část. Tato informace může být použita pro omezení počtu možných segmentačních schémat na základě vztahů mezi segmenty. Segmenty, které mají příznaky podřízenosti, musejí být zanořeny vždy o jednu úroveň hlouběji než předchozí segment. Další omezující pravidlo je, že při výskytu čárky a spojky, před kterou se podle pravidel pravopisu čárka nepíše, má následující segment index zanoření vyšší než ten předchozí.

6.2 Tag segmentu

Pro popis vlastností segmentů byly nadefinovány v teorii speciální objekty, tzv. příznaky. Příznaky jsou tokeny ve větě, které se vyskytují v segmentu, a tím segmentu přiřazují jasnou a neměnnou vlastnost (např. vztah k předchozímu segmentu nebo vliv na následující segment). Příznak podřízenosti, který se v současné době používá, je jasným příkladem. Pro účely této práce však bylo nutné zachytit více vlastností jednotlivých segmentů. Pokud by docházelo k popisu těchto vlastností podobně, jako je definován příznak podřízenosti, získal by každý segment velmi mnoho příznaků, které by bylo nutné pro každou analýzu znovu zjišťovat. Proto je v této práci zaváděn tag pro segment i hranici. Jedná se, stejně jako v případě morfologického tagu u tokenu v PDT, o poziční tag. Každá pozice obsahuje konkrétní jednotkovou informaci. Celkově tag obsahuje komplexní informaci o každém objektu ve větě. Každý objekt, který je výstupem segmentace, má specifické vlastnosti. Každá informace v tagu je výsledkem snadno testovatelné otázky. Obsahuje objekt ku příkladu aktivní sloveso, pracujeme se segmentem nebo hranicí? Jednotlivé vlastnosti vycházející ze slov v objektu mají vliv na chování celého objektu vůči ostatním částem. Zároveň tagy segmentů mají pomoci ke skládání segmentů do klauzí. Díky tomu, že každý tag je získán pomocí přímočarých testů, při kterých nepracujeme se žádnou pravděpodobností, je možné tag použít jako zástupce objektu ve větě. Pokud by se tyto vlastnosti hledaly přímo, mohl by takový postup mít velký vliv na efektivitu testování jednotlivých segmentů na různé vlastnosti.

Tag segmentu i tag hranice segmentu je poziční, který má v tuto chvíli 6 společných složek – alfanumerické znaky. Tag segmentu by měl zachycovat nejdůležitější informace ze segmentu, které jsou důležité při určování vztahů mezi segmenty a následném slučování do klauzí.

Na první pozici se nachází rozlišení mezi segmentem a hranicí. Na této pozici je možné nalézt pouze dvě možnosti – B(oundary) nebo S(egment). Další důležitou informací pro určení klauze je, na jaké úrovni se objekt nachází. Následuje pozice s informací, zda objekt obsahuje sloveso. Další pozice vypovídá o tom, zda se v objektu vyskytuje reflexivní zájmeno. Předposlední pozice obsahuje informaci o přítomnosti příznaku podřízenosti a jeho typu. Pokud je tagovaný objekt hranice, na konci značky je obsažena informace o typu hranice. Celkový popis značek segmentu a vysvětlení možných hodnot je možné nalézt v tabulkách 3 – 6.

Pozice	Možné hodnoty	Význam
1	S | B	rozlišuje významový segment od hranice segmentu
2	0| 1 | 2 | 3…	příslušnost segmentu do úrovně zanoření (0 je považována za defaultní hodnotu)
3	C | X	informace, zda segment obsahuje aktivní sloveso (infinitiv je ignorován).
4	C | X	informace, zda segment obsahuje reflexivní sloveso
5	, | E | Q | u | 9 | J | Y | z | 1 | b | X	pokud je hodnota různá od X, segment obsahuje příznak podřízenosti a jeho hodnota odpovídá typu tohoto příznaku; hodnota je získána jako druhá pozice morfologického tagu příznaku podřízenosti
6	P | D | K | L | M | N | C | X	typ hranice

Tab. 3: Význam jednotlivých pozic značky segmentu

Hodnota	Význam
C(ontain)	hodnota označuje, že segment danou vlastnost obsahuje; pro 3. pozici obsahuje aktivní sloveso, pro 4. pozici reflexivní sloveso
X	obecně (nejen pro 3. a 4. pozici) znamená nepřítomnost hledané vlastnosti

Tab. 4: Význam hodnot pro 3. a 4. Pozici

Hodnota	Význam
,	spojka podřadicí
E, Q, J, Y	vztažné zájmeno
u	násobná číslovka
z	řadová číslovka
b	příslovce

Tab. 5: Význam hodnot pro 5. pozici

Hodnota	Význam
P	čárka ,
D	pomlčka –
C	hranice složená z více slov
J	spojka
Z	ostatní interpunkce
K	dvě spojky
L	interpunkce a spojka
M	dvě interpunkční znaménka
N	spojka a interpunkční znaménko
U	neznámá kombinace dvou slov

Tab. 6: Význam hodnot pro 6. Pozici

6.3 Odhad počtu klauzí

Aktuálně neexistuje žádný ručně anotovaný korpus, proti kterému je možné výsledky spojení evaluovat. Díky lingvisticky orientovaným jednotkám, které spojením segmentů vznikají, lze provést přibližný odhad počtu klauzí ve větě, který je založen na jednoduchém pozorování. Počet aktivních sloves je přibližně stejný jako počet klauzí. V následující kapitole je popsáno, jakým způsobem koreluje počet aktivních sloves v souvětí s počtem klauzí.

Protože klauze by měla reprezentovat jednu větu v souvětí, je odhad minimálního počtu klauzí přímočarý a toto číslo určíme jako maximum z množiny {1, počet aktivních sloves}. Při testech byly zjištěny věty, které se skládaly z mnoha segmentů, ale neobsahovaly ani jedno aktivní sloveso. Takovými vstupy mohou být nadpisy. Tento odhad vznikl pouze na základě povrchové struktury věty. V případě, že věta již prošla segmentací, je možné rozšířit množinu, ze které se vybírá, ještě o počet úrovní segmentů.

Maximální počet klauzí je omezen počtem segmentů. Ve většině případů je možné říci, že v povrchové struktuře věty nemohou existovat dvě klauze bez aktivního slovesa. V tomto pozorování nalezneme výjimky v podobně výpustek (např. Mladí ležáci, staří žebráci). Takovéto příklady by mohly vytvořit falešnou představu, že výjimku tvoří věty bez aktivního slovesa, ale stačí použít nepřímou řeč a výsledkem jsou obecné věty, které nelze lehce identifikovat.

6.4 Určení klauzí

Určení klauzí je proces, který závisí na formátu dat, ze kterých jsou klauze určovány. V kapitole 3.5 se nachází dvě definice segmentů podle vstupu do procesu. V případě anotované analytické roviny je definice rozsáhlejší, ale jasně definovaná, a proto transformace z analytické roviny může probíhat na základě jednoduchých pravidel a identifikace jednotlivých uzlů a vztahů mezi uzly v závislostním stromě.

Druhá definice vychází z povrchové roviny, přesněji z dat, která získáme aplikováním segmentace na věty na povrchové rovině. Definice pracuje se segmenty a s jejich úrovněmi. Předpokládá zároveň tři podmínky pro správné určení klauzí:

1) máme správný počet segmentů a segmenty mají správné indexy,

2) jedna klauze se může skládat pouze ze segmentů jedné úrovně,

3) segmenty a hranice segmentů mohou tvořit klauzi, pokud se nacházejí na jedné úrovni zanoření.

První bod je rozpracován v předchozích kapitolách, z nichž plyne, že většinu kroků k úspěšnému splnění prvního bodu lze ošetřit s vysokou úspěšností pomocí pravidlového přístupu, který vychází z definic. Zároveň však bylo poukázáno na to, že výskyt segmentů, u kterých může dojít ke špatnému odhadu, není zanedbatelný a je nutné tyto segmenty nějakým způsobem ošetřit.

Druhý bod lze ověřit na základě definice klauze v analytické rovině PDT, popř. na základě lingvistické intuice. Podle definice klauze pomocí PDT je klauze podstrom predikátu s výjimkou predikátových tokenů, koordinací predikátů nebo podřadicí spojky předcházející predikátům. Je možné předpokládat, že není k dispozici plná segmentace věty, ale musí existovat nějaké rozdělení klauze na segmenty. Tokeny,

které spolu v povrchové struktuře sousedí a spadají do jiných klauzí, musí být z různých segmentů. V případě, že mezi klauzemi je vztah podřízenosti, musí cesta ve stromě procházet přes predikát, který ve stromě uvozuje podřízenou klauzi. Pokud bychom vycházeli z předpokladu, že existují dva segmenty s různým indexem zanoření, které tvoří jednu klauzi, musely by existovat i klauze s různými úrovněmi, které by byly s předpokládanou klauzí koordinovány.

Třetí bod je pouze důsledkem druhého bodu.

6.5 Nalezení hranic klauzí

V případě správného nastavení indexu jednotlivým segmentům je nutné vyřešit problém s hranicemi klauze. Pokud dojde ke správnému identifikování hranic klauzí, je úkol nalezení klauzí ze segmentované věty úspěšně dokončen. Hledání hranic je možné opět rozdělit na kroky, které lze aplikovat globálně se 100% úspěšností, a na kroky, u nichž je třeba vyhodnotit, jaký přístup bude mít v jejich případě největší šanci. Následující seznam popisuje situace, v nichž je možné prohlásit o určité hranici segmentu, že je zároveň hranicí klauze:

1) začátek a konec věty jsou hranicemi klauze,
2) předěl mezi segmenty a hranicemi segmentů s různou hloubkou zanoření,

    a) následující objekt je hlouběji zanořen než předchozí objekt, tzn. objekt hlouběji zanořený je začátek nové klauze,

    b) následující objekt má nižší index zanoření než ten předchozí, tzn. předchozí objekt je posledním objektem klauze, která závisí na předchozí klauzi o úroveň výše,

    c) v situaci popsané v bodě a) má objekt s nižším indexem zanoření možnost spojit se s následujícím objektem na dané úrovni, pokud není jeden z objektů určen jako hranice klauze,

    d) v situaci popsané v bodě b) má objekt s nižším indexem zanoření možnost spojit se s předchozím objektem na stejné úrovni, pokud není jeden z objektů určen jako hranice klauze,

Obr. 7: Schéma možného uvození přímé řeči

e) výjimku tvoří věty obsahující přímou řeč. Tyto věty mohou obsahovat podobnou konstrukci (viz Obr. 7).

