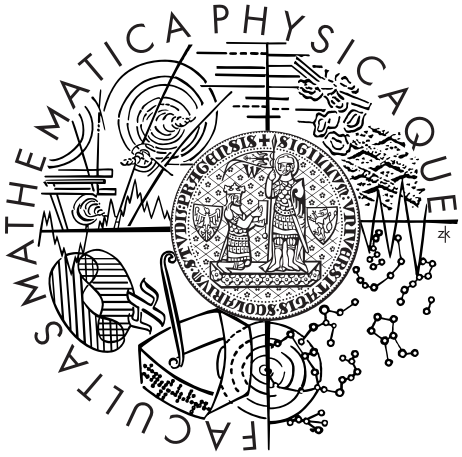
Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Marek Tlustý

Jazykové modelování pro němčinu

Ústav formální a aplikované lingvistiky

Vedoucí bakalářské práce: RNDr. Ondřej Bojar, Ph.D.
Studijní program: Informatika
Studijní obor: Obecná informatika

Praha 2013

Rád bych poděkoval především vedoucímu práce
RNDr. Ondřeji Bojarovi, Ph.D.
za cenné rady a připomínky při vedení práce.

Dále bych rád poděkoval
RNDr. Danielu Zemanovi, Ph.D.
za poskytnutí německých nbestlistů
Bc. Rudolfu Rosovi
za identifikaci anglických klauzí.

Velký dík patří i mé rodině, přátelům a známým za jejich podporu při studiu.

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) 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 24. 5. 2013 Podpis autora

Název práce: Jazykové modelování pro němčinu

Autor: Marek Tlustý

Katedra: Ústav formální a aplikované lingvistiky

Vedoucí bakalářské práce: RNDr. Ondřej Bojar, Ph.D.

Abstrakt: Práce se zabývá jazykovým modelováním pro němčinu. Soustředí se na specifika německé gramatiky, která činí běžným n-gramovým modelům problémy. Nejprve popisuje statistické metody jazykového modelování a vysvětluje problematické jevy němčiny. Následně navrhuje vlastní varianty n-gramových jazykových modelů s cílem tyto problémy zlepšit. Vlastní modely jsou trénovány jednak jako standardní n-gramové, a jednak také metodou maximální entropie s n-gramovými rysy. Oba typy jsou vždy porovnány z hlediska korelace ručně hodnocené plynulosti vět a automatického hodnocení – perplexity. Srovnány jsou zároveň výpočetní nároky potřebné k natrénování jednotlivých modelů. Dále je navrhnuta množina vlastních rysů reprezentující počet gramatických chyb vybraných jevů. Úspěšnost se ověřuje na schopnosti predikovat ručně hodnocenou plynulost. Využito je modelů maximální entropie a vlastních modelů klasifikujících jen na základě mediánů hodnot rysů vypočtených z trénovacích dat. U rysů, jejichž hodnota dobře koreluje s plynulostí, zkoušíme taktéž predikci pomocí lineární regrese. Výsledky n-gramových modelů ukazují určité zlepšení. S vlastními rysy se dařilo predikovat plynulost alespoň přibližně.

Title: Language Modelling for German
Klíčová slova: jazykové modelování, němčina, n-gram, maximální entropie

Author: Marek Tlustý

Department: Institute of Formal and Applied Linguistics

Supervisor: RNDr. Ondřej Bojar, Ph.D.

Abstract:

Keywords: language modelling, German, n-gram, maximum entropy

Obsah

Úvod

Lidé si už od pradávna chtěli ulehčit překlady mezi různými jazyky. Prvopočátky se objevují už v 17. století, kdy se německý učenec Joachim Becher snažil usnadnit překlad mezi více jazyky tím, že slovům přidělil logický kód. Věty převedené na takový logický kód pak představovaly univerzální mezijazyk a do dalších jazyků se překládaly za pomoci speciálních slovníků. Mechanizace překladu ale přichází až později ve 20. století. Ve Francii ve 30. letech sestrojili první mechanický slovník pracující s děrovanými páskami. Teprve až s příchodem počítačů začíná éra strojového překladu. Po druhé světové válce se na zprovoznění funkčního systému intenzivně pracuje a předpokládá se, že pro počítače nemůže být překlad nic složitého. Při realizaci se ale naráží na spoustu problémů, jež se mimojiné nepodařilo úplně vyřešit dodnes, neboť dodnes nemáme překladový systém, který by byl schopen nahradit lidského překladatele. První systémy pracovaly jen s několika gramatickými pravidly a měly malý slovník. Kvalita překladu tak byla velmi špatná.

Postupem času se začaly více uplatňovat statistické metody zpracování přirozeného jazyka. S příchodem myšlenky zašumněného kanálu [1], byly na světě konečně i jazykové modely, kterými se tato práce bude zabývat. Princip zašumněného kanálu spočíval v myšlence nahradit překladový model z výchozího do cílového jazyka modelem obráceným tj. z jazyka cílového do jazyka výchozího s využitím právě jazykového modelu. Ten má za úkol z více navrhovaných hypotéz vybrat tu, která vypadá jako nejhezčí věta.

Mimo strojový překlad se rozvíjely i další aplikace zpracování přirozeného jazyka jako např. rozpoznávání mluvené řeči nebo automatická oprava překlepů v psaném textu. I v těchto oblastech našly jazykové modely své uplatnění a jejich užití je dnes poměrně široké.

Jedny z nejrozšířenějších a stále nejpoužívanějších modelů jsou modely *n-gramové*, které sledují jen krátké posloupnosti po sobě jdoucích slov a na základě několika slov se snaží předpovědět slovo následující. Takový přístup úspěšně funguje na analytické jazyky, které pro gramatické jevy nepoužívají ohýbání slov – flexi. Mezi takové jazyky se řadí např. angličtina, ta využívá flexi jen minimálně z historických důvodů. Flektivní jazyky, mezi které patří právě němčina nebo čeština, využívají časování, skloňování, předpony, přípony a další, což podstatně zvyšuje počet přípustných slovních tvarů, a tedy i počet platných n-gramů, které by model potřeboval v trénovacích datech vidět. Němčina navíc velice často dává nějaké slovo na konec věty, což krátká posloupnost slov nemůže zachytit.

V této práci se proto budeme zabývat jazykovými modely pro němčinu. Vyzkoušíme jednak využít morfologickou analýzu pro natrénování *n-gramových modelů* s morfologickými značkami. Modely založené na morfologických třídách zkoušely třeba Chaoui, Yvon, Mokbel, Chollet (ZDROJ) na arabštině. Využili kombinace se standardními n-gramovými modely a uvádějí zlepšení. Kladně hodnotí využití rozšířeného slovního druhu i Wakita, Kawai, Iida (ZDROJ). Na němčině zkoušel tento přístup třeba Geutner (ZDROJ). Za pomocí interpolace kombinuje modely se slovy a modely založené na slovních druzích. Trochu jiný způsob popisuje

Popović (ZDROJ), která kombinuje slovní druhy a morfémy. S morfémy experimentovali i Fraser, Marion, Weller, Cap [3]. Ti nejprve překládali kmen slova a poté predikovali jeho formu za pomocí modelování pádu, rodu a čísla. S úspěšností predikce formy se dostali až téměř k 95 %. My zkusíme využít jen modelů na morfologických značkách, u nichž budeme zkoumat korelaci automatického a ručního hodnocení s cílem korelaci zlepšit, aby hodnocení počítačem (perplexita) korelovalo s hodnocením lidmi (plynulost vět).

Ručním hodnocením plynulosti se budeme zabývat i z jiné stránky a zkusíme navrhnout vlastní množinu rysů pro využití potenciálu modelů maximální entropie s cílem postihnout německou gramatiku. Gramatické rysy zkoušel využít např. Rukolaien (ZDROJ). Udává zlepšení perplexity o 16 % a zkouší zároveň i kombinaci s n-gramovými modely se slovy. Amaya, Benedí (ZDROJ) využívají také gramatické rysy modelované za použití slovních druhů stochastickými bezkontextovými gramatikami. Naše modely budou sice také vycházet z gramatiky, ale nebudou modelovat správné gramatické jevy, nýbrž gramatické chyby. Na základě nich se pak budeme snažit predikovat ručně hodnocenou plynulost.

Hlavní obsah práce je strukturován do čtyř kapitol. První z nich se zabývá jazykovými modely. Popisuje matematické základy a statistické metody, které se pro jazykové modelování používají. V další kapitole jsou pak stručně vysvětleny problematické jevy německé gramatiky s ukázkou analýzy chybných hypotéz. Následuje první kapitola s experimenty popisující princip a výsledky experimentů týkajících se jazykových modelů s morfologickými značkami. Takové modely trénujeme jednak jako standardní n-gramové, a jednak metodou maximální entropie s n-gramovými rysy. Graficky srovnáváme korelaci perplexity a ručně hodnocené plynulosti u obou typů. Celkové srovnání je pak provedeno za pomocí korelačních koeficientů. Poslední kapitola se pak zabývá druhou sérií experimentů, kde je zkoumána korelace navrhnutých rysů s ručně hodnocenou plynulostí. Jsou zde natrénovány modely maximální entropie a také vlastní modely hodnotící jen na základě mediánů vypočtených dle jednotlivých plynulostí z trénovacích dat. Srovnání provádíme podle úspěšnosti predikce plynulosti.

1. Jazykové modely

Jazykový model se snaží charakterizovat zákonitosti v přirozeném jazyce. K tomu je možné přistupovat za pomoci více či méně podrobné statistiky. Zákonitosti můžeme popisovat pravidly nebo je zkusit automaticky vypozorovat z velkého množství textů – tzv. statistický přístup. Proces určování parametrů ve statistickém přístupu se nazývá *trénování modelu*. Lingvistické znalosti pak můžeme do modelů přidat například tak, že model nenecháme trénovat jenom na samotném textu, ale i na morfologických nebo jiných značkách či gramatických vztazích. Právě takovými modely se budeme zabývat.

1.1 N-gramové modely

N-gramové jazykové modely nepotřebují téměř žádné lingvistické zpracování. Využívají skutečnosti, že některá slova se často vyskytují v určitých dvojících (obecně n-ticích) – pro němčinu typicky třeba člen a podstatné jméno. Jistě častěji spatříme v trénovacích datech *der Hund* než *das Hund*. Stejně jako po slovese *fragen* uvidíme předložku *nach* nebo *um* spíše než *auf* nebo *an*.

Pro danou posloupnost slov $w_1, \ldots, w_m$ bychom rádi věděli, s jakou pravděpodobností je to správná německá věta. Tuto pravděpodobnost vypočítáme tak, že spočítáme výskyty všech těchto posloupností v datech a normalizujeme je velikostí dat. Trénovací data jsou ale obvykle řídká[1], a proto budeme chtít pozorované vlastnosti zobecnit.

Z Bayesovy věty víme, že

$$P(A|B) = \frac{P(A,B)}{P(B)} \tag{1.1}$$

odtud vyjádříme $P(A,B)$ a dostaneme

$$P(A,B) = P(A|B) \cdot P(B) \tag{1.2}$$

nyní aplikujeme tento vztah na $P(w_1, \ldots, w_m)$ m-krát

$$P(w_1, \ldots, w_m) = P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_1, w_2) \cdot \ldots \cdot P(w_m|w_1, \ldots, w_{m-1}) \tag{1.3}$$

Tento postup se nazývá **pravidlo zřetězení** a díky němu můžeme pravděpodobnost $P(w_1, \ldots, w_m)$ modelovat postupně člen po členu (např. slovo po slově).

Model můžeme dále zjednodušit tím, že přistoupíme na tzv. **Markovův předpoklad**. Ten říká, že každý člen posloupnosti $w_1, \ldots, w_m$ závisí jen na k předchozích. Potom tedy:

$$P(w_m|w_1 \ldots w_{m-1}) \simeq P(w_m|w_{m-k}, \ldots, w_{m-1}) \tag{1.4}$$

Toto tvrzení vede k zavedení pojmu n-gram a Markovovské chování vět v přirozeném jazyce je předpokladem pro fungování n-gramových modelů.

[1] Řídkostí dat rozumíme nízký počet různých kombinací slov, které můžeme pozorovat v trénovacích datech, vzhledem k celkovému počtu možných správných vět. Těch je totiž nesrovnatelně více.

N-gram je n po sobě jdoucích členů $w_1, \ldots, w_n$ z dané posloupnosti $w_1, \ldots, w_m$ (např. n po sobě jdoucích slov ve větě). Pro $n = 1, 2, 3$ používáme označení *unigram*, *bigram* a *trigram*.

Pravděpodobnost $P(w_m|w_{m-k}, \ldots, w_{m-1})$ z (1.4) přesně určit nelze, a proto se používá odhad maximální věrohodnosti (**MLE**):

$$P_{MLE}(w_m|w_{m-k}, \ldots, w_{m-1}) = \frac{count(w_{m-k}, \ldots w_m)}{\sum_l count(w_{m-k}, \ldots, w_{m-1}, w_l)} = \tag{1.5}$$

$$= \frac{count(w_{m-k}, \ldots w_m)}{count(w_{m-k}, \ldots, w_{m-1})}$$

Takto se rozdělí pravděpodobnost mezi všechny spatřené n-gramy v trénovacích datech a právě toto rozdělení pravděpodobnosti tvoří **n-gramový model**.

Problémem však stále zůstává skutečnost, že pro neviděné n-gramy v testovacích datech dostaneme nulovou pravděpodobnost celé věty.

1.2 Good-Turingovo vyhlazování

Good-Turingovo vyhlazování se snaží vyhradit část rozdělení pravděpodobnosti od frekventovanějších n-gramů pro ty méně frekventované a neviděné. Používá k tomu frekvenci frekvencí n-gramů N_r, které se v trénovacích datech vyskytly r-krát. Tedy například pro $r = 3$ je N_3 rovno počtu n-gramů vyskytujících se v trénovacích datech právě třikrát.

Zajímavějším příkladem je ale N_0 tj. počet neviděných n-gramů. Ty nemůžeme spočítat přímo, ovšem výpočet také není nijak složitý. Stačí vzít počet všech možných n-gramů a odečíst počet n-gramů viděných. Pokud uvažujeme model slov, pak pro $n = 3$, velikost slovníku 100 a počet viděných 3-gramů 350 000 je $N_0 = 100^3 - 350\,000 = 650\,000$.

Good-Turingova metoda bere n-gramy, které se vyskytly v trénovacích datech r-krát, jakoby se vyskytly r*-krát:

$$r^* = (r + 1) \cdot \frac{N_{r+1}}{N_r} \tag{1.6}$$

V jednodušší variantě se pro vhodně zvolenou konstantu k pravděpodobnost n-gramu vypočítá jako:

$$P_{GT}(w_1, \ldots, w_n) = \begin{cases} \dfrac{r^*}{\sum_r r \cdot N_r} & \text{je-li } r < k \\ \text{MLE} & \text{jinak} \end{cases} \tag{1.7}$$

Pokud bychom počítali pravděpodobnost pro všechny n-gramy podle prvního vzorce, nejen pro $r < k$, dostaly by ty nejvíce spatřené nulovou pravděpodobnost, neboť pro ně bude $N_{r+1} = 0$. Z tohoto důvodu je potřeba vhodně volit konstantu k a pro $r >= k$ počítat pravděpodobnost standardně odhadem maximální věrohodnosti (MLE), který dává dobré výsledky.

Důkaz, že takto přerozdělíme jenom zbývající část pravděpodobnosti, se lze dočíst v původním dokumentu – Good (ZDROJ).

Good-Turingovo vyhlazování podává dobré výsledky pro málo frekventované n-gramy, a proto se v praxi často používá. Je také výchozím nastavením SRILM toolkitu[2] při trénování n-gramových modelů. Podrobněji o Good-Turingově vyhlazování píše třeba .....[x] nebo ....[y].

1.3 Katzovy back-off n-gramové modely

V trénovacích datech se nemusel objevit n-gram, který zrovna chceme, a bez použití vyhlazování bychom dostali nulovou pravděpodobnost. V trénovacích datech ale mohl být podobný n-gram lišící se jen délkou historie. Pro zužitkování této informace od kratších n-gramů se proto využívá kombinace n-gramových modelů nižších řádů pomocí **lineární interpolace**.

K lineární interpolaci potřebujeme vektor vah λ, pro který platí:

$$\forall i : 0 \leq \lambda_i \leq 1 \quad \text{a} \quad \sum_i \lambda_i = 1 \tag{1.8}$$

Výsledná pravděpodobnost pro trigramový model pak vypadá takto:

$$P(w_3|w_1, w_2) = \lambda_3 P(w_3|w_1, w_2) + \lambda_2 P(w_3|w_2) + \lambda_1 P(w_3) \tag{1.9}$$

Vektor vah lze určit např. pomocí *EM algoritmu* (viz ZDROJ).

Na podobné myšlence kombinace n-gramových modelů s různou délkou historie jsou právě založeny **back-off n-gramové modely**. Ty ovšem neurčují pravděpodobnost vždy podle všech n-gramových modelů nižších řádů, ale využívají nižší řády pouze, pokud ty vyšší neposkytují dostatečnou informaci. Začíná se u modelů s nejvyšším řádem, pokud tento n-gram nebyl spatřen, proběhne tzv. **back-off** k nižšímu řádu a n-gramu se zkrátí historie o poslední člen (např. slovo). Pokud ani tento nižší řád n-gram se zkrácenou historií nikdy neviděl, pokračuje se v back-off operacích, dokud takový řád není nalezen.

Stejně jako se v případě lineární interpolace pravděpodobnosti jednotlivých modelů musely přenásobit vahami λ, aby se stále jednalo o validní rozdělení pravděpodobnosti, musíme najít takový způsob i u této metody. Zde musíme určit složitější normalizační faktor, neboť modelů nižších řádů nebudeme využívat vždy.

Katzovy back-off modely proto odhadují pravděpodobnost n-gramu následovně:

$$P_{BO}(w_n|w_1, \ldots, w_{n-1}) = \begin{cases} d_{w_1,\ldots,w_n} \cdot P_{MLE}(w_1, \ldots, w_n) & \text{pro } count(w_1, \ldots, w_n) > k \\ \alpha_{w_1,\ldots,w_{n-1}} \cdot P_{BO}(w_n|w_2, \ldots, w_{n-1}) & \text{jinak} \end{cases} \tag{1.10}$$

kde

[2]Sada nástrojů pro jazykové modelování. V tomto toolkitu budeme také trénovat všechny n-gramové modely. Více viz (odkaz na zdroj)

- P_{MLE} označuje odhad maximální věrohodnosti zavedený ve vzorci (1.5)
- k je nejméně důležitý parametr a často je voleno $k = 0$
- d je snižující parametr, který zajišťuje vyhrazení určité části pravděpodobnosti pro odhady pravděpodobností s použitím back-off operací
- α je normalizační faktor přerozdělující zbývající část pravděpodobnosti

Parametr d je možné stanovit na základě popsaného Good-Turingova vyhlazování následovně:

$$d_{w_1,\ldots,w_n} = \frac{count(w_1,\ldots,w_n)^*}{count(w_1,\ldots,w_n)} \tag{1.11}$$

přičemž $count(w_1,\ldots,w_n)^*$ se spočítá dle vzorce (1.6) z Good-Turingova vyhlazování.

Výpočet normalizačního faktoru α je o něco složitější. Nejprve zavedeme β jako doplněk pravděpodobnosti součtu všech n-gramů s počtem výskytu (count) vyšším než k. β tak bude představovat zbývající vyhrazenou část pravděpodobnosti pro (n-1)-gramy.

$$\beta_{w_1,\cdots,w_{n-1}} = 1 - \sum_{\{\text{n-gram}|count(\text{n-gram})>k\}} d_{w_1,\cdots,w_n} P_{MLE}(w_1,\ldots,w_n) \tag{1.12}$$

Potom se normalizační faktor α vypočítá jako podíl zbývající pravděpodobnosti β a součtu pravděpodobností n-gramů vyskytujících se nejvýše k-krát. Tím se zajistí vždy ještě dostatek pravděpodobnosti pro další přechod k n-gramům nižších řádů back-off operacemi.

$$\alpha_{w_1,\cdots,w_{n-1}} = \frac{\beta_{w_1,\cdots,w_{n-1}}}{\sum_{\{\text{n-gram}|count(\text{n-gram})\leq k\}} P_{BO}(w_n|w_2\cdots w_{n-1})} \tag{1.13}$$

Back-off n-gramové modely podávají dobré výsledky, a proto jsou v praxi často využívány. Tento typ modelů je ve spojení s Good-Turingovým vyhlazováním výchozím nastavením nástroje `ngram-count` pro trénování modelu z již zmíněného SRILM toolkitu a právě takové modely budeme v této práci vyrábět.

1.4 Vyhlazování Kneser-Ney

Vyhlazování Kneser-Ney se snaží nahradit unigramovou pravděpodobnost, která závisí pouze na frekvenci výskytu slova v trénovacím korpusu, chytřejší pravděpodobností, která bude zohledňovat, v kolika různých kontextech se toto slovo vyskytuje. Tato metoda předpokládá, že slovo vyskytující se ve více kontextech je pravděpodobnější i pro výskyt v kontextu novém.

Pro příklad se často uvádí věta se San Franciscem a brýlemi: Pro příklad uvedeme větu se San Franciscem a psacím strojem:

- Mějme část věty: *V muzeu se mi líbil starý psací ...*

- Naším úkolem je uhádnout slovo, které bude následovat.
- Předpokládejme, že unigramový model by nabídnul slovo Francisco. Proč? Protože se v trénovacím textu vyskytovalo nejčastěji.
- Vyhlazování Kneser-Ney zavádí pravděpodobnost zohledňující počet kontextů, kde se dané slovo vyskytlo. Tato pravděpodobnost proto odhalí, že ačkoliv se Francisco objevovalo často, pak jenom po slovu San. Naproti tomu stroj se vyskytoval v o mnoho více kontextech, a proto mu bude přidělena vyšší pravděpodobnost.

Pravděpodobnost zohledňující počet kontextů je definována jako:

$$P_{CONTINUATION}(w_i) = \frac{|\{w_{i-1} : count(w_{i-1}, w_i) > 0\}|}{\sum_{w_j} |\{w_{i-1} : count(w_{i-1}, w_j) > 0\}|} \tag{1.14}$$

Čitatel představuje počet slov, která se v trénovacím textu objevila před slovem w_i. Jmenovatel pak celkový počet slov objevujících se před všemi možnými slovy.

$P_{CONTINUATION}$ lze využít jak u interpolace, tak u back-off modelů jako náhrada unigramového modelu. Podrobnější informace se lze dočíst ve (ZDROJ: http://www.ee.columbia.edu/ stanchen/papers/h015a-techreport.pdf).

1.5 Modely maximální entropie

Entropie je minimální průměrný počet bitů potřebný k zakódování popisu hodnoty nějaké náhodné veličiny. Pro náhodnou veličinu X a její distribuci P_X je dána entropie vztahem:

$$H(P_X) = -\sum_x P_X(x) \cdot log_2 P_X(x) \tag{1.15}$$

Ideou modelů **maximální entropie** (nebo též modelů **maxentových**) je najít podmíněné rozdělení pravděpodobnosti, které má za daných podmínek (pozorovaných dat) maximální entropii. Jinými slovy se snažíme najít co nejjednodušší popis na základě toho, co známe – *princip Occamovy břitvy*. Díky tomu se popis co nejvíce blíží rovnoměrnému rozdělení a má tak co nejvyšší entropii.

Z trénovacího textu se budeme snažit napozorovat jen některé důležité vlastnosti, které jsou reprezentovány pomocí binárních funkcí (indikátorů) a nazývají se **rysy** (features). Tyto funkce mohou být např. použity pro reprezentování nám již známých n-gramů. Pro trigram w_1, w_2, w_3 a historii h může funkce vypadat následovně:

$$f_{w_1,w_2,w_3}(h, w) = \begin{cases} 1 & \text{pokud } h \text{ končí } w_1, w_2 \text{ a } w = w_3 \\ 0 & \text{jinak} \end{cases} \tag{1.16}$$

Díky takovému popisu nejsme omezeni jen na n-gramy. Rysy mohou představovat jakoukoliv skutečnost z historie, ať už se jedná třeba o začáteční písmeno prvního

slova věty nebo morfologickou třídu předchozího slova. Na takové rysy můžeme pohlížet jako na jednotlivé modely a budeme hledat jejich vhodné kombinace. Modely maximální entropie ale nestaví modely samostatně, nýbrž vytváří hned jediný kombinovaný model.

Na základě toho nebudeme používat při určování pravděpodobnosti jen posloupnosti slov, ale zavedeme obecnější pojmy. **Kontextem** budeme rozumět jakoukoli historii tj. data, která máme k dispozici v době predikce. **Výsledkem** pak výstup, jejž chceme predikovat. Dvojice kontext a výsledek je označována jako **událost**. V případě modelů čistě se slovy může být událostí n-gram $w_1, \ldots, w_n$, kde predikujeme slovo w_n na základě historie slov $w_1, \ldots, w_{n-1}$.

Výsledný model má následující podobu:

$$P(x|h) = \frac{e^{\sum_i \lambda_i f_i(x,h)}}{Z(h)}, \tag{1.17}$$

kde

- x je predikovaný výsledek
- h je kontext představující historii
- λ_i jsou váhy
- $f_i(x, h)$ jsou funkce reprezentující rysy
- $Z(h)$ je normalizační faktor definovaný takto:

$$Z(h) = \sum_{x_i \in V} e^{\sum_j \lambda_j f_j(x_i,h)} \tag{1.18}$$

- V je množina všech možných výsledků (např. slov)

Během trénování modelu maximální entropie se snažíme naučit optimální váhy λ_i korespondující s funkcemi rysů f_i. To je ekvivalentní hledání odhadu maximální věrohodnosti vah Λ s využitím logaritmu věrohodnostní funkce $\mathcal{L}(X|\Lambda)$ trénovacích dat X. Váhy jsou určovány speciálními metodami, nejčastěji *GIS – Generalized Iterative Scaling* (Darroch, Ratcliff [ZDROJ]) nebo *LBFGS – Limited Memory BFGS* (Liu, Nocedal [ZDROJ]). *BFGS* jsou počáteční písmena příjmení autorů původní metody pro řešení neomezených nelineárních optimalizačních problémů – Broyden-Fletcher-Goldfarb-Shanno.

Stanovení optimálních vah je náročná a složitá operace, která může trvat dlouhou dobu, pokud se k ní přistupuje zcela přímočaře. V každé iteraci algoritmu se musí spočítat normalizační faktor $Z(h)$ pro všechny spatřené kontexty v trénovacích datech. Pro každý kontext je zapotřebí projít přes všechna slova ze slovníku, tedy i přes ta, která se neobjevila v daném kontextu.

Jednou z technik jak snížit složitost počítání normalizačního faktoru jsou vnořené nepřekrývající se rysy – tedy např. n-gramové rysy. Pro ně totiž můžeme

normalizační faktor spočítat takto: Mějme historii trigramového modelu w_{i-1}, w_{i-2}, pak

$$\begin{aligned} Z(w_{i-1}, w_{i-2}) = & \sum_{w_i \in V} e^{f w_i} + \\ & + \sum_{w_i \in V_{w_{i-1}}} (e^{f w_{i-1} w_i} - 1) \cdot e^{f w_i} + \\ & + \sum_{w_i \in V_{w_{i-2} w_{i-1}}} (e^{f w_{i-2} w_{i-1} w_i} - 1) \cdot e^{f w_{i-1} w_i}, \end{aligned} \tag{1.19}$$

kde

- V je slovník
- $V_{w_{i-1}}$ je množina slov pozorovaných po kontextu w_{i-1}
- $V_{w_{i-2} w_{i-1}}$ je množina slov pozorovaných po kontextu $w_{i-2} w_{i-1}$

První suma nezávisí na kontextu a může být předpočítána. Druhá je stejná pro všechny kontexty končící na w_{i-1} a její hodnotu proto můžeme mezi nimi sdílet. Poslední suma vyžaduje součet přes všechna slova spatřená po kontextu $w_{i-2} w_{i-1}$, takových je ale pro většinu kontextů málo.

1.6 Vyhlazování modelů maximální entropie

Stejně jako u n-gramových modelů se u modelů maximální entropie používá vyhlazování. Technice vyhlazování se zde často říká **regularizace**.

Jednou z nejčastějších je metoda **Gaussian priors**, která přidává ke všem vahám rysů apriorní pravděpodobnost s nulovou střední hodnotou a daným rozptylem σ. Optimalizační kritérium modelu se tak změní na:

$$\mathcal{L}'(X|\Lambda) = \mathcal{L}(X|\Lambda) - \sum_i \frac{\lambda_i^2}{2\sigma_i^2} \tag{1.20}$$

Typicky se používá $\sigma_i = \sigma$ pro všechny parametry. Optimální rozptyl je obvykle stanoven z vývojových dat.

Vyhlazování Gaussian Prior je implementováno i v *MaxEnt Toolkitu* od Le Zhanga [ZDROJ], který také budeme využívat pro trénování maxentových modelů s vlastní množinou rysů.

Složitější technikou vyhlazování je $\ell_1 + \ell_2^2$ **regularizace**. Zde má optimalizační kritérium následující podobu:

$$\mathcal{L}_{\ell_1+\ell_2^2}(X|\Lambda) = \mathcal{L}(X|\Lambda) - \frac{\alpha}{D} \sum_i |\lambda_i| - \sum_i \frac{\lambda_i^2}{2\sigma_i^2 D}, \tag{1.21}$$

kde

- D je počet trénovacích pozorování
- α a σ jsou regularizační parametry

Parametry α a σ bývají stanoveny empiricky – např. Chen [ZDROJ]. ([4] z Tanela)

$\ell_1 + \ell_2^2$ regularizaci využívá rozšíření[3] *SRILM Toolkitu* od Tanela Alumäe a Mikko Kurima [ZDROJ]. Toto rozšíření slouží pro trénování maxentových modelů s n-gramovými rysy. Pomocí tohoto rozšíření budeme vyrábět i naše maxentové n-gramové modely.

1.7 Hodnocení modelů

Abychom mohli vyhodnotit a porovnat kvalitu jazykových modelů, potřebujeme zavést taková kritéria, která budou dostatečně vypovídající a vzájemně porovnatelná i při použití různých druhů modelů a metod trénování.

1.7.1 Křížová perplexita

Jedním z hlavních měřítek pro kvalitu jazykového modelu je **křížová perplexita**. Udává, jak moc jsme překvapeni z následujícího pozorování (např. slova) a je dána vztahem:

$$PPL = 2^{H(P_E, P_{LM})}, \tag{1.22}$$

kde $H(P_E, P_{LM})$ je křížová entropie, P_E distribuce pravděpodobnosti trénovacích dat a P_{LM} distribuce pravděpodobnosti jazykového modelu.

Křížová entropie je obdobou entropie ze vzorce (1.15). Křížová ale udává vztah mezi dvěma distribucemi pravděpodobnosti namísto jedné a vypočítá se jako:

$$H(P_E, P_{LM}) = -\sum_x P_E \cdot log_2 P_{LM}(x), \tag{1.23}$$

Distribuce testovacích dat bývá nejčastěji stanovena jako $P_E(x) = \frac{n}{N}$, pokud se x vyskytlo n-krát v testovacích datech velikosti N.

Čím je perplexita nižší, tím lépe umí jazykový model předpovídat následující slovo a tím je samozřejmě lepší.

1.7.2 Adekvátnost a plynulost překladu

Pro hodnocení jazykových modelů se můžeme také opřít o data z ručního hodnocení strojového překladu. Jedna ze zavedených technik totiž hodnotí překlad dvěma kritérii – adekvátností a plynulostí.

- **Adekvátnost** (adequacy) udává, zda překlad zachovává význam, či zda je změněn nebo nekompletní.

[3] Od verze SRILM 1.7.1 je toto rozšíření již součástí základní instalace.

- **Plynulost** (fluency) hodnotí, jak je překlad plynulý, zda má přirozený slovosled apod.

Obě metriky nabývají hodnot $1, 2, \ldots, 5$ a nesou následující význam:

Hodnota	Adekvátnost	Plynulost
1	žádný význam	nesrozumitelný
2	málo z původního významu	neplynulý jazyk
3	dostatečně významu	nepřirozený
4	většina významu	dobrý jazyk
5	veškerý význam	bezchybný jazyk

Tabulka 1.1: Význam jednotlivých hodnocení adekvátnosti a plynulosti

Ruční hodnocení má ale nevýhodu v tom, že je pomalé, drahé a subjektivní. Mezianotátorská shoda ukazuje, že se lidé shodnou více na plynulosti než na adekvátnosti.

V našich experimentech se zkusíme podívat, jak spolu koreluje právě automatické hodnocení (perplexita) s ručním hodnocením plynulosti. Zároveň vyzkoušíme, zda budeme schopni na základě nalezených chyb plynulost dané věty predikovat.

1.8 Aplikace jazykových modelů

Jazykové modely mají široké využití. Používají se například ve strojovém překladu, kde se z nabízených překladových hypotéz snaží vybrat tu, jež vypadá jako nejhezčí věta. Stejnou úlohu mají i v rozpoznávání mluvené řeči nebo tištěného textu. Mezi další aplikace patří např. obnovení diakritiky, korekce pravopisu nebo třeba prediktivní psaní SMS zpráv.

2. Problémy s němčinou

Němčina patří do skupiny flektivních jazyků tj. takových, které gramatické funkce vyjadřují pomocí flexe – ohýbání. Mimo časování a skloňování je pro němčinu typický složitý slovosled. Proto mají tradiční n-gramy s němčinou problémy. V trénovacích datech se nám nemohou objevit všechny gramatické kombinace – např. spojení přídavného a podstatného jména ve všech pádech a kontextech. Techniky vyhlazování modelů gramatiku nesledují explicitně a mají proto obtíže za přídavné jméno daného tvaru doporučit podstatné jméno vhodného rodu, pádu a čísla.

2.1 Skloňování jmen

Německá gramatika zná 4 pády – *nominativ*, *genitiv*, *dativ* a *akuzativ*. Skloňování probíhá pomocí členů a koncovek.

- **Podstatná jména**
  Podstatná jména jsou skloňována především za pomoci členů, koncovka *-(e)s* se přidává ve druhém pádě rodu mužského a středního čísla jednotného a koncovka *-(e)n* ve třetím pádě čísla množného. *Např. der Hund, des Hundes.* Takto se skloňuje většina podstatných jmen.