Oba výše uvedené body je možné analyzovat na základě správně provedené segmentace věty. K dokončení identifikace klauzí musí dojít k nalezení hranic klauzí na jedné úrovni. Hlavní problém plyne z definice hranic klauze a nedostatku sémantických informací. Hranice klauze vznikají v povrchové struktuře z některých hranic segmentů změnou (prohloubením) jejich oddělovací funkce. Snaha o odhad hranic klauzí znamená snahu o nalezení syntaktických hranic mezi podstromy klauzí (Obr. 8). V této práci nejsou k dispozici hlubší informace z vyšších rovin, protože výsledek této práce by měl sloužit jako předzpracovaný vstup pro syntaktickou analýzu. Problémy nastávají v několika případech:

1) koordinace členská nebo větná,

2) rozlišení různých specifických jazykových konstrukcí (apozice, vsuvka, výpustka).

V obou případech se jedná o jevy snadno řešitelné, pokud jsou k dispozici data z hlubší analýzy věty. V případě této práce jsou tyto jevy velmi těžko rozeznatelné. V některých případech lze však objevit jisté náznaky a jisté jazykové vlastnosti, které mohou napomoci řešení jednotlivých problémů.

Koordinace segmentů znamená, že mezi dvěma segmenty je hranice a všechny tyto objekty mají stejný index zanoření. Správné analyzování výše zmíněných jevů ve větě znamená předzpracování pro analytickou rovinu a odhadnutí, které části v povrchové struktuře označují samostatné podstromy. Pro výše zmíněné jevy je

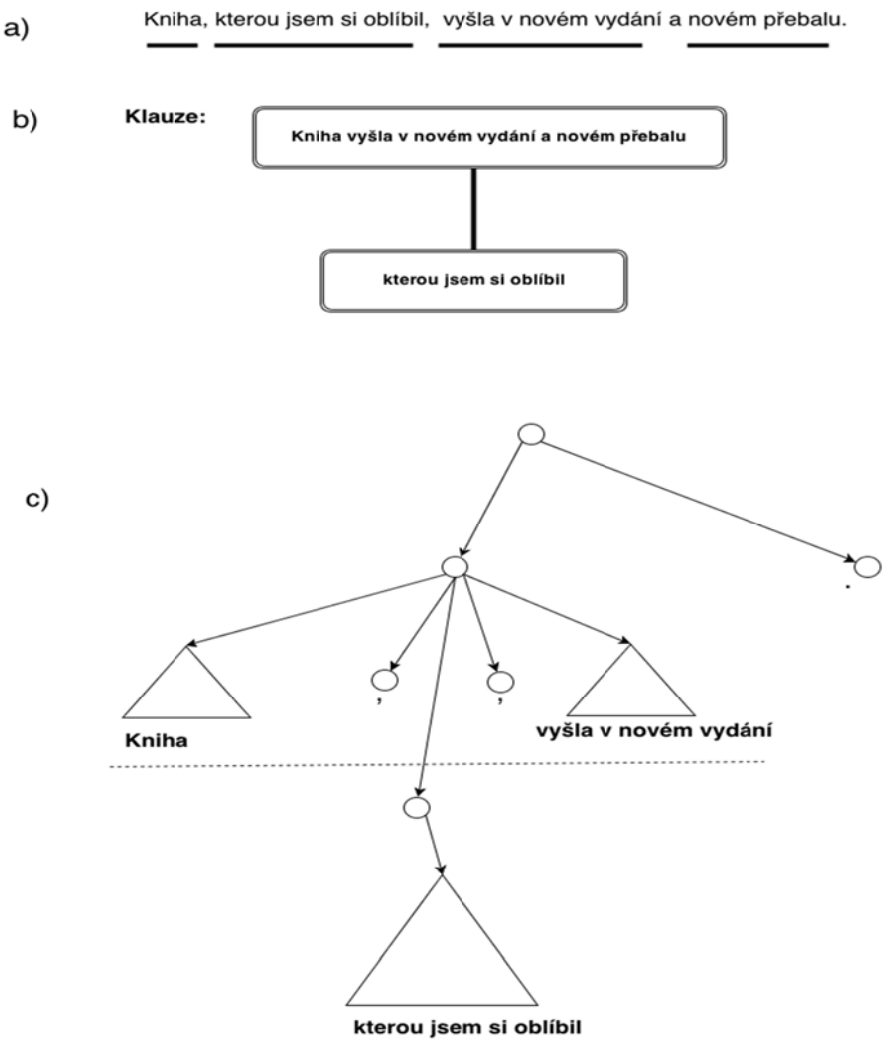
možné nalézt jisté nápovědy, které umožňují vyloučit, že provádíme analýzu právě některého z popsaných jevů.

První nápovědou jsou aktivní slovesa. Pokud se v koordinaci vyskytnou dva segmenty a oba obsahují aktivní sloveso, jedná se o dvě různé klauze. V takovém případě je možné uzavřít zároveň předchozí klauzi pro další připojování segmentů.

Dalším signálem, jak se zachovat při spojování segmentů, které jsou v povrchové struktuře vzdálené, je detekce reflexivních sloves. Předpokladem je, že sloveso a zvratné zájmeno se nacházejí ve vzdálených segmentech.

Další pomůckou může být pozorování, že segmenty, které obsahují pouze gerundium, lokál a instrumentál, jsou často pouze částí klauze a netvoří samostatnou klauzi, protože v těchto pádech se často nacházejí dodatečné informace k celkové výpovědi.

Dalším zdrojem informací může být valenční slovník Vallex, ze kterého budou získány pouze pády jednotlivých členů valenčního rámce. Přiřazení jednotlivých členů klauze k členům valenčního rámce je úkol velmi náročný, proto může být Vallex využit opačným směrem. Pokud se v segmentu objeví tag ohebného slovního druhu, jehož pád není možné přiřadit k žádnému valenčnímu rámci konkrétního slovesa, segment s tímto slovem nepatří do klauze, ve které je testované sloveso.



Obr. 8: Ukázka postupu zpracování segmentů a klauzí na jedné větě.

a) rozdělení věty na segmenty

b) rozdělení věty na klauze

c) závislostní strom (přerušovaná čára odděluje segmenty)

7 Analýza přístupů

Kapitola 6 je věnována především částem, které lze zpracovat na základě teorie nebo obecných pravidel. Pokud by vstup (věta obohacená o morfologické informace) byl zpracován předchozí analýzou, získali bychom sadu segmentačních schémat a dvě pomyslné funkce, např. *f, g*. Funkce *f* zpracovává větu rozdělenou na segmenty a jedno segmentační schéma. Výsledkem jsou hranice klauzí ve větě. Funkce *g* zpracovává větu, set segmentačních schémat a funkci *f*, na jejímž konci získáme klauze.

Následující přístupy tedy zkoumají, jak získat ještě menší počet segmentačních schémat pro funkci *g* (v ideálním případě jedno schéma) a jak nadefinovat funkci *f* (jakým způsobem budou označeny klauze).

7.1 Pravidlový přístup

Z analýzy obecného přístupu je možné určit hlavní problémy, které bude tato práce řešit. Dosud byla pravidla dělena na pravidla, která vychází přímo z teorie, a pravidla, která jsou získána pozorováním a tvořením statistik jednotlivých jevů, které byly prováděny nad kolekcemi dat. Všechna pravidla lze rozdělit do čtyř skupin:

	Teorie	**Analýza**
Segmenty	počáteční segment a závěrečná hranice	příznaky podřízenosti
Klauze	klauze může obsahovat pouze segmenty jedné úrovně	segmenty obsahující aktivní slovesa jsou segmenty různých klauzí

Tab. 7: Přehled získání pravidel

Některá pravidla jsou implicitní a není třeba je speciálně implementovat. Například přiřazením hlavní (nulté) úrovně poslednímu objektu ve větě získáme téměř 100% pokrytí. Pokud toto implicitní pravidlo rozšíříme pouze na ukončovací interpunkční znaménka, získáme 100% pokrytí. Pro pravidla, která pro svou aplikaci

vyžadují pouze triviální lokální podmínky (např. jen porovnání slovního tvaru nebo tagu), není třeba zavádět seznam pravidel, ale efektivnější je implementovat tato pravidla přímo do segmenteru nebo analyzátoru klauzí. Toto platí zejména pro seznam hranic a seznam příznaků a jejich vlastností. Zatímco slovní tvary a tagy se mohou měnit nebo rozrůstat (např. v případě aplikace na nový jazyk), vlastnosti nadřazených objektů jsou konstantní. Na základě této argumentace je možné seznamy objektů načítat dynamicky.

Dalším typem pravidel jsou taková, která platí, ale jsou jazykově závislá nebo jejich aplikace vyžaduje příliš globální nadhled. Příkladem může být čárka před některou ze souřadicích spojek {a, i, nebo}, pokud nenásleduje za první souřadicí spojkou další spojka. Tato konstrukce obvykle znamená, že další segment je v úrovni zanoření výše, ale díky splývání čárek pro jednotlivé segmenty je jen na základě morfologických informací těžké říci, kolik úrovní bylo takto přeskočeno. Tento typ pravidel se může velmi často měnit, a proto by bylo velmi vhodné vytvořit formalismus, ve kterém by se mohla pravidla jednoduše zapsat. Při analýze tohoto formalismu jsem však došel k závěru, že nároky na něj kladené by byly mnohem vyšší než přímá aplikace pravidel ve zdrojovém kódu. Problém byl především v granularitě, kdy byl hlavním objektem zájmu segment, což funguje v jednoduchých případech. Například dvě zdánlivě jednoduché věty:

a) ***Petr a Pavel šli do kina.***
b) ***Petr, který má v oblibě české filmy, a Pavel, kterému se líbí spíše zahraniční produkce, šli do kina.***

Zatímco pro větu a) pravidlo vyjádřené pomocí xml formalismu není problém si představit. Pro větu b) je to sice možné, ale při tvorbě pravidel pro složitější typ vět jsem narážel na dva extrémy. V prvním případě jsem pravidlo vytvořil příliš konkrétní. Vytvoření pravidla padnoucího přímo na celou větu a přidáním nevýznamného segmentu nebo hranice by se muselo reflektovat přidáním výjimky nebo nového pravidla. V druhém případě bylo pravidlo naopak příliš obecné a dokázalo pojmout i jevy, které by jasně měla řešit jiná pravidla.