  Mimo pravidelného (silného) skloňování existuje ještě skloňování slabé. Slabé skloňování přijímá koncovku *-en* ve všech pádech kromě prvního. *Např. der Student, des Studenten.* Tímto způsobem se obvykle skloňují podstatná jména rodu mužského označující živé bytosti, příslušníky národností nebo slova cizího původu.

- **Přídavná jména**
  U přídavných jmen je situace ještě složitější. Mimo členu se v naprosté většině případů mění i koncovka. Ta je závislá mimo jiné i na tom, zda předcházel člen určitý nebo neurčitý. Jednoduše se dá však říci, že koncovka má za úkol vyjádřit rod, pokud není zřejmý ze členu. *Např. ein schönes Haus x das schöne Haus.*

2.2 Pořádek slov

V němčině se rozlišují dva pořádky slov, a sice pořádek přímý a pořádek nepřímý. Speciálním případem je pak ještě pořádek slov ve vedlejší větě.

- **Pořádek přímý**
  Pořádek přímý se vyznačuje pořadím – podmět, přístudek na začátku věty. Používá se hlavně v oznamovacích větách
  Např. Jsem doma. – Ich bin zu Hause.

- **Pořádek nepřímý**
  Pořádek nepřímý se používá především v tázacích větách. Často se ale používá i ve větách oznamovacích, kde se předsune větný člen na začátek věty pro zdůraznění. Pořadí podmětu a přísudku se pak mění a podmět následuje hned za přísudkem.
  Např. Znáš ji – Kennst du sie? Dnes jsem doma. – Heute bin ich zu Hause.

- **Pořádek ve vedlejší větě**
  Vedlejší věty mají speciální pořádek slov. Po podřadící spojce následuje hned podmět a sloveso je umístěno až na konci věty.
  Např. Nevím, jestli ho zná. – Ich weiß nicht, ob sie ihn kennt.

2.3 Větný rámec

Němčina dává v mnoha případech nějaké slovo na konec věty – nejčastěji se jedná o sloveso nebo odlučitelnou předponu. Tomuto jevu se říká větný rámec a k jeho tvorbě dochází v několika případech:

- **Způsobová slovesa**
  Po způsobovém slovesu patří plnovýznamové sloveso vždy na konec věty ve formě infinitivu.
  Např: Neumíme to říct. – Wir können es nicht sagen.

- **Minulý čas – perfektum**
  Perfektum se v němčině tvoří pomocí pomocného slovesa a příčestí minulého. Příčestí minulé patří na konec věty.
  Např. Neřekl jsem to. – Ich habe es nicht gesagt.

- **Budoucí čas**
  Budoucí čas se tvoří pomocným slovesem werden a infinitivem, který umísťujeme na konec věty.
  Např. Řeknu mu to. – Ich werde es ihm sagen.

- **Odlučitelné předpony sloves**
  Mnoho německých sloves má odlučitelnou předponu, která se v určitých tvarech od zbytku slovesa odlučuje a patří opět na konec věty.
  Např. Zítra odjedu domů. – Morgen fahre ich nach Hause ab. (sloveso abfahren)

- **Vedlejší věty**
  Jak už bylo zmíněno, vedlejší věty mají speciální pořádek slov a určité sloveso v nich patří na konec věty.
  Např. Ptám se, jestli jsi doma. – Ich frage, ob du zu Hause bist.

K tvorbě větného rámce dochází i v dalších případech, jako je třeba trpný rod nebo čas předminulý (*plusquamperfektum*). Platí však stejná pravidla, tj. sloveso plnovýznamové nebo příčestí minulé patří na konec věty.

Vzdálenost mezi pomocným slovesem a slovesem plnovýznamovým nebo příčeštím minulým může být poměrně velká a běžné n-gramy o několika slovech nemohou

tuto závislost zachytit. Navíc by jazykový model musel vidět přesně takový n-gram (tj. přesně stejná slova) v trénovacích datech.

2.4 Pozorování na hypotézách strojového překladu

Na základě výše uvedeného popisu gramatických jevů, u kterých jsme předpokládali, že budou činit největší problémy, jsme pozorovali desítku původně anglických vět a jejich návrhy strojových překladů do němčiny (tzv. hypotézy). Pro každou větu jsme měli k dispozici 100 hypotéz. Zkoumali jsme především, v čem se jednotlivé návrhy liší, neboť to pomyslně znamená právě ty oblasti, kde si není jazykový model příliš jistý.

Z pozorování vyplynulo, že hypotézy se často liší jen ve tvarech přídavných jmen nebo členů. Modely si tak nebyly schopné poradit se skloňováním. Větný rámec se nepovedl takřka nikde, ba co víc, v některých případech plnovýznamové sloveso či příčestí minulé ve hypotéze úplně chybělo.

2.4.1 Příklady konkrétních hypotéz

Zde uvedeme několik příkladů hypotéz, ve kterých budou červeně zvýrazněné některé gramatické chyby. Pokud se daný jev v některé další hypotéze povedl, bude označen zeleně. Zaměříme se vždy na nějaký gramatický jev, nikoliv na samotný špatný překlad některých slov.

- *Barack Obama erhält als vierte US - Präsident den Friedensnobelpreis*
- *Barack Obama wird als vierte US - Präsident den Friedensnobelpreis*
- *Barack Obama bekommt als vierte US - Präsident den Friedensnobelpreis*
- *Barack Obama erhält als vierter US - Präsident den Friedensnobelpreis*

Jak vidíme v první, druhé i třetí hypotéze, číslovka *vierte* je špatně vyskloňovaná. Má špatnou koncovku, neboť před ní nestojí určitý člen a patří k podstatnému jménu rodu mužského. Z tohoto důvodu je správně tvar označený zeleně ve čtvrté větě. Druhá hypotéza ještě navíc obsahuje sloveso *werden*, které pravděpodobně bylo pokusem o budoucí čas. Plnovýznamové sloveso však chybí.

- *US - Präsident Barack Obama přiletí des norwegischen in Oslo auf 26 Stunden , um sich hier als vierte US - Präsident in der Geschichte übernahm den Friedensnobelpreis .*

V této větě jsme zvýraznili dvojici *um* a *übernahm*. Pokud byla vedlejší věta uvozena spojkou *um*, pak se zřejmě mělo jednat o zkrácenou vedlejší větu a měl následovat infinitiv s *zu* na konci věty. Mimo tohoto jevu se ve větě vyskytují samozřejmě další gramatické chyby, jako třeba již zmíněné špatné skloňování.

- *Diplom , Medaille und Scheck auf 1,4 Millionen US - Dollar erhalten hat , unter anderem für die außergewöhnliche Anstrengungen zur Stärkung der Diplomatie und Zusammenarbeit zwischen den Völkern .*
- *Diplom , Medaille und Scheck auf 1,4 Millionen Dollar wird unter anderem für die außergewöhnliche Anstrengungen zur Stärkung der Diplomatie und Zusammenarbeit zwischen den Völkern .*

Zde byl v první hypotéze vytvořen větný rámec s pořádkem slov vedlejší věty, ačkoliv zde být neměl, neboť se o vedlejší větu nejedná. Druhá hypotéza obsahuje sloveso *werden*, které má zřejmě funkci slovesa pomocného. Infinitiv nebo příčestí minulé už ale chybí. V obou hypotézách je určité sloveso ve třetí osobě čísla jednotného, přestože takový podmět, s nímž by měl přísudek utvořit shodu, se ve větě nenachází.

- *der Präsident hat sich diesem Thema vermeiden will , weil sie erkennt , dass die Kosten übernimmt wie ein Präsident , der derzeit ein Krieg in zwei Ländern .*

V první klauzi se vyskytují dvě určitá slovesa, v poslední naopak sloveso úplně chybí. Třetí klauze je vedlejší větou a špatný pořádek slov zde staví *die Kosten* do role podmětu neshodujícího se s přísudkem, neboť *die Kosten* je pomnožné podstatné jméno a *übernimmt* je v čísle jednotném.

3. Modely s morfologickými značkami

Jak jsme popsali v předchozí kapitole, německá gramatika je díky požadavku na shodu jmen, pořádku slov a tvorbě větného rámce složitá. Běžné n-gramové modely, které sledují jen posloupnosti po sobě jdoucích slov nezachycují gramatiku jako takovou. Vyzkoušíme proto, zda dopadnou lépe modely, které budeme trénovat a testovat na datech, v nichž nahradíme slova různými morfologickými značkami. Budeme na nich zkoumat, jak spolu souvisí perplexita a ručně hodnocená plynulost.

Předpokladem je, že pokud nahradíme slova jejich morfologickými značkami, dojde ke zhuštění[1] dat a model se bude schopen doporučit slovo v patřičném tvaru vycházejícím z morfologické analýzy.

3.1 Zdrojová data

Modely budeme trénovat na německých datech z WMT[2] 2012 – News Commentary[3].

Pro otestování použijeme data z výstupů překladových systémů z WMT 2006, které se účastnily překladu z angličtiny do němčiny. Volíme takto starší ročník, neboť některé z překladových hypotéz obsahovaly ručně ohodnocenou plynulost překladu. Právě takové hypotézy použijeme pro zkoumání korelace perplexity a plynulosti. Celkem jich je k dispozici 2028, z toho 58 je hodnoceno dvakrát a 2 třikrát, celkem tedy 2090 hodnocení. Následující tabulka 3.1 ukazuje přesné počty hodnocení hypotéz:

Plynulost	Počet hodnocení
1	150
2	445
3	932
4	387
5	176

Tabulka 3.1: Počty hodnocení jednotlivých plynulostí

Počty jednotlivých plynulostí nejsou vyvážené a zejména kandidátů hodnocených 1 a 5 je poměrně málo. Výsledky je proto potřeba brát s příslušnou rezervou.

[1] Zhuštěním dat rozumíme zvýšení počtu správných kombinací slov (v tomto případě morfologických značek), které můžeme pozorovat v trénovacích datech, vůči všem možným správným kombinacím.

[2] WORKSHOP ON STATISTICAL MACHINE TRANSLATION

[3] http://statmt.org/wmt12/training-parallel.tgz - soubor news-commentary-v7.de-en.de

Hypotézy pochází ze 400 různých vět překládaných osmi systémy. Plynulost posuzovali 4 hodnotitelé, kteří se u 58 hypotéz hodnocených dvěma z nich shodli následovně:

Shoda	Počet hypotéz	V procentech
shodli se	34	58.6 %
lišili se o 1	19	32.8 %
lišili se o 2	4	6.9 %
lišili se o 3	1	1.7 %

Tabulka 3.2: Přehled shody dvou různých hodnotitelů v posouzení plynulosti

Dvě hypotézy hodnocené třikrát byly taktéž posouzeny dvěma hodnotiteli, třetí hodnocení bylo vždy vykonáno jedním z nich a slouží pouze jako kontrola – ta dopadla v obou případech úspěšně, tedy udělením stejného hodnocení. U jedné hypotézy se hodnotitelé shodli, u druhé se lišili o 1.

Ačkoliv nevýhodou ručního hodnocení je vždy určitá míra subjektivity, výše uvedená tabulka 3.2 uvádí nadpoloviční shodu a budeme-li brát jako uspokojivé i případy, kdy se hodnocení lišilo o 1, pak jsme dokonce lehce nad 90 %.

3.2 Princip experimentů

Princip experimentů bude následující

- na trénovacích datech provedeme morfologickou analýzu
- slova trénovacího textu nahradíme odpovídajícími morfologickými značkami
- natrénujeme standardní 6-gramový model a maxentový 6-gramový model
- stejně jako trénovací data, připravíme i data testovací
- změříme perplexitu na testovacích datech a provedeme vyhodnocení

Pro morfologickou analýzu použijeme parser ParZu[4]. Jedná se o nástroj, který kombinuje tagger Tree-Tagger (ZDROJ) a morfologický analyzátor Morphisto (ZDROJ). Pro Morphisto použijeme předkompilovaný model `morphisto-02022011.a`[5]. S využitím nástroje Morphisto uvádí ParZu přesnost parsingu 86.5 % (ZDROJ). Allauzen a Bonneau-Maynard uvádějí přenost taggingu u Tree-Taggeru okolo 96 %(ZDROJ - http://www.lrec-conf.org/proceedings/lrec2008/pdf/856_paper.pdf)

ParZu spouští nejprve vlastní tokenizér. Vzhledem k tomu, že data z WMT 06, která používáme, jsou již tokenizovaná, tento tokenizér vynecháme a pouze upravíme formát – jeden token na řádku, věty oddělené prázdným řádkem. Data z WMT 12 tokenizovaná nejsou, proto u nich tokenizér necháme běžet.

[4]The Zurich Dependency Parser for German, formálněji známý také pod názvem Pro3GresDE

[5]ZDROJ

Příklad výstupu ParZu:

1	Der	der	ART	ART	Def|Masc|Nom|Sg	3	det	_ _
2	schönste	schön	ADJA	ADJA	Sup|Masc|Nom|Sg|Sw|	3	attr	_ _
3	Satz	Satz	N	NN	Masc|_|Sg	0	root	_ _
4	in	in	PREP	APPR	Dat	3	pp	_ _
5	der	der	ART	ART	Def|Fem|Dat|Sg	7	det	_ _
6	ganzen	ganz	ADJA	ADJA	Pos|Fem|Dat|Sg|_|	7	attr	_ _
7	Welt	Welt	N	NN	Fem|Dat|Sg	4	pn	_ _
8	.	.	$.	$.	_	0	root	_ _

Pro účely následujících experimentů nás bude zajímat pátý a šestý sloupec – rozšířený slovní druh a morfologická analýza.

Nahrazení slov různými morfologickými značkami z výstupu ParZu zajišťuje program **MorfModel**, který vzniknul jako součást této práce a je k dispozici na přiloženém DVD.

Standardní n-gramové modely budeme vyrábět v již zmíněném toolkitu SRILM nástrojem `ngram-count` s výchozím nastavením, tj. technika back-off s Good-Turingovým vyhlazováním. Pro trénování modelů maximální entropie (maxentových) využijeme již taktéž zmíněné rozšíření toolkitu SRILM od Tanela Alumäe a Mikko Kurima. Pro vyhlazování používá toto rozšíření popsanou metodu regularizace $\ell_1 + \ell_2^2$.

3.3 Způsob vyhodnocení

U každého natrénovaného modelu bude změřena perplexita pro každou větu zvlášť. Výsledky pak vykreslíme do grafu společně s odpovídající plynulostí pro znázornění jejich korelace. Pro lepší znázornění bude u každé plynulosti vykreslen boxplot[6] znázorňující oblast s nejvyšším výskytem hypotéz ohodnocených danou perplexitou. Čím vyšší je plynulost, tím nižší by měla být perplexita. Boxploty pro skupiny vět s daným hodnocením plynulosti by proto měly klesat. Mediány boxplotů bude ještě proložena přímka, aby byla tendence zřetelnější.

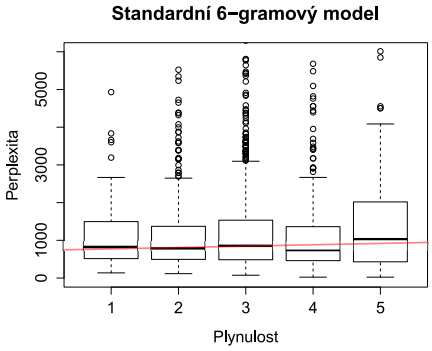
Srovnání standardních a maxentových modelů provedeme graficky umístěním dvou grafů přes sebe vykreslených odlišnou barvou. Mimo toho provedeme ještě srovnání z hlediska výpočetních nároků[7]. Na závěr uvedeme shrnutí popisující naměřené hodnoty ze všech modelů a provedeme srovnání pomocí korelačních koeficientů.

[6]Boxplot (krabicový graf) – vykreslí obdélník v oblasti, kde se vyskytuje 50 % hodnot. Horní a dolní hranice obdélníku odpovídají hornímu a dolnímu kvartilu. Uprostřed obdélníku se vykresluje ještě tučně medián. Vertikálně vedou z obdélníků tzv. vousy, jejichž hranice leží v maximální (minimální) hodnotě, maximálně však v 1.5 násobku mezikvartilového rozmezí (horní − dolní kvartil) nad horním nebo pod dolním kvartilem. Body mimo tyto hranice se nazývají extrémní hodnoty a jsou vykresleny samostatně jako body.

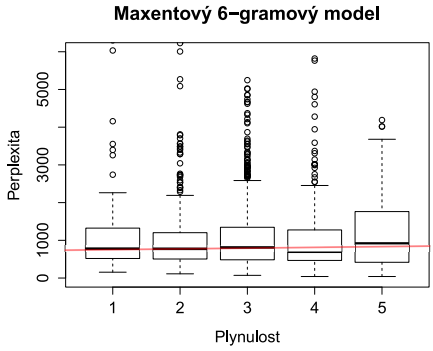
[7]Všechny modely trénovány na netbooku Asus EEE 1201N – Intel Atom 330 1.6 GHz dual core, 2 GB RAM.

3.4 Běžné modely se slovy

Jako první jsme zkusili natrénovat 6-gramové modely se slovy, abychom viděli, jak spolu souvisí perplexita a plynulost u takových modelů a mohli výsledek použít pro další srovnání.

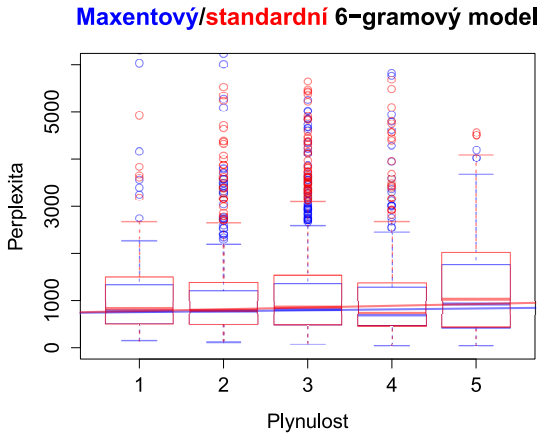


Obrázek 3.1: Standardní 6-gramový model se slovy



Obrázek 3.2: Maxentový 6-gramový model se slovy

Z obou grafů (obrázky 3.1, 3.2) je patrné, že plynulost nekoreluje s perplexitou tak, jak bychom chtěli. Perplexita by měla se zvyšující se plynulostí klesat – čím nižší perplexita, tím lepší a tedy i plynulejší překlad. Na obou grafech však boxploty neklesají, nýbrž kolísají. Dokonce hypotézy hodnocené plynulostí 5 mají rozsah nejčastějších perplexit nejvyšší. To ale může být částečně způsobeno malým počtem hypotéz ohodnocených plynulostí 5.



Obrázek 3.3: Porovnání modelů se slovy

Srovnání ukazuje, že maxentový model dopadl o něco lépe, neboť jednotlivé boxploty mají nižší horní hranici nejčastějších perplexit než v případě standardních modelů. Rozdíly ve spodních hranicích jsou zanedbatelné. I proložená přímka stoupá v případě standardního n-gramového modelu více než v případě maxentového (obrázek 3.3).

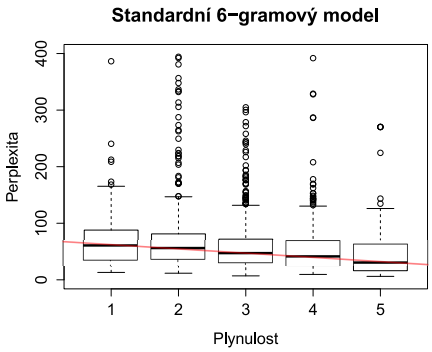
Čas nutný k natrénování se však výrazně liší – natrénování standardního n-gramového trvalo zhruba 3 minuty oproti téměř 12 hodinám u modelu maxentového.

3.5 Rozšířený slovní druh + morfologické značky

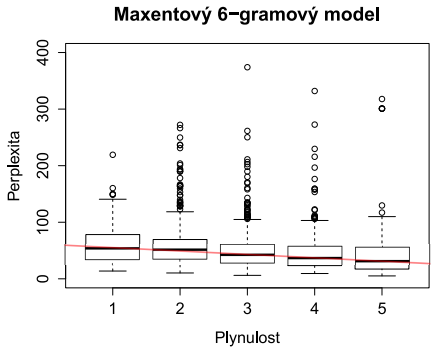
Nyní zkusíme natrénovat model, kde slova nahradíme rozšířeným slovním druhem a všemi morfologickými značkami z výstupu ParZu. Pro oddělení použijeme dvojtečku.

Příklad věty:

Die	unabhängige	Justiz	
ART:Def|Fem|Akk|Sg	ADJA:Pos|Fem|Akk|Sg|_|	NN:Fem|Akk|Sg	
und	die	freien	
KON:_	ART:Def|Neut|Akk|Pl	ADJA:Pos|Neut|Akk|Pl|_|	
Medien	zu	unterdrücken	.
NN:Neut|Akk|Pl	PTKZU:_	VVINF:_	$.:_

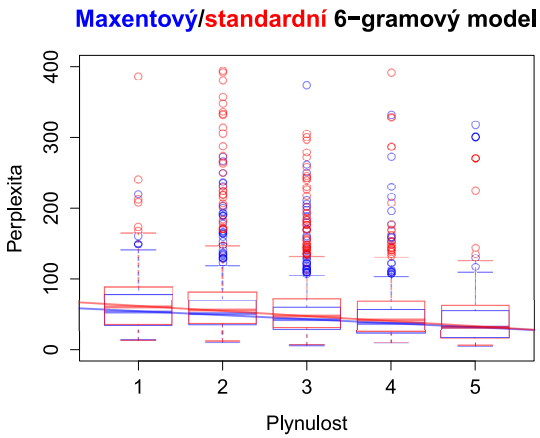


Obrázek 3.4: Standardní 6-gramový model – rozšířený slovní druh + morfologické značky



Obrázek 3.5: Maxentový 6-gramový model – rozšířený slovní druh + morfologické značky

Oba modely (obrázky 3.4, 3.5) dopadly lépe než modely trénované na slovech. Boxploty vykazují lehce klesavou tendenci.



Obrázek 3.6: Porovnání modelů – rozšířený slovní druh + morfologické značky

Maxentové modely opět, co se perplexity týče, dopadají lépe než standardní n-gramové. Avšak proložená přímka klesá u standardních modelů strměji (obrázek 3.6). Z hlediska výpočetních nároků jsou na tom stadardní n-gramy oproti maxentovým znovu výrazně lépe – 72 sekund proti takřka 8 hodinám.

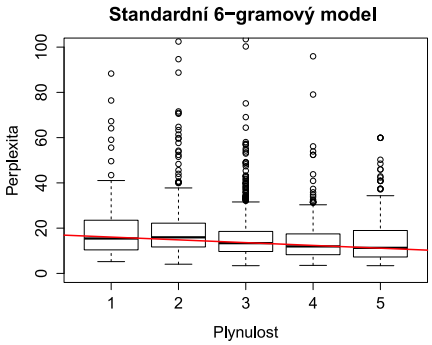
Tyto modely sice obsahují morfologickou analýzu, ale nerozumí jejímu obsahu. Nedokáží rozlišit, zda se sousední jména shodují v rodě, ale už ne v pádě apod. Natrénování maxentového modelu na slovech s rysy, které by vycházely z morfologické analýzy (rod, pád, číslo, ...), rozšíření SRILMu od Tanela Alumäe a Mikko Kurima bohužel neumožňuje a jiné dostupné toolkity, např. Maxent toolkit od LeZhanga, nejsou vhodné z hlediska výpočetních nároků na velká data. Zkusíme proto natrénovat další n-gramové modely, ve kterých nahradíme slova vždy jedním z potenciálních rysů. Takové modely by potom bylo možné kombinovat.

3.6 Rozšířený slovní druh

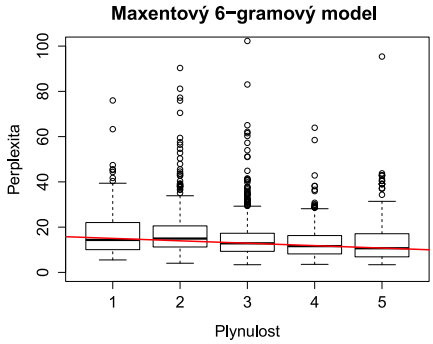
Zkusíme data ještě více zhustit a slova nahradit jen jejich rozšířeným slovním druhem. U členů upřesníme, zda se jedná o člen určitý nebo neurčitý přidáním `Def` nebo `Indef` za `ART`.

Příklad věty:

Die	unabhängige	Justiz	
ARTDef	ADJA	NN	
und	die	freien	
KON	ARTDef	ADJA	
Medien	zu	unterdrücken	.
NN	PTKZU	VVINF	$.

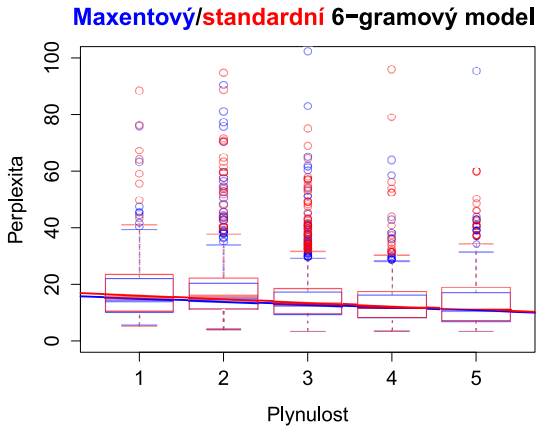


Obrázek 3.7: Standardní 6-gramový model – rozšířený slovní druh



Obrázek 3.8: Maxentový 6-gramový model – rozšířený slovní druh

Modely dopadly (obrázky 3.7, 3.8) o něco hůře než v případě, kdy byl rozšířený slovní druh ještě upřesněn další analýzou. Bez dalšího určení nemůžeme např. kontrolovat správné vyskloňování, model proto neaspiruje na kontrolu flexe, ale jen slovosledu. Stále je to však lepší než při natrénování na slovech.



Obrázek 3.9: Porovnání modelů – rozšířený slovní druh

V porovnání je na tom maxentový model z hlediska perplexity opět mírně lépe. Ovšem stejně jako u předchozích modelů i proložená přímka klesá u standardního n-gramového modelu strměji (obrázek 3.9).

Z hlediska výpočetních nároků je tentokrát rozdíl menší. Standardní n-gramový model potřeboval pro natrénování 22 sekund, maxentový 14 minut.

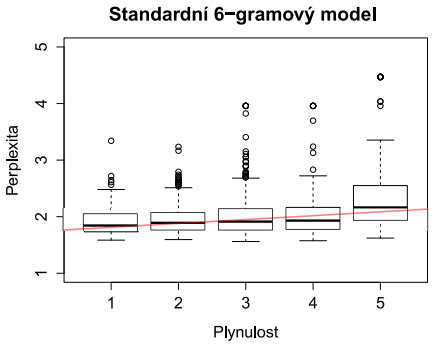
Zde je vidět, nakolik ovlivňuje velikost slovníku dobu trénování maxentových modelů. Oproti modelům se všemi morfologickými značkami potřebovaly standardní n-gramy 3.27x méně času, maxentové 34.29x, což je obrovský rozdíl.

3.7 Rod

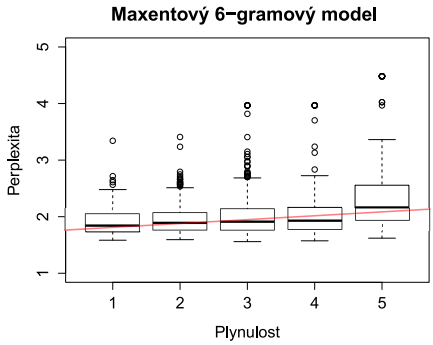
První z modelů s jedinou morfologickou značkou budou modely obsahující rod. Slova budou nahrazena znakem `w`, ke kterému se připojí patřičný rod, lze-li u slova určit. Tím dojde k dalšímu zhuštění dat a velikost slovníku se zmenší na pouhá 4 slova.

Příklad věty:

Die	unabhängige	Justiz	und	die	freien	Medien
wFem	wFem	wFem	w	wNeut	wNeut	wNeut
zu	unterdrücken	.				
w	w	w				

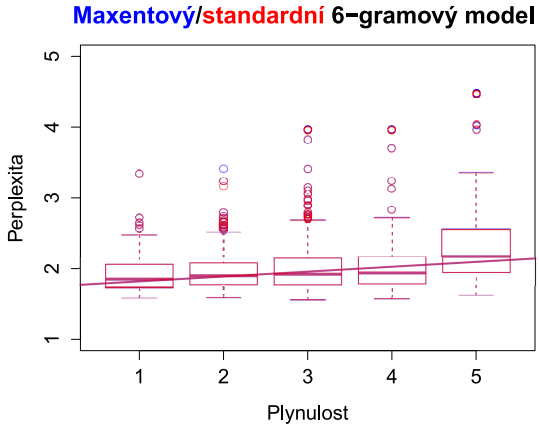


Obrázek 3.10: Standardní 6-gramový model – rod



Obrázek 3.11: Maxentový 6-gramový model – rod

Modely ale dopadly přesně obráceně, než jsme chtěli. Boxploty neklesají, ale stoupají (obrázky 3.10, 3.11). S trochou nadsázky by se dalo říct, že zde čím je perplexita vyšší, tím je lepší plynulost. Ovšem skutečnost je taková, že perplexita pro plynulosti 1-4 vyšla velmi podobně a nejsme pouze na základě ní schopni rozlišit, o kterou plynulost by se mělo jednat.



Obrázek 3.12: Porovnání modelů – rod

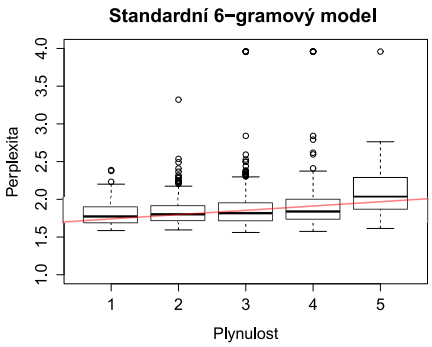
Rozdíl mezi standardními a maxentovými n-gramovými modely je prakticky neznatelný (obrázek 3.12). Stejně je tomu tentokrát i u výpočetních nároků. Standardní n-gramy potřebovaly k natrénování 3 sekundy, maxentové n-gramy pak sekundy 4.

3.7.1 Rod stejný s předchozím

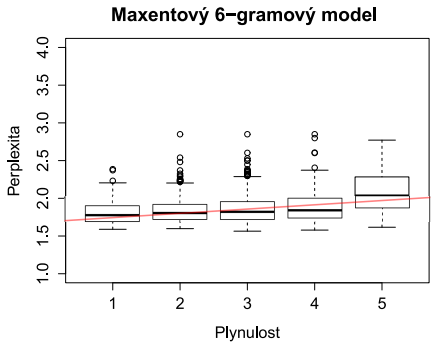
Ačkoliv modely pouze s rodem nebyly úspěšné, zkusíme ještě slova nenahrazovat jenom rodem daného slova, ale pokusíme se sledovat, zda se rody za sebou shodují. Slova tedy nahradíme opět písmenem `w`, k němuž přidáme slovo `rod`, lze-li u slova určit, a slovo `stejne`, pokud lze jednak rod u slova určit a jednak, pokud se rod shoduje s předchozím slovem.

Příklad věty:

Die	unabhängige	Justiz	und	die	freien	Medien
wrod	wstejny	wstejny	w	wrod	wstejny	wstejny
zu	unterdrücken	.				
w	w	w				

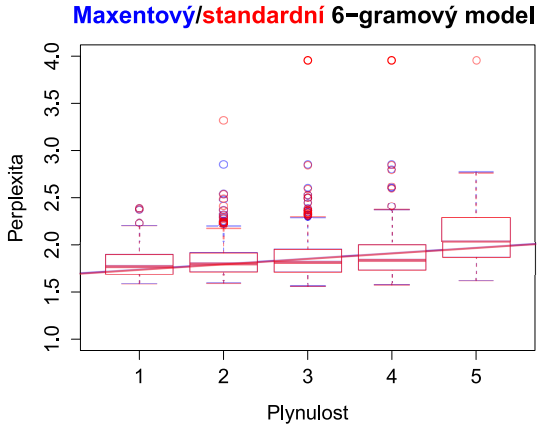


Obrázek 3.13: Standardní 6-gramový model – rod stejný s předchozím



Obrázek 3.14: Maxentový 6-gramový model – rod stejný s předchozím

Výsledky jsou ale stejně špatné jako v případě samotného rodu. Grafy se sobě velmi podobají (obrázek 3.13, 3.14).



Obrázek 3.15: Porovnání modelů – rod stejný s předchozím

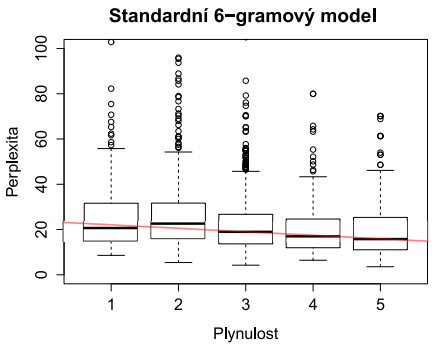
Rozdíly mezi standardním n-gramovým a maxentovým n-gramovým modelem je taktéž nepatrný (obrázek 3.15). Doba nutná k natrénování obou typů byla v tomto případě shodně 3 sekundy.

3.7.2 S rozšířeným slovním druhem

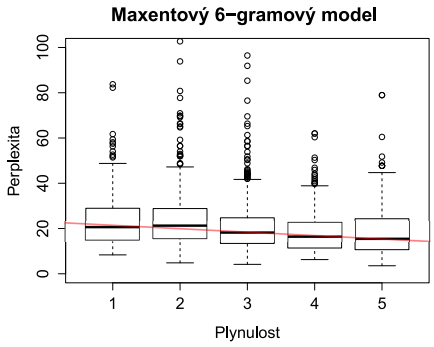
Jelikož samotný rod byla pro model nedostatečná informace, zkusíme namísto písmene `w` slova nahrazovat rozšířeným slovním druhem a k němu přidávat za dvojtečku rod.