Obsáhlost pravidel je jeden z problémů. Dalšími jsou pak implementace a vlastní použití formalismu. Implementací se jedná o syntaktický analyzátor pro gramatiku

typu 0. Složitost použití formalismu jsem nakonec shledal stejně náročnou jako úpravu pravidel implementovaných přímo v kódu.

Implementace těchto pravidel v kódu je ukryta v několika objektech, ale jsou shluknuty do jednoho místa. Navíc díky volbě jazyka, ve kterém jsou metody implementovány, je velmi snadné rozšířit rozsah pravidel zasažením přímo do kódu a zároveň přímo ověřit účinnost nového pravidla. Implementace v kódu je variabilní a může ovlivnit jak segmenty, tak celé klauze. Pokud bychom se snažili vytvořit formalismus, z větší části by musel plnit také úkol segmentace. V takovém případě by se formalismus musel nejdříve vyrovnat se samotnými segmenty a poté i s klauzemi. Vytvoření samostatného dosti pružného formalismu pro segmenty i klauze se nakonec ukázalo jako úkol, který by mohl být zpracován jako samostatná práce. Z uživatelského hlediska použití takového analyzátoru by se nejspíše zdálo jako velmi vhodné vytvořit formalismus pro jednoduché testy. Již první náznaky (po kterých jsem formalismus zamítl) ukázaly, že vytvoření nového pravidla v navrhovaném formalismu by bylo složitější než ho napsat ve zvoleném implementačním jazyce. Tento názor zastávám i proto, že problém je zatím zkoumán hlavně z pohledu efektivity a možnosti dalšího využití jako heuristika pro složitější lingvistické úlohy.

Pokud by zvolený postup kopíroval teoretické postupy, vznikly by celkem čtyři typy pravidel, které by kopírovaly rozložení pravidel podle tabulky 7. V některých případech by podmínky mohly být duplicitní, což by vedlo k neefektivnosti.

Pravidla rozdělíme podle jejich vlivu na klauze a segmenty, nebo pouze na segmenty (odpovídá řádkům v tabulce 7). Pravidla, která ovlivňují zanoření segmentů, jsou většinou pouze omezujícími prostředky. Ve výsledku získáme pomocí těchto pravidel co nejmenší počet segmentačních schémat, z nichž jedno je jistě správné.

V druhé fázi jsou aplikována pravidla, která zajišťují především spojování segmentů do klauzí nebo nalezení hranic klauzí. Tato pravidla by měla především zajišťovat spojení a rozdělení klauzí. V případě nemožnosti aplikace pravidla na následující segment je nutné segmentu zajistit přiřazení jiného indexu zanoření, jinými slovy prověřit jiné segmentační schéma věty.

Věta	Počet segmentů	Počet klauzí
Jan a Jana se, pokud vím, *vzali letos v červnu.*	3	2
Jan a Jana se, pokud vím **a** nic závažného se nestalo, *vzali letos v červnu.*	4	3
Jan a Jana se, pokud jsou mé zdroje aktuální, přesné **a** pravdivé, *vzali letos v červnu.*	5	2

Tab. 8: Ukázka „pumpování" pravidel. Tučně jsou zvýrazněné hranice segmentů, okrajové segmenty vyznačené kurzívou tvoří jednu (hlavní) klauzi

Pokud pravidlo identifikuje segment nebo skupinu segmentů, na který má být aplikováno, je dalším krokem předání informace o tom, jak se zachovat oproti následujícím i předchozím segmentům. Tato informace je součástí pravidla a výsledných efektů pomocí jednoho pravidla může být více. Ve formalismu jsou navrženy příznaky, které určují efekt ve větě.

7.2 Pravděpodobnostní přístup

Segmentace i nalézání klauzí pomocí segmentů se jeví z předchozích analýz jako problém, pro který se jiný přístup příliš nehodí. V jazyce však existují jevy, které bez hlubších znalostí vypadají velmi podobně, avšak jejich chování z pohledu určení hloubky zanoření nebo klauze, do které patří, je velmi odlišné. Například již při prvních kontrolách se zdály být problémové věty s pomlčkou. Pokud by přišlo na zkoumání pouze tohoto jevu, je často velmi různorodý. V tabulce 9 vidíme, k jakým změnám dochází mezi dvěma segmenty oddělenými pomlčkou.

	0	**1**	**-1**	**2**	**-2**	**zbylé rozdíly**
Rozdíl úrovně zanoření	176	25	41	2	1	0
Rozdíl klauzí	207	11	16	2	4	5

Tab. 9: Rozdíly mezi segmenty oddělené pomlčkou

Z teoretického pohledu je jistě možné, aby jev, který je pozorován na datech, měl jisté odchylky. Dalším faktorem, je „selhání" předchozích systémů. V případě zanoření to může být špatně přiřazený tag. Několikrát jsem nalezl v datech slovíčka

„až“ a „než“ označené jako spojky podřadicí, přestože byly použity v jiném významu. Z pohledu klauzí je to chybné určení úrovně segmentů, u něhož také dochází ke ztrátě spolehlivosti při zpracování delších vět.

Řešení pomocí rozdělení segmentů do skupin a učení se podobně jako při klasifikaci není v tomto případě příliš šťastné. Zatímco v klasifikačních problémech záleží čistě na vlastnostech objektu a historie je v takových problémech nevýznamná, v našem případě je plná historie důležitá vlastnost segmentu a její zanedbání značně přispívá k neúspěchu.

Z výše uvedených důvodů jsem se rozhodl nahlížet na oba problémy podobně jako na problém značkování, tedy nalézt celkově nejvíce pravděpodobnou sestavu úrovní zanoření, popř. klauzí. Pro tento typ problému se nejlépe hodí skryté markovovské modely.

V rámci trénování jistě nemůžeme použít celé segmenty. Místo nich byl použit vyvinutý systém tagů segmentu (viz kap. 6.2). Tyto tagy věrně reprezentují obsah jednotlivých segmentů a jejich vlastností a je možné je při trénování plně zaměnit za segmenty.

Mezi pravděpodobnostním a pravidlovým přístupem se už během analýzy ukázaly některé podstatné rozdíly v přístupu. V pravidlovém přístupu lze podle předchozí kapitoly určit úroveň zanoření i číslo klauze v rámci jednoho zpracování. Úroveň zanoření lze v pravidlovém systému chápat jako zkratku za několik pravidel. Při práci s pravděpodobnostními metodami by tento postup mohl být také použit, ale v takovém případě by byla doména jevů poměrně omezující. Rozložením na dva systémy zvýšíme jejich variabilitu, protože tímto rozložením získáme i pravděpodobnosti dvojic, které nebyly v trénovacích datech pozorovány.

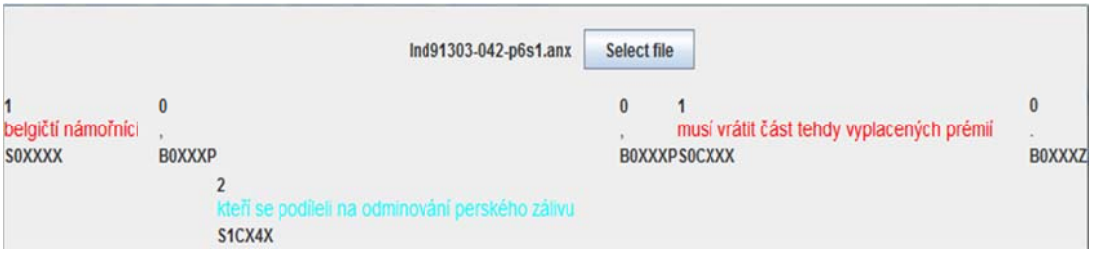
Druhým rozdílem je způsob, jakým určujeme jednotlivé vlastnosti. Při pravidlovém přístupu určí pravidlo vztah mezi segmenty a z kombinace aktuální hodnoty zkoumané vlastnosti a vztahu získáme novou hodnotu. U zvoleného pravděpodobnostního přístupu je možné nahlížet na určenou sekvenci dvěma způsoby. V prvním případě odhadujeme přímo hodnotu hledané vlastnosti. V takovém případě získávají především jevy, které se již vyskytly. V případě věty delší (z pohledu počtu klauzí), než byly pozorovány v testovacích datech, je

výsledek velmi nejistý. Čím vyšší je číslo klauze, tím nižší pravděpodobnost má. I když v pravděpodobnostním modelu použiji vyhlazování, určitě se nalezne věta, která bude obsahovat vyšší číslo klauzí, než bylo v datech pro učení. Druhým způsobem, jak odhadnout novou klauzi, je pouze analyzovat vztah mezi segmenty, potažmo klauzemi. V takovém případě je nutno odhadnout, zda spadá do stejné nebo jiné úrovně zanoření. V případě zjištění, že segmenty spadají do jiné úrovně zanoření, je podle teorie téměř jasné, že se jedná o dvě různé klauze. Existují však výjimky v podobně klauzí, které jsou rozděleny pomocí jiných klauzí. Zároveň je nutné rozložit segmenty na jedné úrovni zanoření do různých klauzí (např. spojení segmentů obsahuje dvě slovesa vyjadřující děj). Při pravděpodobnostním přístupu bude vstup analyzován dvakrát. V prvním kroku zjišťuji úrovně zanoření segmentů a v druhém kroku rozložení klauzí.

8 Implementace

Na začátku jsem plánoval vyvinout dva systémy, které budou využívat společný základ, který větu předzpracuje, a jeho výstup použiji jako svůj vstup, který jej upraví do výsledné podoby a vrátí jako výsledek. Přestože jsem během implementace společného základu dosáhl na testovacích datech poměrně dobré úspěšnosti, chtěl jsem přistoupit k problému i s čistě pravděpodobnostním algoritmem bez pravidlových úprav. Ze společného základu se postupně vyvinula implementace pravidlového přístupu.

Pravidlový i pravděpodobnostní přístup mají společné rysy, díky kterým je možné oba přístupy lépe porovnat. Hlavním společným rysem je rozdělení odhadu na pomocný modul, který odhaduje úroveň zanoření a přispívá k práci hlavního modulu, který určuje klauze. Dalším rysem je označování hloubky zanoření segmentů a číslování klauzí. Z teoretického náhledu vyplývá, že hloubka zanoření je myšlené číslo, které určuje vzdálenost od hlavní výpovědi. Hlavní výpověď má hloubku 0. Klauze jsou číslovány vzestupně od 1 dle prvního segmentu.



Obr. 9: Grafický výstup - určení klauzí

Číslování probíhá od 1, existuje však nultá klauze, kam spadají především hranice klauzí. Jedná se spíše o technický detail, který byl převzat z PDT 2.5.

Implementace probíhala v IDE Eclipse v jazyce Scala.

8.1 Implementace společného základu

Společným základem měl být první jednoduchý odhad, který je založen na základních pravidlech. Tato pravidla měla být podle prvních analýz snadněji implementovatelná než pomocí předpokládaného formalismu a měla být neměnná.

Nakonec se však ukázalo, že těmto podmínkám vyhovují všechna pravidla, která jsem dokázal vypozorovat, a proto modul společného základu plně splynul s modulem pravidlového přístupu.

Na základě prvních analýz jsem předpokládal, že pro pravidlový přístup bude mít vyšší význam segmentace a určování úrovně zanoření, proto je modul rozdělen na dva podmoduly. Prvním podmodulem je odhadnuta úroveň zanoření jednotlivých segmentů. Druhým modulem se odhaduje příslušnost segmentu ke klauzi. Tyto odhady jsou založeny na podmínkách, které se soustředí především na blízké okolí, max. dva segmenty. Vytvořil jsem ovšem také pravidla, která musí reagovat na aktuální klauzi, nebo dokonce na zbytek zpracovávané věty.

Modul, který odhaduje úrovně zanoření segmentů, pracuje pouze s informací, zda segment má příznak podřízenosti, zda jsou segmenty uzavřeny v závorkách nebo zda je segment poslední. Naopak původní označení vsuvky pomocí pomlčky vytvoří více chyb než správně označených segmentů. Následující tabulka (Tab. 10) ukazuje, jak rostla úspěšnost určení správné úrovně zanoření během vývoje, kdy byla do implementace postupně zapojována nová a nová pravidla. Už na této tabulce je vidět několik zajímavých jevů. Například pravidlo, které má jasně lingvisticky motivované pozadí (druhé pravidlo v tabulce 11), má poměrně malý vliv na počet úspěšně určených segmentů, naopak úspěšnost na celých větách prudce stoupla. Takovéto a podobné jevy se objevovaly během celé doby vývoje pravidlové části.

Výsledky během vývoje:

	Zapojené pravidlo	**Úspěšné určení segmentů**	**Úspěšné určení celých vět**
1	základní odhad	79.5 %	56.4 %
2	segment před souřadicí spojkou má příznak podřízenosti, úroveň zanoření se nemění	79.95 %	58.2 %
3	první objekt je hranice, získá úroveň 0	81.69 %	62.2 %
4	použití slova „jak“ jako příznak podřízenosti	81.59 %	62.6 %
5	použití závorek	82.16 %	63.4 %

Tab. 10: Porovnání postupného zapojování pravidel pro úroveň zanoření

Modul, který odhaduje příslušnost segmentů ke klauzím, je o něco složitější. I tak pokrývá především ta nejzákladnější pravidla. Například pokud dva sousední

segmenty (nikoliv hranice) obsahují slovesa, patří tyto segmenty do dvou různých klauzí. Podobně pokud aktuální segment obsahuje příznak podřízenosti, patří tento segment do jiné klauze než segment předchozí.

Pohled na tabulku 11 ukazuje, že prvotní odhad úspěšnosti byl o téměř 10 % horší v celkovém počtu segmentů, ale úspěšnost v rámci celých vět byla velmi nízká (hned prvním pravidlem úspěšnost v rámci vět prudce vzroste).

	Zapojené pravidlo	Úspěšné určení segmentů	Úspěšné určení celých vět
1	základní odhad	66.02 %	17 %
2	pokud nejde o hranici klauze, je hranice součástí klauze	79.73 %	56.2 %
3	hranice na začátku věty spadá do první klauze	80.02 %	56.2 %
4	v případě objektu na začátku věty, je klauze vždy 1	82.36 %	59.2 %
5	pokud je první objekt hranice s tagem „Z:“, nastav klauzi ns 0	82.88 %	62 %

Tab. 11: Porovnání postupného zapojování pravidel pro spojování segmentů[2]

Při této fázi vývoje se vkládala pravidla, která platí vždy (např. příznak podřízenosti nebo zvýšení klauze při zanoření), a především pravidla techničtějšího rázu (viz pravidla v kapitole 5).

8.2 Implementace pravidlového přístupu

V teoretické části bylo ukázáno, že pokud jsem schopen nalézt správné zanoření hranic, neměl by být problém vytvořit správné rozložení na klauze. Naopak problémem jsou konce úrovní jednotlivých segmentů. Proto místo pravidel v xml formě jsem zvolil postup rozšíření jednoduchého společného základu a využití xml pravidla pouze na změnu úrovně zanoření segmentů.

Samotná implementace pravidlového přístupu se podobně jako při implementaci společného základu skládá ze dvou částí. První je určení hloubky zanoření jednotlivých segmentů a jejich následné spojování. Část úkolu, která se zabývá

[2]Spojování segmentů v tomto případě znamená určení správné klauze každého segmentu.

odhadem úrovně zanoření, není hlavní částí této práce, a proto se od této části neočekávají výraznější výsledky. Ale jak ukázaly výsledky, existuje mezi úspěšností tohoto modulu a celkovou úspěšností programu jasná relace. Hlavním důvodem vlastní implementace tohoto úkolu byla absence modulu, který by tento úkol řešil za daných podmínek. Na základě výsledků této analýzy jsou poté pospojovány jednotlivé segmenty do klauzí.

Postup odhadování úrovně zanoření probíhá rekurzivně, s občasnou zpětnou i dopřednou kontrolou blízkého okolí aktuálně zkoumaného segmentu. Skládá se z následujících pravidel:

1) Jedná-li se hranici, která je na konci věty (poslední segment), úroveň nastav na 0.

2) Jedná-li se o hranici:

a) následující segment obsahuje příznak podřízenosti, aktuální hranice je „„“, v předchozím a následujícím segmentu nacházím příznak podřízenosti a tento příznak není příliš vzdálen od čárky – dochází k zanoření.

b) následující segment obsahuje příznak podřízenosti, ale neplatí podmínky z a) – zůstávám na aktuální úrovni zanoření.

c) aktuální hranice je „„“, další je souřadicí spojka „a“ nebo „i“, následující segment má příznak podřízenosti – úroveň zůstává zachována. V případě neplatnosti poslední podmínky, je úroveň zanoření snížena o 1.

d) aktuální hranice je „„ “, předchozí segment měl příznak podřízenosti a zároveň předchozí i následující segment obsahují aktivní slovesa – úroveň zanoření je snížena o 1.

e) aktuální hranice je „„ “ a na této úrovni již byla použita souřadicí spojka – úroveň je snížena o 1.

f) v ostatních případech je úroveň zanoření stejná jako u předchozího segmentu.

3) Nejedná-li se o hranici:

a) obsahuje příznak podřízenosti – úroveň je zvýšena o 1.

b) předchozí segment je hranice, úrovně zanoření předchozího a aktuálního segmentu jsou různé a aktuální segment obsahuje sloveso předchozímu segmentu – nastav úroveň zanoření na 0.

c) úroveň zanoření je stejná jako u předchozího segmentu.

Předpokladem použití těchto pravidel je implicitní nastavení počátečního segmentu na 0 a nemožnost při snižování úrovně přejít přes 0. Dalším předpokladem je zachování pořadí pravidel (např. pravidla 2c – 2e). Dalším krokem (již neimplementovaným) ke zlepšování tohoto systému je možnost odhalení chyby v případném pokusu a snížení úrovně zanoření pod 0. I přes tato jednoduchá pravidla jsem dosáhl poměrně dobrých výsledků, které pomohou spojování segmentů do klauzí.

Celý pravidlový postup je rozdělen na dvě fáze. V první fázi zkoušíme přiřadit jednotlivým segmentům čísla klauzí. Druhá zkontroluje konzistenci klauzí ve větě, tedy zda každá nalezená klauze má sloveso. V tomto pravidle jsou povoleny pouze dvě výjimky. Na nultou klauzi se pravidlo nevztahuje, protože tato klauze nemá sloveso už ze své podstaty (viz začátek kapitoly), a pravidlo se neuplatňuje v případě, že celá výpověď nemá sloveso. V tomto případě jsou všechny segmenty vloženy do jedné klauze s číslem 1.