Příklad věty:

Die	unabhängige	Justiz	und	die	freien	Medien
ART:Fem	ADJA:Fem	NN:Fem	KON	ART:Neut	ADJA:Neut	NN:Neut
zu	unterdrücken	.				
PTKZU	VVINF	$.				

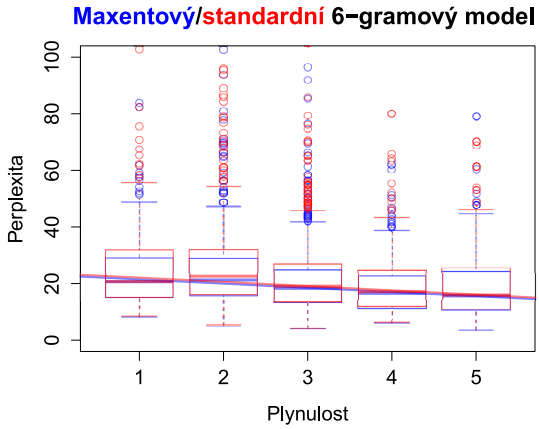


Obrázek 3.16: Standardní 6-gramový model – číslo



Obrázek 3.17: Maxentový 6-gramový model – číslo

S rozšířeným slovním druhem už modely dopadly lépe, přesto perplexita s rostoucí plynulostí neklesá nijak výrazně, což dokazuje proložená přímka (obrázky 3.16, 3.17).



Obrázek 3.18: Porovnání modelů – rozšířený slovní druh + rod

Ve srovnání jsou oba typy modelů na tom podobně, maxentové dostávaly jen o něco málo nižší perplexitu (obrázek 3.18). Z hlediska výpočetní náročnosti je ale rozdíl velký – 36 sekund standardní n-gramový model naproti 27 minutám u modelu maxentového.

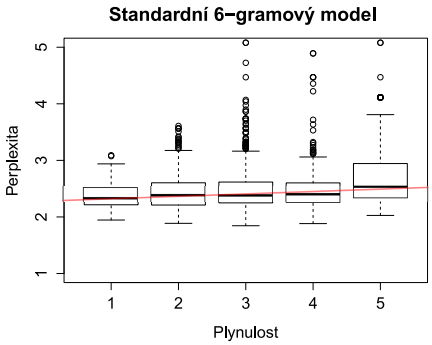
3.8 Číslo

Stejné modely zkusíme natrénovat i v případě čísla. Jako první znovu zkusíme, zda bude modelu postačovat informace pouze o čísle daného slova tj. `wSg`, `wPl`

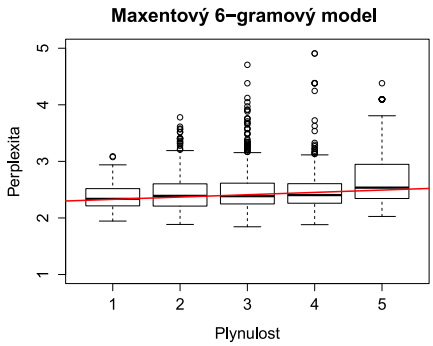
nebo `w`.

Příklad věty:

Die	unabhängige	Justiz	und	die	freien	Medien
wSg	wSg	wSg	w	wPl	wPl	wPl
zu	unterdrücken	.				
w	w	w				

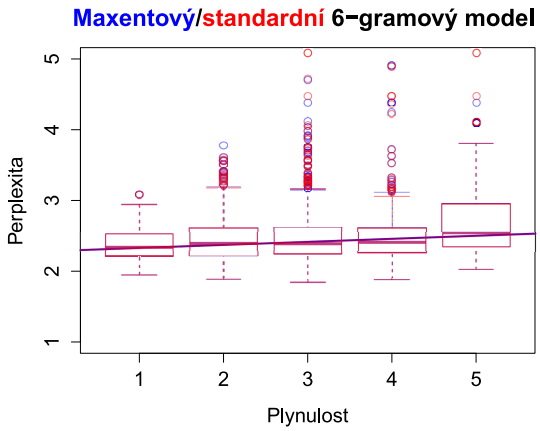


Obrázek 3.19: Standardní 6-gramový model – číslo



Obrázek 3.20: Maxentový 6-gramový model – číslo

Výsledky ale dopadly obdobně špatně jako v případě rodu. Pomocí perplexity nejsme schopni rozlišit, o jakou plynulost by se mělo jednat (obrázky 3.19, 3.20).



Obrázek 3.21: Porovnání modelů – číslo

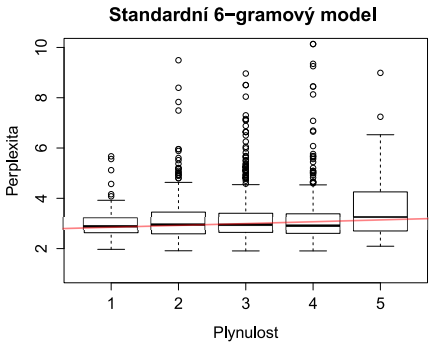
Oba typy modelů dopadly takřka stejně (obrázek 3.21). Stejná byla i doba nutná k natrénování – shodně po třech sekundách.

3.8.1 Přidání osoby

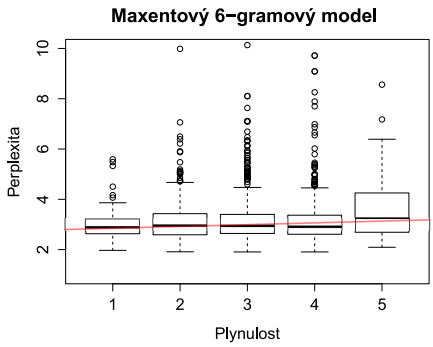
Číslo se bude často pojit s nějakou osobou. Zkusíme proto tuto informaci k číslu přidat. Slovo nahradíme písmenem `w`, poté bude následovat osoba a číslo, jdou-li u daného slova určit.

Příklad věty:

Zur	Belohnung	erhielt	Pakistan	von	Amerika
w	wSg	w3Sg	wSg	w	wSg
finanzielle	Unterstützung	und	Waffen	.	
wSg	wSg	w	wPl	w	

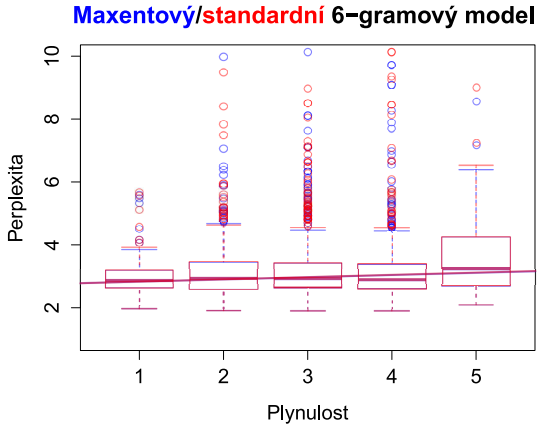


Obrázek 3.22: Standardní 6-gramový model – osoba + číslo



Obrázek 3.23: Maxentový 6-gramový model – osoba + číslo

Modely ale nedopadly o nic lépe. Proložené přímky opět mírně stoupají, namísto aby klesaly (obrázky 3.22, 3.23).



Obrázek 3.24: Porovnání modelů – osoba + číslo

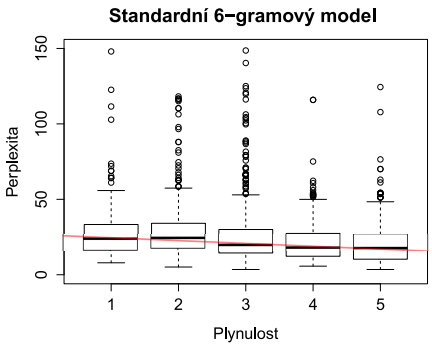
Ve srovnání obou typů modelů opět nejsou patrné výrazné rozdíly (obrázek 3.24). Z hlediska výpočetních nároků se ale tentokrát trochu liší. Standardní n-gramové potřebovaly k natrénování 4 sekundy, maxentové 11 sekund.

3.8.2 S rozšířeným slovním druhem

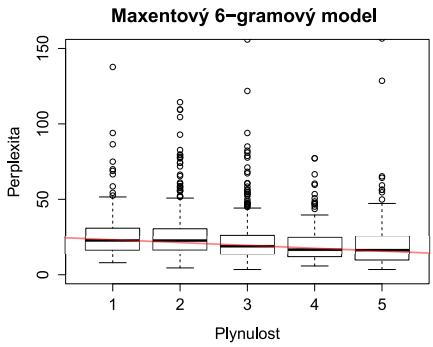
Vzhledem k tomu, že přidání osoby žádné patrné zlepšení nepřineslo, zkusíme znovu přidat rozšířený slovní druh. Slova tedy budeme nahrazovat jejich druhem a číslem, lze-li určit.

Příklad věty:

Zur	Belohnung	erhielt	Pakistan	von	Amerika
APPRART	NN:Sg	VVFIN:Sg	NE:Sg	APPR	NE:Sg
finanzielle	Unterstützung	und	Waffen	.	
ADJA:Sg	NN:Sg	KON	NN:Pl	$.	

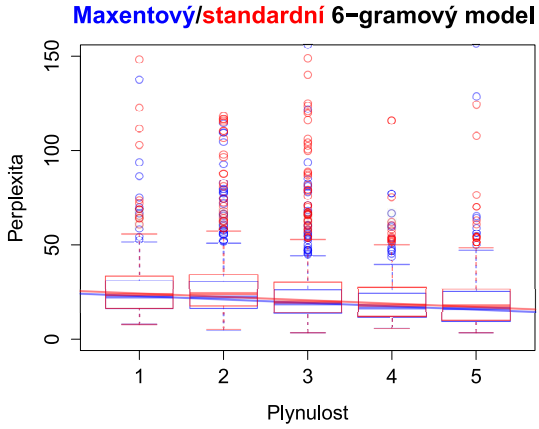


Obrázek 3.25: Standardní 6-gramový model – rozšířený slovní druh + číslo



Obrázek 3.26: Maxentový 6-gramový model – rozšířený slovní druh + číslo

Modely s rozšířeným slovním druhem (obrázky 3.25, 3.26) dopadly znovu lépe. Ovšem výsledky stále nejsou nijak dobré.



Obrázek 3.27: Porovnání modelů – rozšířený slovní druh + číslo

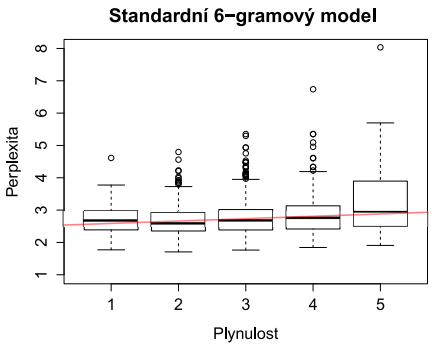
V porovnání dostávaly maxentové modely o něco nižší perplexitu (obrázek 3.27). Čas potřebný k natrénování byl u standardních n-gramových modelů 33 s, u maxentových 24 minut.

3.9 Pád

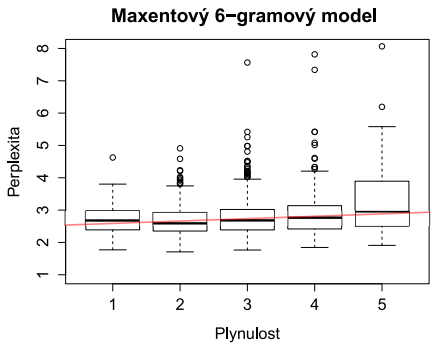
Jako poslední zkusíme ještě natrénovat modely, kde slova nahradíme znovu písmenem `w` a přidáme k němu pád, lze-li u daného slova určit.

Příklad věty:

Zur	Belohnung	erhielt	Pakistan	von	Amerika
wDat	w	w	wNom	wDat	wDat
finanzielle	Unterstützung	und	Waffen	.	
wAkk	wAkk	w	wAkk	w	

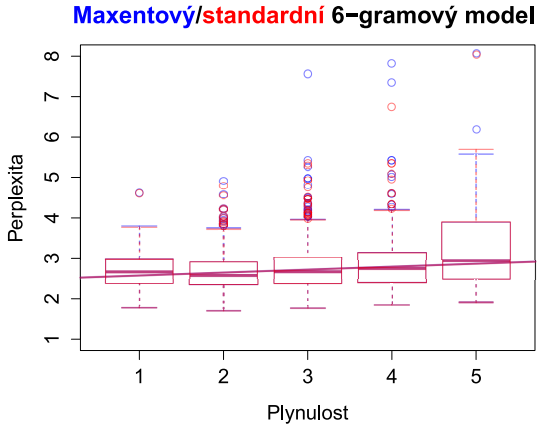


Obrázek 3.28: Standardní 6-gramový model – pád



Obrázek 3.29: Maxentový 6-gramový – pád

Výsledky opět nejsou dobré (obrázky 3.28, 3.29) a jen potvrzují, že poskytnutí modelu značky pouze z jedné morfologické kategorie je nedostatečná informace.



Obrázek 3.30: Porovnání modelů – pád

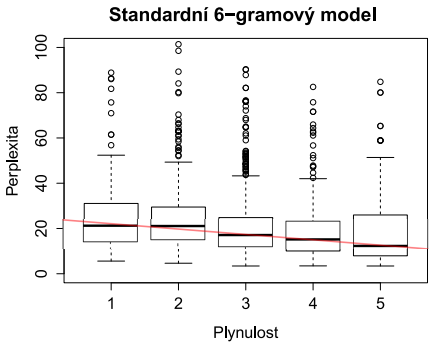
V porovnání jsou taktéž standardní n-gramové modely s modely maximálně entropie srovnatelné, bez větších rozdílů (obrázek 3.30). Natrénování trvalo shodně 4 sekundy.

3.9.1 S rozšířeným slovním druhem

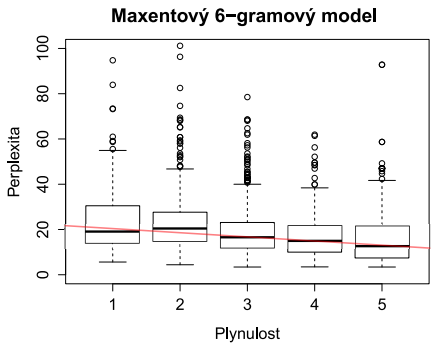
Jako v případech předchozích modelů se značkami z jedné morfologické kategorie, zkusíme ještě přidat k pádu ještě rozšířený slovní druh.

Příklad věty:

Zur	Belohnung	erhielt	Pakistan	von	Amerika
APPRART:Dat	NN	VVFIN	Ne:Nom	APPR:Dat	NE:Dat
finanzielle	Unterstützung	und	Waffen	.	
ADJA:Akk	NN:Akk	KON	NN:Akk	$.	

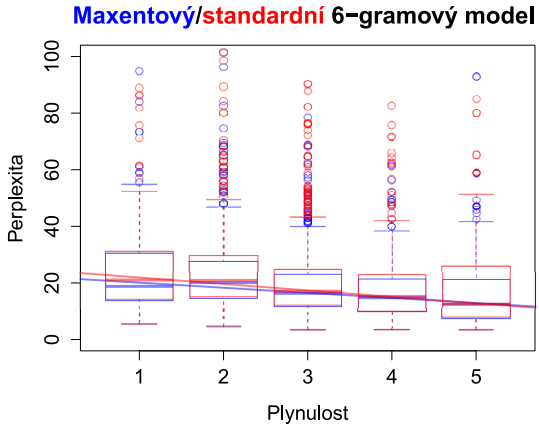


Obrázek 3.31: Standardní 6-gramový model – rozšířený slovní druh + pád



Obrázek 3.32: Maxentový 6-gramový model – rozšířený slovní druh + pád

Zlepšení je znatelné (obrázky 3.31, 3.32) a podobá se modelům s rozšířeným slovním druhem a všemi morfologickými značkami. Výsledky jsou nejpříznivější v rámci modelů se značkami jedné morfologické kategorie s rozšířeným slovním druhem.



Obrázek 3.33: Porovnání modelů – rozšířený slovní druh + pád

Maxentové modely dopadly o něco lépe a zvláště hypotézy hodnocené plynulostí 5 posunuly v perplexitě níže, což je správně. Nicméně proložená přímka je strmější u standardních n-gramů (obrázek 3.33). Výpočetní nároky se ale výrazně liší – 34 sekund v případě standardních n-gramových modelů oproti 27 minutám v případě modelů maxentových.

3.10 Shrnutí

Souhrný pohled nabízí tabulka 3.3. Tabulka uvádí pro od každý model Pearsonův korelační koeficient[8], Spearmanův korelační koeficient[9] a čas potřebný k natrénování. Zkratka RSD v tabulce označuje rozšířený slovní druh.