Pravidla uplatňující se při odhadu klauzí:

1) aktuální segment je hranice:

a) je první a je to otvírací závorka – aktuální segment je vložen do nulté klauze,

b) je otevírací nebo zavírací závorka (nebo se jedná o segment uvnitř závorky),

1. pokud vnitřní segmenty obsahují sloveso, zvýší se číslo klauze o 1 a samotné oddělovače jsou vloženy do klauze 0,

2. v jiném případě ponech aktuální klauzi,

c) aktuální hranice je spojka, před kterou se nepíše čárka, předchozí slovo byla „„“ a následuje segment s příznakem podřízenosti – aktuální segment spadá do aktuální klauze,

d) aktuální hranice není na začátku – patří do nulté klauze.

e) je na začátku a jedná se o interpunkci – spadá do nulté klauze.

2) aktuální segment není hranice:

a) má příznak podřízenosti a předchozí hraniční segment není „a “ nebo „ i “,

b) předchozí segment byla hranice a předchozí i aktuální jsou na stejné úrovni zanoření,

1. je-li možné aktuální segment vložit do aktuální klauze, aktuální segment spadá do aktuální klauze,

2. není-li možné aktuální segment vložit do aktuální klauze, všechny hranice mezi aktuálním segmentem a předchozím aktuálním segmentem jsou vloženy do nulté klauze a je založena nová klauze,

c) předchozí nehraniční segment má vyšší úroveň zanoření a předchozí segment není pomlčka, zároveň jsou opět všechny hranice mezi segmenty vloženy do nulté klauze,

1. je-li možné aktuální segment vložit do nadřazené klauze, aktuální segment je vložen do nadřazené klauze,

2. není-li možné je sloučit založ novou klauzi,

d) předchozí nehraniční segment má nižší úroveň zanoření je nutné založit nový segment,

3) v nepopsaných případech je segment součástí aktuální klauze.

Ve výše uvedeném seznamu je jednou z vlastností informace o tom, zda je

možné spojit dva segmenty, které nejsou hranice (pravidla 2b a 2c). Tato podmínka v sobě obsahuje několik jazykových jevů, které omezují možnost vložit segment do aktuální klauze.

Pravidla pro spojení dvou segmentů:

1) V případě, že číslo klauze, se kterou má být segment spojen, je 0, nebo větší než aktuální maximum, nebo nemáme žádnou klauzi, je pokus zamítnut.

2) Po nalezení klauze,

a) nesmí být klauze uzavřená – nesmí tedy na stejné úrovni začínat pozdější klauze,

b) nesmí mít dvě slovesa v aktivním tvaru – výjimku tvoří podmiňovací způsob a minulý čas se slovesem „být“,

c) při spojení klauzí nesmí dojít k jasné neshodě mezi podmětem a přísudkem,

1. v případě slovesa vyžadujícího první nebo druhou osobu, je hledána shoda, jen pokud existuje zájmeno, které může být považováno za podmět (kontroluje se i shoda čísla),

2. pokud sloveso vyžaduje třetí osobu v singuláru a pokud se v klauzi vyskytuje slovo v nominativu, musí se shodovat v rodě a čísle,

3. pokud sloveso vyžaduje třetí osobu v plurálu, pak pokud existuje pouze jeden předpokládaný subjekt, musí být v plurálu nebo je třeba shoda v rodě,

4. pokud sloveso vyžaduje třetí osobu, ale neexistuje žádné zájmeno nebo substantivum v předpokládaných subjektech.

8.2.1. Proč ne formalismus?

Původní návrh pravidlového přístupu předpokládal existenci pravidel v nějaké snadno modifikovatelné struktuře (XML). Hlavním důvodem tohoto návrhu byla možnost jednoduše vkládat nová pravidla, aniž by bylo potřeba znát zdrojový kód nebo programovací jazyk, a zároveň zvýšit čitelnosti jednotlivých pravidel.

Během analýz se ukázal můj přístup k formalismu jako velmi naivní. Především pokud se jednalo o granularitu, kdy navržený formalismus počítal spíše se segmenty, ale pozorování ukázalo, že je třeba pracovat i s klauzemi a skupinami klauzí. Takováto pravidla by se velmi obtížně zapisovala čistě jen pomocí segmentů. Avšak kromě možností zápisu jednotlivých vzorů rostl také repertoár jednotlivých efektů. Tímto způsobem by narůstala složitost jak zápisu pravidel, tak analyzátoru těchto pravidel.

Zkušenost dále ukázala, že nové pravidlo by většinou vyžadovalo alespoň drobnou změnu implementace, protože se týkalo vlastnosti, která nebyla zohledněna, popř. došlo ke změně náhledu na danou vlastnost.

8.2.2. Implementace pravděpodobnostního přístupu

Z analýzy pravděpodobnostního přístupu (kap. 6.2) vyplývá i jeho přímá implementace. Základem jsou dva jazykové modely, které je možné trénovat. První model, unigramový, reprezentuje pravděpodobnost, že tag segmentu se vyskytne u příslušné skryté vlastnosti (úroveň zanoření, klauze, rozdíl úrovní klauze, rozdíl klauzí). Druhý model, bigramový, reflektuje „historii“ segmentů ve větě, snaží se tedy zohlednit segmenty, které mu předcházely.

Modely jsou vyhlazovány pomocí jednoduché lineární interpolace, kdy byly jednotlivé parametry nastaveny tak, aby největší váhy získaly jevy, které byly během učení nalezeny.

Dalším prvkem je implementace Viterbiho algoritmu. Problémem při prvních testech bylo příliš mnoho cest, které bylo nutné prověřit, díky nimž došlo k přetečení bufferu. Proto bylo nutné prořezávat mnohem „přísněji“ a prověřovat méně cest s největší pravděpodobností na úspěch. Kromě základního prořezávání je v modelu

pro klauze použita technika, která je založena na číslování, které může být na rozdíl od úrovně zanoření poměrně omezené (pokud aktuálně zpracovávám první klauzi není možné přeskočit do třetí).

Implementovány byly dva modely, které byly trénovány s jinou sémantikou. V první fázi předpokládáme číselné označení úrovně zanoření a klauze. Správné určení úrovně zanoření segmentu a jeho příslušnost ke klauzi je podmíněno především předchozími segmenty. Tento model by měl mít poměrně vysokou úspěšnost při kratších větách (cca max. 4 až 5 klauzí). Těchto krátkých vět je v dostupných datech nejvíce.

8.2.3. Rozšíření pravděpodobnostního přístupu

První výsledky pravděpodobnostního přístupu nebyly příliš úspěšné. V extrémním případě byla úspěšnost nižší než při prvním odhadu založeném pouze na úrovních segmentů (viz Tab. 13 a Tab. 15).

Krok, kterým je možné zlepšit výsledky pravděpodobnostního přístupu, aniž bychom měnili jeho parametry, je rozšíření sledovaných vlastností, tzn. rozšíření značky pro segment. Toto rozšíření vychází z podobné myšlenky na jaké je postavený slovník Vallex (viz [11]), který obsahuje slovesné rámce pro česká slovesa. Vallex zkoumá, jaká doplnění mají jednotlivá slovesa. Tato doplnění jsou povinná a nepovinná (obligatorní a fakultativní).

Myšlenka rozšíření odpovídá předpokladu, že každé sloveso má základní rámec: podmět, přísudek a předmět. Základním pádem pro podmět je 1. pád (nominativ), pro předmět 3. a 4. pád (dativ a akuzativ). Pozorováním bylo zjištěno, že může záležet i na poloze slova v rámci segmentu. Například pokud první segment končí a druhý segment začíná slovem, které má shodu v pádu, je pravděpodobnější, že půjde o segmenty ze stejné klauze. Celkově jsem značku rozšířil o tři nové pozice. Určují, zda segment obsahuje slovo s daným pádem a na jaké relativní pozici se slovo v rámci segmentu nachází.

Nové pozice značky segmentu:

- 7. znak pro slovo s 1. pádem (nominativ)
- 8. znak pro slovo s 3. pádem (dativ)
- 9. znak pro slovo s 4. pádem (akuzativ)

Na těchto pozicích je poté možno najít tyto hodnoty:

- „3“ – segment obsahuje jedno slovo s daným pádem.
- „2“ – poslední slovo s daným pádem je na konci segmentu.
- „1“ – první slovo s daným pádem je zároveň první slovo segmentu.
- „0“ – slova s daným pádem jsou mimo krajní pozice v segmentu.
- „X“ – v segmentu neexistuje slovo s daným pádem.

8.3 Sloučení pravděpodobnostního přístupu s pravidlovým

Sloučení pravděpodobnostního přístupu s pravidlovým je možné provést několika způsoby. Nejjednodušší způsob je oprava výsledků pravděpodobnostního přístupu. Další možností je využití vztahů mezi nejbližšími segmenty a informace o tomto vztahu použít pro učení (konkrétně využití rozdílu v klauzích mezi jednotlivými významovými segmenty). Pro určení jsou důležitá obecně platná pravidla (viz. kap. 8.3.2).

8.3.1. Oprava statistického přístupu

Samotný pravděpodobnostní přístup nebyl příliš úspěšný, na rozdíl od pravidlového. Pozorováním výsledků z pravděpodobnostního přístupu se ukázalo, že některé výsledky nemohou být považovány za správné, i když neznáme větu, ale pouze výsledek. Jedná se tedy spíše o opravu výsledků. Pokud čísla v závorkách označují klauzi a počet číslic počet segmentů věty, získáme podobný objekt jako je segmentační schéma, ale místo úrovní získáváme informace o klauzích. Pokud máme schéma např. (1, 0, 2, 0, 5, 0), máme šest segmentů, z nichž tři jsou zároveň klauze

a tři jsou hranicemi segmentů. Z definice označování segmentů je jasné, že předchozí schéma je chybné a mělo by jít nejspíše o schéma (1, 0, 2, 0, 3). Na základě podobných pozorování byla vytvořena čtyři jednoduchá pravidla, která mohou opravit výstup primárně z pravděpodobnostního přístupu.

Pravidla pro opravu schémat klauzí:

1) pokud číslování klauzí netvoří nepřerušovanou číselnou řadu, je nutné klauze přeznačit tak, aby číselnou řadu tvořily (viz příklad v předchozím odstavci),

2) hranice klauzí spadá do speciální nulté klauze, ale hranice mezi segmenty uvnitř klauze získají stejné číslo klauze jako ta, která je obklopuje,

3) segmenty s různou úrovní zanoření nesmí být v jedné klauzi v rámci jedné věty. V některých případech pravděpodobnostní algoritmus nerozeznal hranici tvořenou různou úrovní zanoření,

4) klauze může být rozložena klauzí nebo klauzemi pouze s vyšší číselnou hodnotou, než má sama.

Příklad pro pravidlo č. 1
Chybné schéma klauzí (1, 0, 2, 0, 5, 0)
Správné schéma klauzí (1, 0, 2, 0, 3, 0)

Příklad pro pravidlo č. 2
Chybné schéma klauzí (1, 1, 2, 0, 2, 0)
Správné schéma klauzí (1, 0, 2, 2, 2, 0)

Příklad pro pravidlo č. 3
Chybné schéma klauzí (1, 0, 2, 2, 0, 2, 2, 0)
Segmentační schéma (0, 0, 1, 1, 1, 2, 2, 0)
Správného schéma klauzí (1, 0, 2, 2, 0, 3, 3, 0)

Příklad pro pravidlo č.4
Chybné schéma klauzí (1, 0, 2, 1, 0, 2, 2)
Správné schéma klauzí (1, 0, 2, 1, 0, 3, 3)

Tab. 12: Příklady chyb, které jednotlivá pravidla řeší (červeně jsou označeny chybné klauze).

8.3.2. Použití implicitních pravidel ve statistickém přístupu

V pravděpodobnostním přístupu se využívalo absolutních hodnot, i když segment s připravenou značkou může být v různých větách součástí různých klauzí. V takovém případě je jasné, že problematické pro tento přístup budou delší věty nebo věty s méně obvyklou strukturou. Další přístup je založen na rozdílu označení klauzí pro sousedící segmenty. Jedná se o variantu pravděpodobnostního algoritmu, který ovšem zároveň využívá implicitních pravidel při určování klauzí:

1) První klauze má číslo 1 a každá další kluze je o 1 vyšší.
2) Hranice mezi dvěma klauzemi spadá do nulté klauze.
3) Hranice mezi segmenty stejné klauze spadá do stejné klauze.

8.3.3. Zakomponování úrovně zanoření do předchozího přístupu

Předchozí přístup (v kap. 8.3.2) využívá vztahy mezi segmenty a zároveň pravidla, která jsou obecně platná. Další varianta využívá, kromě obecně platných pravidel i úrovně zanoření a toho, že při každé změně úrovně zanoření musí dojít ke změně klauze.

Vstupní věta je rozložena podle úrovní zanoření na menší celky a jednotlivé celky jsou analyzovány jako samostatná věta. Vztahy jsou popsány pouze dvěma hodnotami, které vyjadřují vztah segmentů na jedné úrovni:

- 0 – zachování aktuální klauze,
- 1 – zvýšení klauze o jedna.

9 Výsledky

Chtěl jsem nalézt metriku, která by odpovídala problémům, které jsem ve své práci řešil a která by dokázala vytvořit ucelený obraz souboru výsledků. Určování hloubky zanoření i čísla klauzí nejsou, i když lze na ně takto pohlížet, problémem čistě klasifikačním. U klasifikačního problému je prvek zařazen do své třídy především pro své vlastnosti. U obou problémů je však také jedním z klíčových prvků vzdálenost a historie. Součástí obou řešení jsou základní varianty, které se ukazují jako velmi účinné (viz kap. 9.2 – 9.4). I když u pravděpodobnostního přístupu zacházím s klauzemi jako při tagování, použít na tento problém metriku F-score mi nepřišlo příliš šťastné. Recall je vždy maximální (neexistuje prvek, kterému by se nepodařilo přidělit třídu). Nakonec jsem jako hlavní ukazatel zvolil klasickou procentuální úspěšnost.

Kromě celkové úspěšnosti jsou vybrány dva speciální typy vět, které by měly reflektovat úspěšnost jednotlivých metod. Prvním typem jsou souvětí (tedy věty, které mají dvě slovesa v různých klauzích) a věty, které obsahují klauzi, která je roztříštěná jinou klauzí (např. vnořené věty).

9.1 Informace o datech

Přístupy byly testovány na dostupných datech, která byla vytvořena na základě ručně anotovaných segmentů a dat z PDT 2.5 (viz kap. 3). Skupina vět, na kterých je možné testovat úspěšnost segmentace, určení hloubky zanoření a označení klauzí, je omezena počtem vět, které byly ručně anotovány z hlediska úrovně zanoření. Vytvořená zlatá data jsem rozdělil na tři balíčky: *Develop, Heldout a Test data*. Dále existuje sada dat pouze pro testování klauzí – *ClauseTest data.*

Data byla rozdělena na balíčky pro vývoj pravidel a statistických algoritmů. *Develop data* jsou data používaná při vývoji. Jejich výpovědní hodnota je minimální pro celkové hodnocení algoritmů, a proto nejsou často v měřeních zahrnuta. *Heldout data* jsou data, která jsem vložil do algoritmu po dokončení jejich vývoje. Zároveň jsou tato data použita při nastavování lamba parametru u statistického přístupu. *Test data* jsou data, na kterých byly prováděné testy v závěrečné fázi. *ClauseTest data*

tvoří balíček vět z PDT 2.5, které byly upraveny tak, aby bylo možné změřit úspěšnost jednotlivých algoritmů, ale pouze z pohledu určení klauzí.

Tabulka 13 ukazuje globální vlastnosti jednotlivých balíčků.

Vlastnost / balíček	Develop data	Heldout data	Test data	ClauseTest data
Počet vět	500	500	1499	4657
Počet segmentů	4026	3954	12610	25417
Počet klauzí	980	967	3005	7534
Maximální úroveň zanoření	4	4	5	--
Maximální počet klauzí	9	7	10	19
Zastoupení souvětí	291 (58,2 %)	288 (57,6 %)	908 (60,57 %)	1787 (38,37 %)
Zastoupení vět s rozštěpenou klauzí	56 (11,2 %)	49 (9,8 %)	138 (9,20 %)	275 (5,90 %)
Baseline pro odhad úrovně zanoření	81.99%	81.20%	81.18%	--
Baseline počtu klauzí	95,6 %	97.00 %	96.13 %	96.56 %
Baseline správně určených klauzí	61,11 %	61.20 %	59.72 %	64.50 %
Baseline správně určených vět z pohledu segmentů	59,65 %	62.00 %	56.03 %	--
Baseline správně určených vět z pohledu klauzí	10.00 %	10.40 %	12.54 %	14.77 %

Tab. 13: Rozložení dat v testovacích sadách

Základní odhady jsou založeny na primárních jevech, které ovlivňují hloubku zanoření a čísla klauzí. Odhad zanoření segmentů je založen na třech pravidlech:

1) hranice má úroveň zanoření 0,
2) segment s příznakem podřízenosti má úroveň zanoření 1,
3) v ostatních případech je úroveň zanoření 0.

První pravidlo je čistě technické. Druhé pravidlo vychází z pozorování, že zanoření není příliš hluboké a hloubka zanoření segmentu s příznakem podřízenosti je nejpravděpodobněji 1. Důvody třetího pravidla jsou stejné jako v předchozím případě. Celková úspěšnost se jeví velmi slibně, přesto musíme brát do úvahy i odhad vět, které mají všechny segmenty určené správně.

Podobně jsem postupoval i při základním odhadu klauzí. Základním prvkem, který určuje číslo klauze, je sloveso. Pravidla jsou následující:

1) hranice spadá do nulové klauze,
2) segment se slovesem zakládá novou klauzi,
3) segment bez slovesa pokračuje v aktuální klauzi.

Z pokusů je jasně vidět, že odhad počtu klauzí není velkým problém. Úspěšnost klesá s požadovanou přesností. Zatímco baseline pro odhad počtu klauzí je velmi vysoký, baseline pro správně určené všechny klauze ve větě klesl o 10 %.

Z tabulky 13 je dále možné si představit, jaká data máme k dispozici. Na sadách dat, které byly vytvořené na základě anotovaných segmentů s úrovní zanoření, je vidět, že se jedná o celkem vyrovnaná data. Obsahují podobné zastoupení vět s pozorovanými jevy (např. souvětí nebo věta s rozdělenou klauzí). U posledního setu je vidět, že pozorované jevy jsou obecně v menším zastoupení. Přesto základní odhady fungují velmi podobně u všech množin vět.

9.2 Výsledky odhadu zanoření segmentu

Z implementace společného základu je vidět, že reálné určení hloubky segmentu v rámci věty je poměrně úspěšné. Během implementace pravidlového systému došlo k drobným úpravám, které způsobily, že celková úspěšnost odhadu je

oproti základním odhadům z pohledu celkového počtu segmentů zvýšena jen o několik procent. Oproti základním odhadům určení celé věty vzrostla úspěšnost poměrně výrazně (viz Tab. 14a). Na rozdíl od jednoduchého pravděpodobnostního přístupu (viz Tab. 14b) je vidět, že pravidlový přístup má celkově větší úspěšnost.

	Heldout data	**Test data**
Celková úspěšnost určení jednotlivých segmentů	88.89 %	88.34 %
Celková úspěšnost určení celých vět	74 %	71.11 %
Úspěšnost určení jednotlivých segmentů v souvětích	86.29 %	84.09 %
Úspěšnost určení celých souvětí	64.58 %	57.26 %
Úspěšnost určení jednotlivých segmentů ve větách, kde je klauze rozdělena jiným segmentem	86.42 %	75.64 %
Úspěšnost určení celých vět, jejichž některá klauze je rozdělena jiným segmentem	57.14 %	34.05 %

Tab. 14a: Výsledky pravidlového systému pro určení hloubky zanoření

	Heldout data	**Test data**
Celková úspěšnost určení jednotlivých segmentů	80.88 %	80.89 %
Celková úspěšnost určení celých vět	61.6 %	55.63 %
Úspěšnost určení jednotlivých segmentů v souvětích	74.95 %	75 %
Úspěšnost určení celých souvětí	46.18 %	39.2 %
Úspěšnost určení jednotlivých segmentů ve větách, kde je klauze rozdělena jiným segmentem	78.84 %	70.57 %
Úspěšnost určení celých vět, jejichž některá klauze je rozdělena jiným segmentem	51.02 %	28.98 %

Tab. 14b: Výsledky statistického systému pro určení hloubky zanoření

9.3 Výsledky spojování segmentů do klauzí – pravidlový přístup

Předchozí kapitola ukázala, jak úspěšné byly jednotlivé přístupy při problému určování hloubky zanoření segmentu. Následující kapitola by měla ukázat, jak ovlivňuje výsledek předchozího úkolu hlavní problém celé práce – tvorbu klauzí.