Model	Pearsonův koeficient		Spearmanův koeficient		Čas trénování	
	standardní	maxentový	standardní	maxentový	standardní	maxentový
Slova	0.05	0.05	-0.004	-0.01	3 min	12 hod
RSD + všechny značky	**-0.03**	0.04	-0.20	-0.22	72 s	8 hod
RSD	0.05	0.03	-0.20	-0.21	22 s	14 min
Rod	0.15	0.15	0.16	0.16	3 s	4 s
Rod stejný s předchozím	0.15	0.14	0.21	0.21	3 s	3 s
RSD + rod	0.04	0.03	-0.19	-0.20	36 s	27 min
Číslo	0.07	0.10	0.11	0.11	3 s	3 s
Osoba + číslo	0.06	0.09	0.67	0.68	4 s	11 s
RSD + číslo	0.03	0.02	-0.19	-0.21	33 s	24 min
Pád	0.15	0.15	0.15	0.15	4 s	4 s
RSD + pád	0.04	0.02	-0.23	**-0.24**	34 s	27 min

Tabulka 3.3: Shrnutí výsledků modelů s morfologickými značkami

U obou korelačních koeficientů bychom rádi dosáhli záporné hodnoty a v nejlepším případě blížící se -1. Pearsonův korelační koeficient má jedinou zápornou hodnotu u standardního n-gramového modelu trénovaného na rozšířeném slovním druhu a všech morfologických značkách. Hodnota je ale malá a nemůžeme z ní přímo usuzovat existenci lineární závislosti mezi perplexitou a plynulostí.

Spearmanův koeficient má záporných hodnot už více. Jedná se o hodnoty u modelů s rozšířeným slovním druhem. Hodnocení Spearmanovým koeficientem víceméně koresponduje s našimi úsudky z grafického znázornění. Jako nejlepší vyšel maxentový model s rozšířeným slovním druhem a pádem.

Je zajímavé, že maxentové modely v rámci hodnocení Pearsonovým korelačním koeficientem nepřinášely takřka žádné zlepšení, dokonce byly naopak v některých případech horší. U Spearmanova koeficientu však zlepšení u úspěšných modelů (myšleno modelů se záporným Spearmanovým koeficientem) nastalo vždy, byť jen minimální.

Na základě výpočetních nároků jsou na tom ale maxentové modely často až mnohonásobně hůře, přičemž jejich přínos pro naše experimenty nebyl dle naměřených výsledků (jak byly uvedeny v tabulce 3.3) nijak velký.

[8]Udává vztah mezi dvěma veličinami. Nabývá hodnot $< -1, 1 >$, přičemž 1 značí závislost přímou a -1 závislost nepřímou. Hodnota 0 indikuje, že vztah mezi veličinami nelze vyjádřit lineární funkcí.

[9]Spearmanův korelační koeficient nabývá hodnot jako Pearsonův, Spearmanův ale vyjadřuje, zda vztah mezi veličinami lze vyjádřit monotonní funkcí.

4. Modely s vlastní množinou rysů

Problémy s německou gramatikou jsme se prozatím snažili řešit prostým zhuštěním dat pro n-gramové modely nahrazením slov morfologickými značkami. Modely s rozšířeným slovním druhem a morfologickou analýzou dopadly sice lépe než běžné modely trénované na slovech, přesto zlepšení není nijak výrazné. V následující kapitole se proto pokusíme upustit od n-gramů a postihnout gramatiku z jiné stránky – vlastní množinou rysů.

4.1 Zdrojová data

Pro následující experimenty používáme stejná data s ručně hodnocenou plynulostí jako v předchozí kapitole. Zde jsme je rozdělili na dva díly. Polovina tj. 1045 hodnocení překladových hypotéz se použije jako vývojová sada a druhá polovina jako sada testovací. Hypotézy byly rozděleny s ohledem na hodnocení plynulosti tak, aby vývojová i testovací množina vět obsahovala stejný počet hypotéz hodnocených plynulostí 1, 2, .., 5 (až na liché počty hypotéz některých plynulostí). Následující tabulka 4.1 ukazuje přesné počty hypotéz a jejich rozdělení:

Plynulost	**Celkem hypotéz**	**Vývojová sada**	**Testovací sada**
1	150	75	75
2	445	222	223
3	932	466	466
4	387	194	193
5	176	88	88
CELKEM	**2090**	**1045**	**1045**

Tabulka 4.1: Rozdělení hodnocení hypotéz na vývojová a testovací data

Vzhledem k tomu, že budou rysy vycházet z německé gramatiky, budeme často potřebovat znát hranice klauzí dané věty. Určit klauze z hypotézy, která není gramaticky správně, je však obtížné – parser je v takovém případě zmatený a neudělá větný rozbor správně. Na základě toho používáme klauze identifikované z výchozího anglického textu, který je gramaticky správně, a na německou stranu je pak převádíme pomocí zarovnání na úrovni slov.

4.2 Implementované nástroje

Abychom mohli následující experimenty provést, bylo zapotřebí naimplementovat několik nástrojů. Jako první potřebujeme nástroj pro projekci anglických klauzí pomocí zarovnání slov na německou stranu. Pro tyto účely jsme vytvořili nástroj

Klauze. Díky identifikovaným německým klauzím jsme se mohli zaměřit na programování hledání chyb v hypotézách, k tomu slouží nástroj **Chyby**. Na základě identifikovaných chyb už stačí jen tyto hodnoty použít jako rysy pro natrénování modelů. Zmínili jsme se však, že zkusíme predikovat plynulost i jenom na základě mediánů. Vzniknul proto nástroj na trénování a testování mediánových modelů – **Klasifikátor**. V následující sekci si všechny nástroje popíšeme.

4.2.1 Nástroj Klauze

K dispozici pro převod anglických klauzí na německé máme vždy identifikované anglické klauze a zarovnání na úrovni slov. Ze zarovnání bereme jen ty nejvíce jisté shody (intersection tj. průnik dvou směrů slovního zarovnání GIZA++, více viz [ZDROJ]), neboť nám nejde o kompletní zarovnání, ale jen o hranice klauzí.

Data vypadají následovně:

Příklad anglické věty s identifikovanými klauzemi:

I|1 think|1 this|2 is|2 a|2 mistake|2 which|3 must|3 be|3 set|3 right|3 .|

Ke stejné větě příslušející zarovnání:

I think this is a mistake which must be set right .

ich denke , dies ist ein fehler , der zu recht setzen .

`0-0 1-1 2-3 3-4 4-5 5-6 6-8 7-9 9-11 10-10 11-12`

U každého slova anglické věty vidíme číslo klauze, do které patří – tedy např. slovo *mistake* patří do druhé klauze. Zarovnání je tvořeno páry odpovídajících si slov číslovaných od nuly. Z páru 4-5 vidíme, že páté slovo anglické věty (*a*) odpovídá šestému slovu (*ein*) věty německé.

Hranice klauzí budeme určovat následovně:

- Půjdeme po anglické větě s identifikovanými klauzemi zleva.
- Pro každé slovo se podíváme, zda je k dispozici zarovnání. Pokud ano, pak slovo přidáme do příslušné německé klauze – ovšem jenom za předpokladu, že jsme toto slovo už nepoužili. V případě, že zarovnání nemáme, pokračujeme dalším slovem.
- Při přidávání do klauzí si všímáme, zda číslo klauze není 0. Nulu dostávají obvykle spojky oddělující dvě věty. Pokud na takové slovo narazíme, nepřidáváme ho do nulté klauze, ale zapamatujeme si ho a přidáme ho do první následující klauze, neboť chceme aby nám německé vedlejší věty začínaly podřadící spojkou, jinak bychom je nemohli vyhodnotit jako vedlejší.
- Po projití celé anglické věty máme náznak německých klauzí. Každou klauzi nejprve setřídíme podle indexů slov, které obsahují – tím zajistíme, že se neporuší pořádek slov německé věty a zároveň dostaneme minimální a maximální index slova v klauzi, tedy jakýsi návrh na hranice.

- Podle navržených hranic projdeme všechny klauze a kontrolujeme, zda se některé nepřekrývají. V případě, že jsou hranice uvnitř hranic klauze jiné, jedná se o vloženou větu a hranice jim prozatím necháme beze změny. Může se ale stát, že se dvě klauze překrývají, aniž by jedna byla vložena do druhé. V takovém případě provedeme zarovnání zleva tj. větě, která začíná více vpravo upravíme počáteční hranici, kterou posuneme až za konec klauze, jež byla vlevo.

- Zarovnání neobsahuje vždy nutně všechna slova, a proto mohou vznikat mezi klauzemi mezery. Cílem je však identifikovat hranice, které pokryjí celou větu. Opět se proto přikloníme k zarovnání zleva a mezery přiřkneme klauzi, která stojí více vlevo. Tento postup se osvědčil na vývojových datech.

  Například pro výše uvedenou větu nebyla zarovnána čárka mezi druhou a třetí klauzí. Díky tomu nám zde vznikla mezera. Čárku dle našeho pravidla přiřadíme druhé klauzi, neboť stojí více vlevo.

Výstupní formát je pak ve formě hranicí klauzí, které jsou reprezentovány pomocí indexů slov. Začáteční hranice je od koncové oddělena pomlčkou a mezi klauzemi je znak |. V případě vložené věty může mít jedna klauze více začátečních i koncových hranic – ty jsou potom odděleny znakem `+`.

Příklad výstupu:
`0-8+13-20|9-12|21-24`

4.2.2 Nástroj Chyby

Pro spuštění nástroje Chyby potřebujeme hranice klauzí německých vět z právě popsaného nástroje Klauze a hypotézy ohodnocené plynulostí.

Soubor s hypotézami by měl být ve formátu `plynulost | hypotéza`. Hypotézám se nejprve odebere plynulost a pošlou se parseru ParZu. S výstupem už začne pracovat nástroj Chyby následujícím způsobem:

- Načteme do paměti slova a k nim morfologickou analýzu z parseru.

- Na základě hranicí klauzí postupně sestavujeme ze slov dané klauze.

- V klauzích se na základě morfologické analýzy a implementovaných pravidel začnou vyhledávat chyby. Klauze délky 1 jsou přeskočeny, neboť by se v nich našly chyby hlásicí nepřítomnost podmětu, slovesa apod. Výsledná věta by tak dostala zbytečně horší skóre. Jednoslovné klauze přitom mohly vzniknout jenom špatnou identifikací a nemusejí se zakládat na skutečné délce klauzí. Jednotlivá pravidla budou blíže popsána v sekci Vlastní rysy, kde proběhne i jejich vyhodnocení na trénovacích datech.

- Z každé klauze získáme pravdivostní hodnoty o jednotlivých pravidlech. Celá věta je pak součtem všech klauzí – tj. byla-li hodnota pravidla `true`, přičteme jedničku.

- Na závěr jsou všechny hodnoty pravidel vypsány na výstup.

K výstupu se vrátí odebraná plynulost a vznikne formát ihned použitelný k natrénování maxentového i mediánového modelu.

Výstupní formát bude např. následující:

```
1  chybi_vfin:2  pp_bez_av:3  chybi_infszu:1
4  chybi_vfin:0  pp_bez_av:0  chybi_infszu:1
2  chybi_vfin:1  pp_bez_av:2  chybi_infszu:0
```

Jak vidíme z příkladu, druhý řádek nám říká, že hypotéza hodnocená plynulostí 4 má hodnotu rysu `chybi_vfin` 0, `pp_bez_av` 0 a `chybi_infszu` 1.

4.2.3 Nástroj Klasifikátor

Pokud se nám podaří najít takové rysy, které budou vhodně korelovat s plynulostí, mohli bychom být schopni predikovat plynulost právě jen na základě mediánů hodnot rysů jednotlivých plynulostí vypočtených z trénovacích dat. Trénovací data by měla být pro optimální funkčnost seřazena podle hodnocení plynulosti od nejhorší po nejlepší.

Nástroj Klasifikátor funguje jednoduchým způsobem. Začneme načítat všechny hodnoty rysů z trénovacích dat a hned je rozdělujeme podle plynulosti dané hypotézy. Poté najdeme medián pro každý rys a každou plynulost. Mediány zapíšeme do souboru jako model použitelný pro predikci.

Predikce si zpětně pro každý rys a každou plynulost načte medián. Hodnoty mezi mediány jsou rozděleny na polovinu mezi dvě nejbližší hodnoty mediánů, přičemž v případě lichého počtu bude přidělena horší plynulosti jedna hodnota navíc. Pokud mají dvě plynulosti stejný medián, zvýšíme hodnotu odpovídající horší plynulosti o jedna. Právě kvůli tomu je vhodné mít trénovací data seřazena podle plynulostí od nejhorších po nejlepší. Potom už jenom na základě hodnoty mediánu predikujeme plynulost. U modelů s více rysy navrhne každý z rysů jednu plynulost, návrhy jsou potom zprůměrovány. Při počítání průměru používáme celočíselného dělení, hodnoty jsou proto zaokrouhleny dolů směrem k horší hodnotě plynulosti.

4.3 Vlastní rysy

Vlastní množina rysů sestává především z gramatických jevů, které n-gramy nemají možnost zachytit. Jedná se hlavně o tvorbu větného rámce. Mimo to ale budeme zkoumat, zda se ve větě třeba nevyskytuje více určitých sloves nebo naopak sloveso úplně chybí.

Celkem jsme navrhli a implementovali následujících 17 rysů, u kterých vysvětlíme, jak fungují a co kontrolují, neboť ačkoliv jejich názvy intuitivně funkci napovídají, může být omezena jen na některé případy.

1. `chybi_infszu` – kontroluje, zda byla klauze uvozena spojkou vyžadující

infinitiv s zu (rozšířený slovní druh *KOUI*[1]), typicky se jedná o zkrácené vedlejší věty, a sleduje, zda ve větě takový infinitiv byl

2. `chybi_podmet` – označuje skutečnost, že se v klauzi nevyskytlo slovo v prvním pádě
3. `chybi_vfin` – v klauzi chybí určité sloveso
4. `chybi_sum` – sčítá hodnoty všech rysů typu `chybi_*`
5. `inf_po_vm_neni_na_konci` – kontroluje, zda se za infinitivem po modálním slovese nevyskytlo ještě další slovo s výjimkou sloves
6. `infszu_neni_na_konci` – byla-li spojka vyžadující infinitiv s zu a zároveň se za infinitivem ještě vyskytlo další slovo s výjimkou sloves
7. `vv_sloveso_neni_na_konci` – po podřadících spojkách (*KOUS*, *PRELS*, *PRELAT*) kontroluje, zda po určitém slovesu nebylo ještě další slovo s výjimkou sloves
8. `pp_neni_na_konci` – pokud věta obsahovala pomocné sloveso, očekáváme příčestí minulé a kontrolujeme proto, zda se za ním ještě nevyskytlo další slovo mimo sloves
9. `neni_na_konci_sum` – sčítá hodnoty všech rysů typu `*_neni_na_konci`
10. `pp_bez_av` – kontroluje přítomnost a pozici pomocného slovesa před příčestím minulým – neplatí ve vedlejších větách a po souřadících spojkách, které by mohly oddělovat dvě věty se dvěma příčestími minulými, ale pomocným slovesem jen v první z nich (*např.: Ich habe gekocht und gelernt.*)
11. `neshoda_podmet_prisudek` – sledujeme výskyt prvního slova v nominativu nebo prvního slovesa, od těchto prvních výskytů si zapamatujeme číslo a osobu (u podstatných jmen ručně nastavíme, že se jedná o třetí osobu), pokud se nenajde shoda v čísle a osobě mezi prvním nalezeným nominativem a slovesem, pak daná klauze dostane tento rys
12. `vice_osob` – funguje stejně jako předchozí, jenom s jiným vyhodnocením – tj. tehdy, když po první nalezené osobě nalezneme ještě další (samozřejmě v prvním pádě)
13. `vice_vfin` – indikuje výskyt více určitých sloves v jedné klauzi
14. `vice_sum` – sčítá hodnoty všech rysů typu `vice_*`
15. `sum` – sčítá hodnoty všech předchozích rysů
16. `root` – projde větný rozbor z výstupu ParZu a spočítá počet kořenů, je-li totiž věta gramaticky správně, nalezneme kořen pouze jeden, v opačném případě je jich více a představují pomyslný počet chyb ve větě

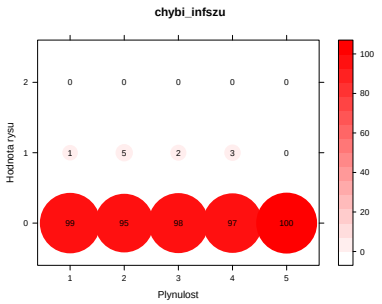
[1]ParZu používá pro označení rozšířeného slovního druhu Stuttgart/Tübinger Tagsets ZDROJ http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/stts.asc

17. `sumr` – jako `sum`, ale včetně `root`

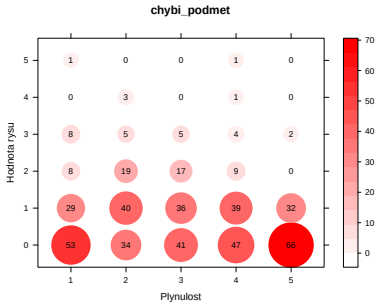
Pomocí programu Chyby, o kterém jsme se již zmínili, jsme změřili na vývojových datech korelaci každého rysu s ručně hodnocenou plynulostí. Každý rys kromě rysů součtových a rysu `root` jsou určovány pro každou klauzi zvlášť a mohou v ní vždy dostat jen hodnotu `true`/`false`. Rysy celé věty jsou pak součtem hodnot rysů ze všech klauzí – přičteme vždy jedničku, když daný rys nabyl v klauzi hodnoty `true`.

Znázornění provedeme pomocí tzv. bublinových grafů. V místě střetu plynulosti a dané hodnoty rysu se vždy vykreslí kruh (bublina) velká tak, kolik procent hodnot dané plynulosti dostalo tuto hodnotu rysu. Bubliny jsou navíc barevně odstupňované podle počtu procent – čím světlejší, tím menší výskyt. Uvnitř každé bubliny je navíc vypsáno toto procentuální vyjádření.

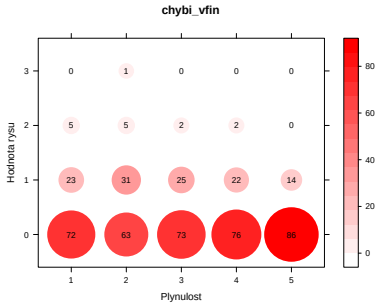
4.3.1 Rysy typu chybi_*



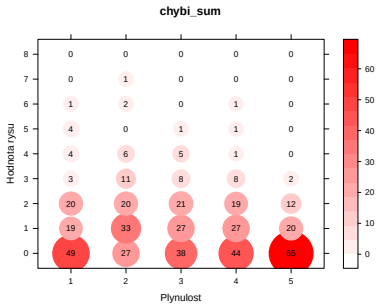
Obrázek 4.1: Korelace hodnoty rysu chybi_infszu a plynulosti



Obrázek 4.2: Korelace hodnoty rysu chybi_podmet a plynulosti



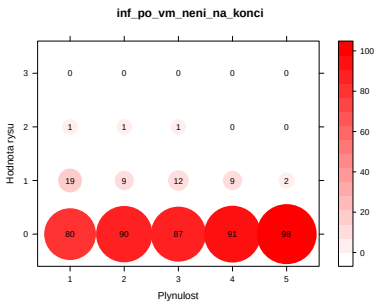
Obrázek 4.3: Korelace hodnoty rysu chybi_vfin a plynulosti



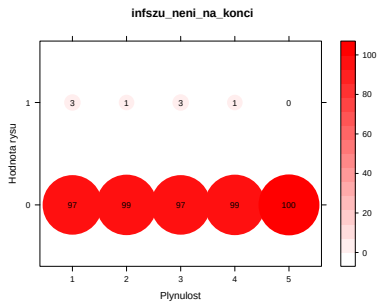
Obrázek 4.4: Korelace součtového rysu chybi_sum a plynulosti

Jednotlivé rysy samostatně neukázaly souvislost s plynulostí (obrázky 4.1, 4.2, 4.3), neboť hodnoty se pohybují hlavně okolo nuly. V součtu je u hypotéz s plynulostí nejvíce hodnot na nule, což je správně, neboť tyto hypotézy by měly být úplně bez chyb (obrázek 4.4). Hodnoty nad nulou jsou způsobené jednak chybami v hranicích klauzí, protože se v nich spoléháme na identifikaci klauzí anglických a na zarovnání slov. Jednak také chybami při morfologické analýze z ParZu a samozřejmě také speciálními případy, se kterými náš program na vyhledávání hodnot rysů nepočítá.

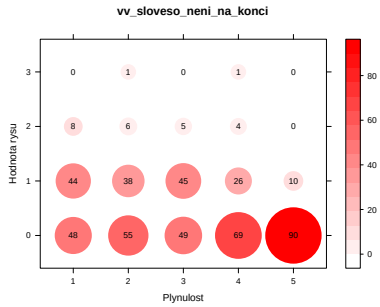
4.3.2 Rysy typu *_neni_na_konci



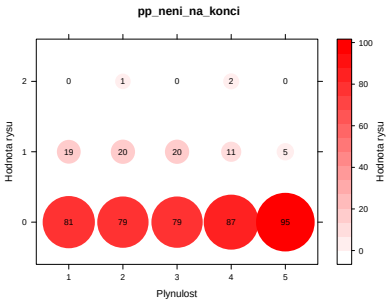
Obrázek 4.5: Korelace hodnoty rysu inf_po_vm_neni_na_konci a plynulosti



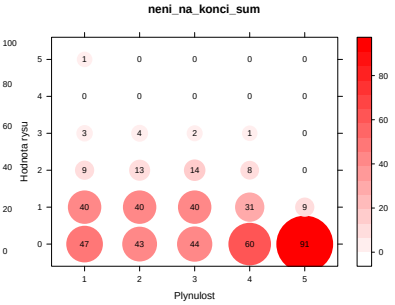
Obrázek 4.6: Korelace hodnoty rysu infszu_neni_na_konci a plynulosti



Obrázek 4.7: Korelace hodnoty rysu vv_sloveso_neni_na_konci a plynulosti



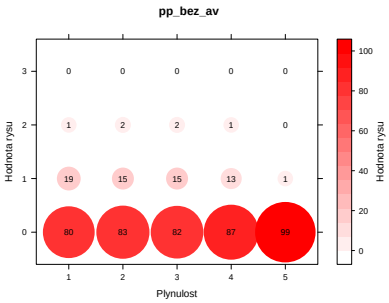
Obrázek 4.8: Korelace hodnoty rysu pp_neni_na_konci a plynulosti



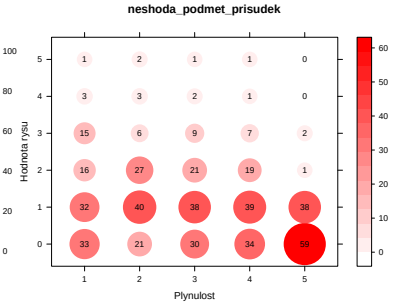
Obrázek 4.9: Korelace součtového rysu neni_na_konci_sum a plynulosti

Zde je situace podobná jako u předchozí skupiny rysů. Jednotlivé rysy samostatně (obrázky 4.5, 4.6, 4.7, 4.8) mají hodnoty nejčastěji okolo nuly. U součtového rysu (obrázek 4.9) alespoň hypotézy s hodnocením plynulosti 5 dostávaly oproti ostatním plynulostem častěji nulu.

4.3.3 Rysy pp_bez_av a neshoda_podmet_prisudek



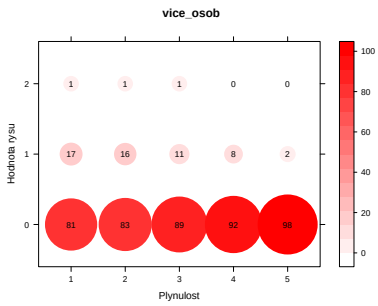
Obrázek 4.10: Korelace hodnoty rysu pp_bez_av a plynulosti



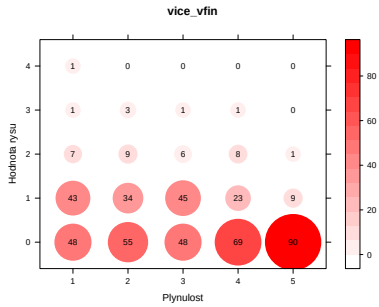
Obrázek 4.11: Korelace hodnoty rysu neshoda_podmet_prisudek a plynulosti

Oba rysy opět nevykazují samostatně souvislost s plynulostí (obrázky 4.10, 4.11).

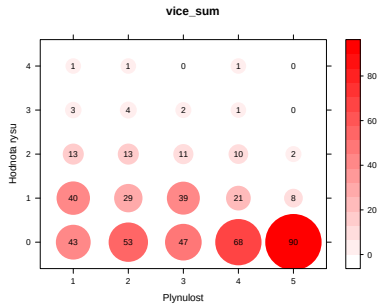
4.3.4 Rysy typu vice_*



Obrázek 4.12: Korelace hodnoty rysu vice_osob a plynulosti



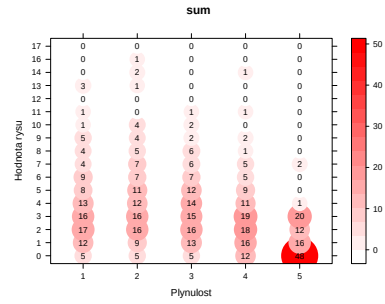
Obrázek 4.13: Korelace hodnoty rysu vice_vfin a plynulosti



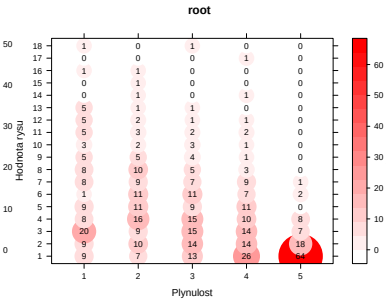
Obrázek 4.14: Korelace součtového rysu vice_sum a plynulosti

Shodně dopadly i rysy typu `vice_*`. Nejčastější hodnoty rysu `vice_osob` jsou nulové (obrázek 4.12). U rysu `vice_vfin` kolísají hlavně mezi nulou a jedničkou (obrázek 4.13), stejně jako i u součtového rysu (obrázek 4.14).

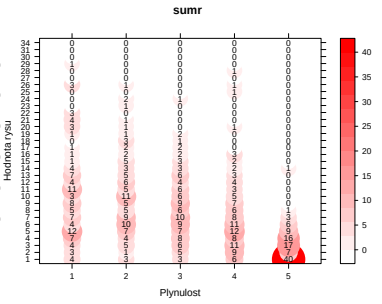
4.3.5 Rysy sum a root



Obrázek 4.15: Korelace hodnoty rysu sum a plynulosti

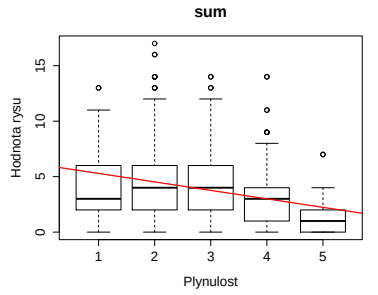


Obrázek 4.16: Korelace hodnoty rysu root a plynulosti

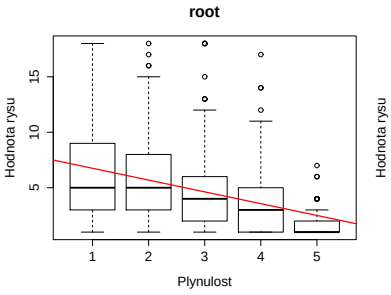


Obrázek 4.17: Korelace hodnoty rysu sumr a plynulosti

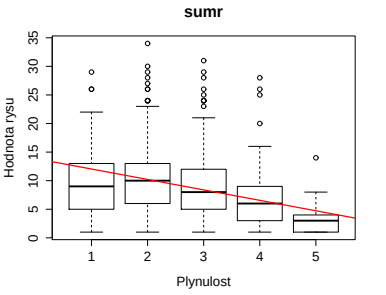
Vzhledem k tomu, že tyto rysy mají větší rozsah hodnot než rysy předchozí a bublinový graf zde situaci spíše znepřehledňuje, vykreslíme pro ně standardní boxplot. Všemi hodnotami ještě proložíme přímku.



Obrázek 4.18: Korelace hodnoty rysu sum a plynulosti



Obrázek 4.19: Korelace hodnoty rysu root a plynulosti



Obrázek 4.20: Korelace hodnoty rysu sumr a plynulosti

Všechny tři rysy `sum`, `root` a jejich součet `sumr` vykazují souvislost svých hodnot s plynulostí (obrázky 4.18, 4.19, 4.20). Proložené přímky mají klesavou tendenci,

nejvýraznější u rysu `root`. Na základě toho zkusíme jednak už zmíněnou predikci dle hodnot mediánů, ale také predikci podle lineární regrese. Nejprve je však potřeba podívat se na přesnost určení hodnot rysů jako takových.

4.3.6 Přesnost určení rysů

Jak jsme již zmínili, určování rysů se musí spoléhat na další výstupy nástrojů, které taktéž nepracují bezchybně. Abychom alespoň orientačně věděli, jak přesné určení rysů je, analyzovali jsme ručně 15 náhodných vět a kontrolovali, zda se hodnota rysu shoduje se skutečností. V rámci ruční kontroly jsme přišli na to, že spoustu chyb v určení zapříčiňuje špatná identifikace slovního druhu parserem ParZu. K chybě dochází především z důvodu, že německé hypotézy jsou psané pouze malými písmeny. V němčině ale často právě velké písmeno rozhodne o tom, zda se jedná o sloveso nebo podstatné jméno, *např. zahlen* × *Zahlen.*

Příklad ruční analýzy:

in vielen ländern , aber die politischen parteien sich schwer vorstellen , daß solche debatten auch .

ok	chybi_infszu:0	-1	chybi_podmet:1
	chybi_sum:1	+1	chybi_vfin:0
ok	inf_po_vm_neni_na_konci:0	ok	infszu_neni_na_konci:0
	neni_na_konci_sum:0	ok	neshoda_podmet_prisudek:1
ok	pp_bez_av:0	ok	pp_neni_na_konci:0
	root:3		sum:5
ok	vice_osob:0		vice_sum:0
ok	vice_vfin:0	ok	vv_sloveso_neni_na_konci:0

Ve sloupci nalevo od názvu rysu je uvedeno stanovisko – ok označuje souhlas, kladná hodnota udává, o kolik měla být hodnota rysu vyšší, záporná udává opak tj. o kolik měla být hodnota nižší. Součtové rysy hodnoceny nebyly, neboť vychází z ostatních. Stejně jako rys `root` nebyl hodnocen, neboť ten stanovujeme pouze na základě výstupu z ParZu.

Stanovení, zda je některý rys správně či špatně, je někdy sporné, neboť mnoho hypotéz nedává smysl a spoustu věcí tak musíme jen odhadovat. V této hypotéze například nevíme, zda se jedná o dvě nebo tři klauze, proto jsme rys `chybi_vfin` mohli ohodnotit jako +1 nebo +2. Zde jsme zvolili +1, pokud bychom ohodnotili jako +2, pak bychom zase u `chybi_podmet` museli namísto -1 zvolit ok.

Vyhodnocení jsme poté stanovili jako precision a recall (viz. ZDROJ) tak, že jsme vždy vzali minimum z vyhodnocení každého rysu, tato minima sečetli a vydělili součtem hodnot, které byly stanoveny programem, v případě precision, a součtem hodnot, které měly být, v případě recall. Vyšly následující hodnoty: 41.1 % precision a 76.92 % recall. Kompletní přehled výsledků je na přiloženém DVD.

Nutno však podotknout, že obě metriky předpokládají, že pokud se ruční hodnocení shodlo s hodnocením od nástroje Chyby, pak se jednalo o ten samý jev na shodném úseku hypotézy. Vzhledem k tomu, že jsme nepozorovali, kdy a na

kterou část hypotézy program rys nahlásil jako pravdivý, ale pouze už konkrétní výsledky na celé hypotéze, nemusí být vždy tento předpoklad splněn.

4.4 Princip experimentů

Princip experimentů bude následující:

- identifikovat německé klauze v hypotézách na základě anglických klauzí a zarovnání anglických a německých slov, jako trénovací data použijeme vývojovou sadu ohodnocených hypotéz
- provést morfologickou analýzu a pokusit se o větný rozbor hypotéz
- uplatnit naše pravidla pro hledání gramatických chyb a použít je jako rysy
- natrénovat maxentový a mediánový[2] model s těmito rysy
- provést stejný postup na testovacích datech a modely otestovat
- u rysů, kde byla zjištěna znázorněnou korelací závislost s plynulostí tj. `root`, `sum` a `sumr`, vyzkoušíme také spočítat lineární regresi v závislosti na hodnotě rysu v trénovacích datech a predikovat podle ní plynulost

U experimentů využijeme tří implementovaných nástrojů, které jsme popsali tj. Klauze pro identifikaci německých klauzí, Chyby pro vyhledání hodnot rysů a Klasifikátor pro trénování mediánových modelů.

Na morfologickou analýzu a větný rozbor použijeme opět parser ParZu. Tentokrát ale využijeme ještě dalších informací, které poskytuje. Zkusíme využít v náš prospěch i skutečnosti, že u gramaticky špatné věty postaví větný rozbor chybně (pro rys `root`).

Maxentové modely budeme trénovat v MaxentToolkitu(ZDROJ) od Le Zhanga. Pro určení vah rysů využijeme výchozího nastavení tj. metody LBFGS.

4.5 Způsob vyhodnocení

U obou typů modelů budeme vyhodnocovat hlavně přesnost predikce tj. procentuálně vyjádříme, v kolika případech se model trefil do správné plynulosti (níže označeno jako *přesná shoda*). Mimo to budeme ještě zkoumat, v kolika případech se plynulost navrhovaná modelem lišila od skutečnosti jen o 1 (*shoda nebo ± 1*). Vzhledem k tomu, že plynulost 3 je nejčastější, můžeme za základní (baseline) prohlásit postup, který právě hodnotu 3 přiřkne každé hypotéze. Baseline dosahuje úspěšnosti 44.59 % při vyhodnocení *přesná shoda* a dokonce 84.4 % pri vyhodnocení *shoda nebo ± 1*.

[2]Mediánovým modelem rozumíme model pro náš klasifikátor hodnotící jen na základě mediánů vypočtených z hodnot rysů v trénovacích datech.

Znázornění hodnot rysů jednotlivých plynulostí ukázalo, že hypotézy hodnocené plynulostí 5, dostávají správně nízké (nejlépe nulové) hodnoty rysů. Tím bychom mohli alespoň odlišit nejlepší (nejplynulejší) hypotézy od těch méně kvalitních. Při vyhodnocení si budeme proto i všímat úspěšnosti modelu predikovat právě plynulost 5. Hypotéz hodnocených touto plynulostí je v trénovacích i testovacích datech shodně 88 z celkového počtu 1045 hypotéz. Pokud bychom tedy tipovali samé 5, pak by úspěšnost byla pouhých 8.42 %. Naším cílem proto bude se nad tuto hranici dostat.