Prvním způsobem řešení je pravidlový přístup. V tabulce níže (Tab. 15a) je ukázána celková úspěšnost v rámci jednotlivých balíčků. Tabulky 15b a 15c se soustředí pouze na vybrané věty z jednotlivých balíčků.

Všechny věty	**Test data – použíti načtených úrovní zanoření**	**Test data – úroveň zanoření je odhadnuta vlastním systémem**	**Test clause data – obsahují informaci pouze o klauzi**
Podíl správně určených segmentů	91.17 %	89.98 %	90,63 %
Podíl správě určených klauzí	81.26 %	79.41 %	84.75 %
Podíl správně určených celých vět	80.12 %	79.51 %	86.98 %

Tab. 15a: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Souvětí	**Test data – použití načtených úrovní zanoření**	**Test data – úroveň zanoření je odhadnuta vlastním systémem**	**Test clause data – obsahují informaci pouze o klauzi**
Podíl správně určených segmentů	89.26 %	87.77 %	87.51 %
Podíl správě určených klauzí	79.67 %	78.08 %	79.19 %
Podíl správně určených celých vět	75.11 %	75.11 %	76.10 %

Tab. 15b: Sumarizace výsledků pro souvětí

Věty obsahující rozloženou klauzi	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	82.86 %	73.89 %	72.84 %
Podíl správě určených klauzí	64.89 %	49.57 %	49.21 %
Podíl správně určených celých vět	51.14 %	34.78 %	33.45 %

Tab. 15c: Sumarizace výsledků pro věty obsahující rozbitou klauzi

Z tabulek je vidět, že pravidlový přístup je poměrně úspěšný. Na tabulkách uvedených níže uvidíme, že i pravděpodobnostní přístup je úspěšnější než odhadovaný baseline, ale zároveň je mnohem méně úspěšný než přístup pravidlový.

Všechny věty	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	69,92 %	68.36 %	70,85 %
Podíl správě určených klauzí	31.96 %	30.01 %	44.99 %
Podíl správně určených celých vět	26.95 %	26.15 %	50.86 %

Tab. 16a: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Souvětí	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	67.18 %	65.07 %	63.93 %
Podíl správě určených klauzí	31.89 %	29.35 %	27.52 %
Podíl správně určených celých vět	22.13 %	20.92 %	19.3 %

Tab. 16b: Sumarizace výsledků pro souvětí

Věty obsahující rozloženou klauzi	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	69.46 %	64.74 %	62.87 %
Podíl správě určených klauzí	29.23 %	20.86 %	21.75 %
Podíl správně určených celých vět	13.03 %	10.86 %	12 %

Tab. 16c: Sumarizace výsledků pro věty obsahující rozbitou klauzi

Výsledky pravděpodobnostního přístupu nejsou příliš úspěšné a rozšíření tagu o další tři pozice vedlo v rámci testovaných dat k horším výsledkům (viz Tab. 17).

Všechny věty	Test data – použíti načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	68.51 %	66.88 %	69.54 %
Podíl správě určených klauzí	26.58 %	24.87 %	40.79 %
Podíl správně určených celých vět	21.74 %	20.54 %	46.81 %

Tab. 17: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

9.4 Výsledky spojování segmentů do klauzí – kombinace přístupů

Celkově jsem vytvořil dva základní přístupy spojení pravděpodobnostního a pravidlového přístupu. První pokus je pouze oprava výstupu z pravděpodobnostního přístupu pomocí jednoduchých vypozorovaných pravidel (viz kap. 8.2). Druhý přístup založený na rozdílu klauzí mezi sousedícími segmenty a implicitních pravidel. Třetí přístup je rozšířením předchozího přístupu a využívá kromě implicitních pravidel i úroveň zanoření. Na oba kombinované přístupy jsem také použil opravná pravidla pro opravu statistického přístupu.

9.4.1. Oprava statistického přístupu

Jako základ tohoto přístupu je použit pravděpodobnostní přístup, který je vylepšen jednoduchými pravidly, která jsou popsána v kapitole 8.3.1.

Všechny věty	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	77,34 %	73.75 %	73.92 %
Podíl správě určených klauzí	53.49 %	48.53 %	55.62 %
Podíl správně určených celých vět	56.90 %	53.30 %	65.19 %

Tab. 18a: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Souvětí	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	72.22 %	66.99 %	64.18 %
Podíl správě určených klauzí	47.35 %	40.67 %	35.28 %
Podíl správně určených celých vět	43.17 %	37.22 %	32.45 %

Tab. 18b: Sumarizace výsledků pro souvětí

Věty obsahující rozloženou klauzi	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	80.12 %	64.91 %	65.05 %
Podíl správě určených klauzí	52.19 %	33.27 %	30.99 %
Podíl správně určených celých vět	44.92 %	27.53 %	25.09 %

Tab. 18c: Sumarizace výsledků pro věty obsahující rozbitou klauzi

9.4.2. Zakomponování implicitních pravidel do statistického přístupu

Na první pohled statistický přístup trpí problémem použití absolutních hodnot v číslech klauzí, tzn. segmenty se stejnými značkami mohou být součástí různých klauzí. Označení klauze tedy není nijak spojeno se segmenty, které obsahuje. Další postup se soustředil na vztahy sousedících segmentů a místo snahy o přesné označení se snaží odhadnout vztah mezi segmenty z pohledu klauzí a aplikováním těchto vztahů odhadnout klauze.

Všechny věty	**Test data – použíti načtených úrovní zanoření**	**Test data – úroveň zanoření je odhadnuta vlastním systémem**	**Test clause data – obsahují informaci pouze o klauzi**
Podíl správně určených segmentů	77.77%	76.60 %	76.76 %
Podíl správě určených klauzí	54.13 %	52.77 %	60.66 %
Podíl správně určených celých vět	60.04 %	58.97 %	70.88 %

Tab. 19: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Metoda zakomponování implicitních pravidel preferuje pravidla lokálního charakteru, předmětem pravidel jsou sousedící segmenty. Následující tabulky ukazují, do jaké míry dojde ke zlepšení při použití opravných pravidel použitých v kapitole 8.3.1.

Všechny věty	Test data – použíti načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data obsahují informaci pouze o klauzi
Podíl správně určených segmentů	79.57 %	76.48 %	76.57 %
Podíl správě určených klauzí	58.84%	53.93 %	60.55 %
Podíl správně určených celých vět	60.50 %	57.17 %	69.10 %

Tab. 20: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Metoda dává pro všechny typy vět dokonce lepší výsledky než pravděpodobnostní přístup s opravnými pravidly. Ve větách s rozdělenou klauzí však dochází k prudkému poklesu úspěšnosti a ani spuštění opravných pravidel úspěšnost příliš nezvýší. Procházením výsledků se ukázalo, že chyby vznikaly díky nerozpoznání přechodu z nižší úrovně do vyšší (vynořování), ve kterém by mělo dojít ke změně čísla klauze. Důvodem může být častý vzor zvýšení úrovně zanoření, ale zároveň velmi nízký výskyt snížení úrovně zanoření a s tím spojenou změnu klauze.

9.4.3. Zakomponování pravidel a úrovně zanoření segmentů do statistického přístupu[3]

Předchozí přístup má poměrně dobré výsledky v celkových odhadech. Ovšem velmi neobstál na větách, které obsahují klauzi rozdělenou jinou klauzí. Procházením nesprávně určených vět se ukázalo, že tento přístup se nedokáže vyrovnat s výstupem o úroveň zanoření výše, i když pravidlově je tento problém velmi dobře zvládnutelný. Dalším krokem jsem se snažil přidat pravidlo rozložení segmentů do úrovní a zkoumat vztahy segmentů pouze na jednotlivých úrovních. Výsledky jsou ukázány v tabulkách 21a – 21c.

[3]Rozšíření kapitoly 9.4.2.

Všechny věty	Test data – použíti načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	79.18 %	74.21 %	74.85 %
Podíl správě určených klauzí	56.83 %	49 %	58.15 %
Podíl správně určených celých vět	60.44 %	54.1 %	68.26 %

Tab. 21a: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Souvětí	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	74.19 %	67.32 %	64.66 %
Podíl správě určených klauzí	50.74 %	41.04%	34.36 %
Podíl správně určených celých vět	47.79 %	38.65 %	32.68 %

Tab. 21b: Sumarizace výsledků pro souvětí v jednotlivých balíčcích

Věty s rozdělenou klauzí	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	76.57%	63.4 %	64.13 %
Podíl správě určených klauzí	47.04 %	28.06 %	27.01 %
Podíl správně určených celých vět	39.85 %	23.91 %	19.63 %

Tab. 21c: Sumarizace výsledků pro všechny věty obsahující rozdělenou klauzi

9.4.4. Přidání opravných pravidel do přístupu předchozí kapitoly

Všechny věty	Test data– použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	79.50 %	74.13 %	74.79 %
Podíl správě určených klauzí	57.72 %	49.06 %	58.19 %
Podíl správně určených celých vět	60.64 %	53.50 %	68.17 %

Tab. 22a: Sumarizace výsledků pro všechny věty v jednotlivých balíčcích

Souvětí	Test data– použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	74.75 %	67.30 %	64.59 %
Podíl správě určených klauzí	52.17 %	41.36 %	35.84 %
Podíl správně určených celých vět	48.23 %	37.77 %	32.62 %

Tab. 22b: Sumarizace výsledků pro souvětí v jednotlivých balíčcích

Věty s rozdělenou klauzí	Test data – použití načtených úrovní zanoření	Test data – úroveň zanoření je odhadnuta vlastním systémem	Test clause data – obsahují informaci pouze o klauzi
Podíl správně určených segmentů	81.11 %	64.51 %	65.62 %
Podíl správě určených klauzí	57.01 %	32.45 %	31.53 %
Podíl správně určených celých vět	49.27 %	26.08 %	24 %

Tab. 22c: Sumarizace výsledků pro všechny věty obsahující rozdělenou klauzi

9.5 Analýza výsledků

Z tabulek v kapitolách 9.3 a 9.4 je vidět hned několik základních pozorování. Jak jsem očekával, úspěšnost je vyšší pro balíčky s předem analyzovanou úrovní zanoření. I když rozdíly jsou jen v řádu procent. Dalším očekávaným výsledkem je větší úspěšnost pravidlového systému. Ovšem stávající pravidlový přístup není evidentně ideálním postupem a úspěšnosti, které se pohybují velmi blízko baseline, se nedají považovat za úspěch.