4.6 Modely se všemi rysy

Jako první zkusíme natrénovat modely se všemi šestnácti rysy.

		Maxentový model	**Mediánový model**
přesná shoda		41.05 %	29.86 %
shoda nebo ±1		80.10 %	77.61 %
shoda v plynulosti 5		0 %	34.09 %
počty plynulostí	1	0	0
	2	250	1
	3	795	299
	4	0	672
	5	0	73

Tabulka 4.2: Naměřené hodnoty modelů se všemi rysy

Přesnost obou modelů je nižší než, kdybychom slepě tipovali samé trojky, tedy baseline. Pro zajímavost ještě navíc sledujeme, které hodnoty plynulosti model kolikrát nabídl (tabulka 4.2). Shoda u hypotéz s hodnocením 5 dopadla u maxentového modelu velice špatně, neboť ji nanabídnul ani jednou a dostal tak 0 %. Mediánový model je na tom lépe a podařilo se mu správně predikovat 34.09 % nejplynulejších hypotéz.

4.7 Modely se součtovými rysy

Při znázorňování korelace rysů a plynulosti vždy vyšly o něco lépe součtové rysy. Konkrétně se jedná o rysy `chybi_sum`, `neni_na_konci_sum`, `sum`, `sumr` `vice_sum`. Zkusíme proto modely natrénovat právě jenom na nich.

Maxentový model se dotýká hranice baseline, bohužel ji ale nepřekonává. Mediánový model je na tom s přesností hůře, ovšem nabídnul každou plynulost alespoň jednou (tabulka 4.3). Shoda v plynulosti 5 dopadla u maxentového modelu opět na nula procent. Mediánový si naopak polepšil na 46.59 %.

		Maxentový model	Mediánový model
přesná shoda		40.96 %	34.07 %
shoda nebo ±1		80.10 %	80.38 %
shoda v plynulosti 5		0 %	46.59 %
počty plynulostí	1	0	2
	2	231	139
	3	814	359
	4	0	407
	5	0	138

Tabulka 4.3: Naměřené hodnoty modelů se všemi součtovými rysy

4.7.1 S rysem root

Mimo součtové rysy, co se korelace týče, dopadnul dobře i rys `root`. Zkusíme jej proto přidat k součtovým rysům.

		Maxentový model	Mediánový model
přesná shoda		41.73 %	34.83 %
shoda nebo ±1		81.15 %	84.11 %
shoda v plynulosti 5		0 %	38.64 %
počty plynulostí	1	0	4
	2	148	244
	3	897	339
	4	0	368
	5	0	90

Tabulka 4.4: Naměřené hodnoty modelů se součtovými rysy a rysem root

Přidání se projevilo pozitivně na obou typech modelů. Maxentový si ve shodě nebo ±1 polepšil o 1.05 %, mediánový o 3.73 % (tabulka 4.4). Mediánový model znovu narozdíl od maxentového nabídnul všechny plynulosti. Shoda v hypotézách hodnocených plynulostí 5 dopadla hůře, neboť mediánovému modelu klesla o 7.95 %.

4.8 Modely s rysem root

Rys `root` vypadal v korelaci s plynulostí slibně. Se součtovými rysy přílišné zlepšení nepřinesl, zkusíme jej proto použít zcela samostatně a poprvé také zkusíme predikovat za pomoci lineární regrese.

Zde došlo u maxentového modelu k popisovanému slepému tipování a jen díky tomu dosáhnul na baseline. Mediánový model nabídnul znovu celou škálu plynulostí (tabulka 4.5), ovšem jeho výsledky shody jsou nízké. Lineární regresí jsme se dostali lehce nad baseline, což je určitě dobrá zpráva. V predikci plynulostí 5 ale maxentový model i lineární regrese selhaly a nenabídly ji ani jednou, proto mají shodu 0 %. Naopak mediánovému modelu se podařilo dostat až na 54.55 %.

		Maxentový model	Mediánový model	Lineární regrese
přesná shoda		44.59 %	23.54 %	44.69 %
shoda nebo ±1		84.40 %	62.30 %	85.36 %
shoda v plynulosti 5		0 %	54.55 %	0 %
počty plynulostí	1	0	316	2
	2	0	99	89
	3	1045	132	954
	4	0	310	0
	5	0	188	0

Tabulka 4.5: Naměřené hodnoty modelů s rysem root

4.9 Modely s rysem sum

Vedle na pohled dobře korelujícího rysu `root`, dobře dopadl i součtový rys `sum`, který je součtem ostatních rysů kromě těch součtových a rysu `root`. Zkusíme proto znovu jako v případě rysu `root`, natrénovat modely na jediném rysu `sum`. Znovu využijeme lineární regrese.

		Maxentový model	Mediánový model	Lineární regrese
přesná shoda		40.96 %	21.53 %	44.98 %
shoda nebo ±1		77.70 %	66.22 %	84.59 %
shoda v plynulosti 5		0 %	57.95 %	0 %
počty plynulostí	1	115	137	0
	2	0	205	32
	3	930	97	1013
	4	0	361	0
	5	0	245	0

Tabulka 4.6: Naměřené hodnoty modelů s rysem sum

Zde je na výsledcích zajímavé rozložení plynulostí, které tipoval maxentový model. Netipoval totiž ani jednou plynulost 2, přitom tipoval plynulosti 1 a 3. K tomu jevu u žádného jiného modelu nedošlo. Mediánový model opět nabídl všechny plynulosti, s přesností je na tom ale špatně (tabulka 4.6). Zato lineární regresí jsme znovu lehce nad baseline. Shoda v plynulosti 5 se povedla jen mediánovému modelu – 57.95 %.

4.9.1 S rysem root

Jelikož rysy `root` a `sum` se zdály být dvěma ze tří nejúspěšnějších v rámci korelace s plynulostí, zkusíme natrénovat modely na obou najednou.

Přidání druhého rysu ale u maxentového modelu znamenalo slepé tipování pouze do plynulostí 3, proto dosáhl na baseline. U mediánových modelů se zvýšila shoda nebo ±1 oproti samostatným modelům s rysy `root` a `sum` zhruba o 10 % (tabulka 4.7). Shoda v plynulostech 5 naopak poklesla na 39.77 %.

		Maxentový model	Mediánový model
přesná shoda		44.59 %	27.56 %
shoda nebo ±1		84.40 %	74.26 %
shoda v plynulosti 5		0 %	39.77 %
počty plynulostí	1	0	185
	2	0	245
	3	1045	195
	4	0	312
	5	0	108

Tabulka 4.7: Naměřené hodnoty modelů se rysy sum a root

4.10 Modely s rysem sumr

Vedle na pohled dobře korelujících rysů `root` a `sum`, dobře dopadl i součtový rys `sumr`, který rys `root` a `sum` obsahuje v sobě započítaný. Zkusíme znovu jako v případě rysu `root` a `sum`, natrénovat modely na jediném rysu `sumr`. Znovu využijeme lineární regrese.

		Maxentový model	Mediánový model	Lineární regrese
přesná shoda		44.59 %	28.90 %	44.88 %
shoda nebo ±1		84.40 %	73.21 %	84.98 %
shoda v plynulosti 5		0 %	77.27 %	0 %
počty plynulostí	1	0	62	1
	2	0	341	79
	3	1045	155	965
	4	0	195	0
	5	0	292	0

Tabulka 4.8: Naměřené hodnoty modelů s rysem sumr

Od maxentového modelu jsme dostali znovu stejnou odpověď jako v případě rysu `root` – tedy slepé tipování jen plynulostí 3 (tabulka 4.8). Mediánový model znovu nabídnul od všech plynulostí minimálně jedno hodnocení a oproti rysu `root` si polepšil o 5.36 % v přesné shodě, ve shodě nebo ± ale propadl takřka o 18 % níže. Podobně dopadl i ve srovnání s rysem `sum`. Lineární regrese se pohybuje lehce nad baseline. Zajímavá je ale i hodnota shody plynulosti 5 u mediánového modelu, který dosáhnul 77.27 %.

4.11 Modely se všemi rysy kromě rysů součtových

Namísto úspěšných rysů, které vycházely ze součtu jiných, zkusíme nechat natrénovat modely na všech rysech kromě těch součtových. Může se stát, že si pro ně maxentový model najde váhy, které budou podávat lepší výsledky než námi podané součty v rysech.

		Maxentový model	Mediánový model
přesná shoda		41.06 %	23.83 %
shoda nebo ±1		80.00 %	72.72 %
shoda v plynulosti 5		0 %	34.09 %
počty plynulostí	1	0	0
	2	255	0
	3	790	115
	4	0	857
	5	0	73

Tabulka 4.9: Naměřené hodnoty modelů se všemi rysy kromě rysů součtových

Bohužel maxentový model nedosáhl ani na baseline. Mediánový je na tom ještě takřka o polovinu hůře, ale u něho lze toto očekávat, neboť jednotlivé rysy samostatně nevykazovaly korelaci s plynulostí. Ani jeden z modelů nenabízel všechny plynulosti (tabulka 4.9). Shoda v plynulosti 5 se opět podařila jen u mediánového modelu.

4.11.1 Bez rysu root

Mimo naše součtové rysy máme ještě jeden rys, který sčítá kořeny větného rozboru – rys `root`. Zkusíme tedy vynechat i ten a uvidíme, zda se situace zlepší nebo zhorší.

		Maxentový model	Mediánový model
přesná shoda		38.09 %	19.81 %
shoda nebo ±1		75.60 %	69.19 %
shoda v plynulosti 5		0 %	42.05 %
počty plynulostí	1	115	0
	2	182	0
	3	748	41
	4	0	889
	5	0	115

Tabulka 4.10: Naměřené hodnoty modelů se všemi rysy kromě součtových a kromě root

Zde se ale výsledky ve všech parametrech kromě shody v hypotézách hodnocených plynulostí 5 u mediánového modelu zhoršily (tabulka 4.10).

4.12 Shrnutí

Zde shrneme všechny naměřené hodnoty do tabulky. Ve sloupci lišící se o 1 uvedeme počet procent, pokud model vyhověl našemu stanovenému kritériu, nebo X v případě opačném.

Model	Přesná shoda		Shoda nebo ±1		Shoda v plynulosti 5	
	maxentový	mediánový	maxentový	mediánový	maxentový	mediánový
Baseline	44.59 %		84.40 %		×	
Všechny rysy	41.05 %	29.86 %	80.10 %	77.61 %	0 %	34.09 %
Součtové rysy	40.96 %	34.07 %	80.10 %	80.38 %	0 %	46.59 %
Součtové rysy a root	41.73 %	**34.83 %**	81.15 %	**84.11 %**	0 %	38.64 %
Rys root	**44.59 %**	23.54 %	**84.40 %**	62.30 %	0 %	54.55 %
Rys sum	40.96 %	21.53 %	77.70 %	66.22 %	0 %	57.95 %
Rysy root a sum	**44.59** %	27.56 %	**84.40 %**	74.26 %	0 %	39.77 %
Rys sumr	**44.59** %	28.90 %	**84.40 %**	73.21 %	0 %	**77.27 %**
Všechny rysy kromě součtových	41.06 %	23.83 %	80.00 %	72.72 %	0 %	34.09 %
Všechny rysy kromě součtových a kromě root	38.09 %	19.81 %	75.60 %	69.19 %	0 %	42.05 %

Tabulka 4.11: Shrnutí výsledků maxentových a mediánových modelů s vlastními rysy

Z tabulky 4.11 vidíme, že ani v jednom případě jsme nebyli lepší než baseline. Baseline dosáhl maxentový model ve třech případech, vždy se ale jednalo o slepé tipování, kdy všechny hypotézy ohodnotil plynulostí 3. Pokud se ale podíváme na kritérium shody u hypotéz hodnocených plynulostí 5, pak zde maxentové modely úplně propadly, neboť dostaly ve všech případech 0 %. Zato mediánovým se v tomto ohledu lišilo lépe a v případě modelu s rysem `sumr` jsme se dostali na shodu 77.27 %. Mediánové modely by proto byly spíše vhodné pro použití v případech, kdy nás zajímá právě ta nejplynulejší hypotéza.

Mimo to jsme ale zkoušeli predikovat i pomocí lineární regrese a výsledky v přesné shodě i shodě nebo ±1 dopadly nejlépe ze všech a lehce překročily baseline. Jejich výsledky shrnuje následující tabulka 4.12:

Model	Přesná shoda	Shoda nebo ±1	Shoda v plynulosti 5
Baseline	44.59 %	84.40 %	×
Rys root	44.69 %	**85.36 %**	0 %
Rys sum	**44.98 %**	84.59 %	0 %
Rys sumr	44.88 %	84.98 %	0 %

Tabulka 4.12: Shrnutí výsledků predikce pomocí lineární regrese

S predikcí pomocí lineární regrese jsme se vždy dostali nad baseline, ačkoliv zlepšení zde není nijak vysoké – 0.39 % u přesné shody a 0.96 % u shody nebo ±1. Metoda stejně jako maxentové modely neuspěla při predikci hypotéz hodnocených plynulostí 5.

Závěr

V práci byly popsány dvě série experimentů. První z nich se snažila modelovat jazyk jen na základě morfologických značek. Pro srovnání byly natrénovány i modely se slovy. Pokud jsme modelu předaly pouze značky z jedné morfologické třídy (rod, pád, číslo), dopadly hůře než modely se slovy. S přidáním rozšířeného slovního druhu došlo ale vždy ke zlepšení a modely vycházely lépe než běžné modely se slovy. Jako nejúspěšnější z hlediska Spearmanova korelačního koeficientu v korelaci perplexity a ručně hodnocené plynulosti vyšel maxentový model trénovaný na rozšířeném slovním druhu a pádu. Natrénování maxentových modelů trvalo někdy až mnohonásobně déle než standardních n-gramových, přičemž přínos maxentových modelů nebyl zvlášť výrazný. Obecně maxentové modely dostávaly nižší perplexitu, ale příliš nezlepšovaly korelaci s plynulostí.

Druhá série experimentů se zabývala modely maximální entropie a vlastními mediánovými modely. Bylo navrženo 16 vlastních rysů reprezentujících počet chyb v dané větě. Samostatně rysy nekorelovaly s plynulostí, a proto byly rozděleny do několika skupin a jejich hodnoty jsme sečetli. Součtové rysy vycházely v korelaci lépe. Jako nejúspěšnější z hlediska korelace dopadly rysy `root` a `sum`, kde první představuje počet kořenů větného rozboru z parseru Parzu a druhý součet přes všechny rysy. Na základě toho jsme zkoušeli predikovat pouze dle hodnot mediánů jednotlivých plynulostí. Experimenty jsme prováděli na různých kombinacích těchto vlastních rysů. V přenosti predikce plynulosti nedopadl bohužel žádný model dobře. Všechny byly horší než baseline. Zkusili jsme proto i sledovat, v kolika případech se modely lišily v plynulosti jen o 1 a stanovili jsme kritérium, kdy tento údaj lze brát v potaz. V závěrečném shrnutí proto uvádíme tento údaj jen u těch, které nabídly pro každou plynulost alespoň jedno hodnocení. Po tomto vyhodnocení dopadl nejlépe náš mediánový model se všemi součtovými rysy a rysem `root`. Výsledky však nejsou tak dobré, jak bychom si přáli. Úspěšnost by bylo možné zvýšit odstraněním chyb jednotlivých nástrojů, které bylo potřeba použít při předzpracování dat pro identifikaci německých klauzí. Určitého zlepšení by se mohlo dosáhnout i v případě, kdy by německé hypotézy nebyly převedené jen na malá písmena, neboť to často dělalo parseru problémy a došlo ke špatnému určení slovního druhu, díky čemuž byla následně i špatně identifikována hodnota rysu. Mimo to by bylo možné dále navrhovat a testovat jiné rysy reprezentující chybu v gramatice. Vycházíme z předpokladu, že hodnotitelé posuzují plynulost právě na základě počtu vyskytujících se chyb.

Modely z obou sérií experimentů potřebovaly data z parseru, kterému musíme předložit celou větu, aby správně určil morfologickou analýzu a větný rozbor. Jejich praktické využití by proto bylo např. při reskórování nbestlistů (n nejlepších hypotéz).

Seznam použité literatury

[1] JURAFSKY, Dan a James H. MARTIN. *Speech and language processing: an introduction to natural language processing, computational linguistics, and speech recognition*. 2nd ed. Upper Saddle River: Pearson Education, 2008. ISBN 978-0-13-187321-6.

[2] SMRŽ, Pavel. *Jazykové modelování* [online]. 2006 [cit. 2013-05-04]. Dostupné z: `http://www.fit.vutbr.cz/study/courses/SRE/public/prednasky/2010-11/01_lm/SRE_LM.pdf`

[3] FRASER, Alexander, Marion WELLER a Cahilly Fabienne CAP. *Modeling Inflection and Word-Formation in SMT* [online]. 2012 [cit. 2013-05-04]. Dostupné z: `http://eprints.pascal-network.org/archive/00009510/01/morphgen_hmm.pdf`

Seznam obrázků

Seznam tabulek

Přílohy