Během vývoje se objevil překvapivý vztah mezi procentuální úspěšností určení segmentů a určení celých vět. Dokud se úspěšnost analýzy segmentů (příslušnost segmentu ke klauzi) pohybovala kolem 70 %, byla úspěšnost v analýze vět cca 40 %. V okamžiku, kdy úspěšnost analýzy segmentu přesáhla 80 %, úspěšnost analýzy vět se zvedla o více jak 20 %. Rozdíly mezi pravidlovým a statistickým přístupem toto pozorování potvrzují.

Během vývoje se také ukázalo, že snaha zachytit některá lingvistická pravidla vede k velmi malým posunům (v rámci setin procent), nebo dokonce úspěšnost snižuje, na rozdíl od pozorovaných jevů, jejichž zachycení zvyšovalo úspěšnost v rámci procent. Zároveň se rozšíření pravděpodobnostního přístupu o lingvisticky motivované nové informace ukázalo jako krok zpět a docházelo k více chybám, než se objevilo pří základním tagu segmentu.

Sloučení pravděpodobnostního a pravidlového přístupu zvedlo, podle očekávání úspěšnost oproti pravděpodobnostnímu přístupu, ale proti pravidlovému přístupu jsou měřené úspěšnosti cca o 10 % nižší.

Také se ukázalo, že největším problémem při určování klauzí nejsou dlouhé věty, ale věty, které obsahují klauzi rozdělenou pomocí jiné klauze. V extrémním případě jsem dokonce nedosáhl ani na základní jednoduchý odhad, tzn. 10 %. I přesto, že úspěšnost určování příslušnosti segmentu do klauzí v těchto typech vět byla více než 50 % a celkové výsledky přesahují pravděpodobnostní přístup s opravnými pravidly, celkové opravení výsledků nepřineslo větší úspěch. Největší úspěch určení pro věty, které obsahují klauzi rozdělenou jinou klauzí, byl při použití pravidlového přístupu. Při jeho použití byla dosažena úspěšnost okolo 33 % (viz kap. 9.3).

Celková nízká úspěšnost statistického nebo kombinovaného přístupu může být výsledkem mnoha faktorů. Máme k dispozici poměrně malý vzorek vět, na kterém je možné se učit. Problémem může být i to, že většina informace o vazbě mezi segmenty jedné klauze v těchto typech vět je spíše na analytické rovině. Problémem může být samotný algoritmus, který zpracovává pouze nejbližší okolí zkoumaného segmentu, navíc pouze historii.

10 Závěr

V této práci byly představeny pojmy segment a klauze. Jedná se o lingvisticky motivované pojmy. Zároveň byly popsány vzájemné vztahy mezi těmito objekty. Kvantitativní analýzou, pozorováním a částečně také lingvistickou intuicí byl vytvořen návrh pravidel pro analýzu vět obohacené o morfologické informace.

Součástí práce je také návrh pozičního tagu pro segmenty. Tento tag podává kompletní informaci o segmentu. Zároveň slouží jako zástupný objekt v analýzách a pravidlech. Původně zamýšlený formalismus se ukázal jako příliš jednoduchý pro daný problém a složitější formalismus by mohl být tématem samostatné diplomové práce.

Podle dosažených výsledků se zdá být problém spojování segmentů do klauzí spíše pravidlovým problémem. Základní statistické metody HMM nejsou pro tento problém příliš vhodným řešením. Důvodem selhání HMM je podle mne převaha jednoho či více vzorů, které poté ovlivní celý výsledek (např. první segment, kterému nepředchází hranice, má vždy úroveň zanoření 0), a zároveň poměrně málo dat, ze kterých by bylo možné se učit. Další možnou příčinou je nedostatek snadno pozorovatelných vlastností segmentu, které by ovlivňovaly příslušnost segmentu ke klauzím. Pokus o rozšíření lingvisticky motivovaných vlastností do značky segmentu snížil celkovou úspěšnost algoritmu.

Ovšem porovnáním pravděpodobnostního přístupu s explicitně vyjádřeným číslem klauze a přístupem, který využíval rozdílu v číslech klauzí sousedících segmentů, nám vyplyne, že se ve větách musí opakovat určité pevné schéma.

Podle tabulek 13a – 13c se ukázalo, že i velmi jednoduchá pravidla zapsaná v kódu dokáží zajistit poměrně vysokou úspěšnost. Tato úspěšnost je porovnatelná s úspěšností metody, která přiřazuje čísla klauzí podstromům v PDT. Rozdíl mezi touto prací a metodou na PDT je, že můj algoritmus se musí vyrovnat s tím, že hranice klauzí a hranice segmentů nejsou rozpoznatelné v povrchové struktuře věty, protože tyto objekty jsou vyjádřené pomocí stejné grafické reprezentace. Přesto jsou výsledky pravidlového přístupu velmi zajímavé.

Dalšími kroky, které by mohly pomoci v aplikaci segmentů, jsou již zmíněný

formalismus a s jeho rozvojem i detailnější pravidla, vylepšování pravidlového přístupu pomocí sofistikovanějších statistických metod, které by kromě lokálních vztahů zohledňovaly také globální vlastnosti. Zároveň by se navrhovaný algoritmus měl být schopen naučit na poměrně malém vzorku dat. Data pro evaluaci klauzí jsou sice dostupná v podobně korpusu PDT, problémem je však korpus obsahující zároveň úroveň zanoření. Tato informace se ukázala jako velmi důležitá. Na základě mých pokusů odhaduji, že jisté mírné zlepšení může přinést i vylepšení algoritmu segmentace a odhadu úrovně zanoření.

V celkovém pohledu doufám, že by tato práce mohla přispět i k praktickému využití při zlepšení syntaktické analýzy nebo strojového překladu. Podle teoretických předpokladů by analýza samostatných klauzí měla být rychlejší než analýza celého složitého souvětí.

11 Přehled odborné literatury

[1] Kuboň Vladislav, Lopatková Markéta: Od segmentů ke klauzím v češtině - analýza vybraných jevů. In: *Informačné technológie – Aplikácie a Teória, Zborník príspevkov prezentovaných na konferencii ITAT*, Copyright © PONT s. r. o., Seňa, Slovakia, ISBN 978-80-970179-3-4, pp. 76-80, 2010

[2] Lopatková Markéta, Holan Tomáš: Vztahy mezi segmenty – segmentační schémata českých vět. In: *Informačné Technológie – Aplikácie a Teória. Zborník príspevkov, ITAT 2008*, Copyright © PONT s.r.o., Seňa, Slovakia, ISBN 978-80-969184-8-5, pp. 15-22, 2008

[3] ÚSTAV PRO JAZYK ČESKÝ, Akademie věd ČR. *Internetová jazyková příručka* [online]. 2008, 2012 [cit. 2013-03-30]. Dostupné z: http://prirucka.ujc.cas.cz/?id=150

[4] Kuboň Vladislav, Lopatková Markéta, Plátek Martin, Pognan Patrice: A Linguistically-Based Segmentation of Complex Sentences. In: *Proceedings of FLAIRS 2007 (20th International Florida Artificial Intelligence Research Society Conference)*, Copyright © AAAI Press, Key West, FL, USA, ISBN 978-1-57735-319-5, pp. 368-373, 2007

[5] Kuboň Vladislav: Problems of Robust Parsing of Czech. In: *disertační práce MFF UK, disertační práce MFF UK*, 2001

[6] Lopatková Markéta, Homola Petr, Klyueva Natalia: *Annotation of sentence structure: Capturing the relationship between clauses in Czech sentences.* In: Language Resources and Evaluation, Vol. 46, No. 1, Copyright © Springer Netherlands, ISSN 1574-020X, pp. 25-36, Mar 2012

[7] Krůza Oldřich, Kuboň Vladislav: Automatic Extraction of Clause Relationships from a Treebank. In: Lecture Notes in Computer Science, Vol. 5449, No. 5449/2009, *Computational Linguistics and Intelligent Text Processing. 10th International Conference, CICLing 2009, Mexico City, Mexico, March 1-7, 2009, Proceedings*, Copyright © Springer, Berlin / Heidelberg, ISBN 978-3-642-00381-3, ISSN 0302-9743, pp. 195-206, 2009

[8] Zeman Daniel: Parsing with a Statistical Dependency Model, In: *disertační práce MFF UK, disertační práce MFF UK*, 2004

[9] Dutkevič Jiří: An Implementation of Methods of Structural Analysis of Czech Complex Sentences, In: *bakalářská práce MFF UK, bakalářská práce MFF UK*, 2004

[10] Pražský závislostní korpus 2.5 [online dokumentace], ÚFAL MFF UK 2011, dostupné z http://ufal.mff.cuni.cz/pdt2.5/

[11] Valency Lexicon of Czech Verbs VALLEX [online dokumentace], ÚFAL MFF UK 2011, dostupné z http://ufal.mff.cuni.cz/vallex/