Západočeská univerzita v Plzni
Fakulta aplikovaných věd

Metody redukce OOV ve statistických jazykových modelech založených na třídách

Ing. Jan Hoidekr

disertační práce
k získání akademického titulu doktor
v oboru Kybernetika

Školitel: Prof. Ing. Josef Psutka CSc.
Katedra kybernetiky

Plzeň, 2012

University of West Bohemia
Faculty of Applied Sciences

Methods of OOV Reduction in Class–based Statistical Language Models

Ing. Jan Hoidekr

doctoral thesis
submitted in conformity with requirements
for the degree of Doctor of Philosophy
the field Cybernetics

Supervisor : Prof. Ing. Josef Psutka CSc.
Department of cybernetics

Plzeň, 2012

ČESTNÉ PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě disertační práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem předloženou disertační práci vypracoval samostatně s použitím odborné literatury a pramenů, jejichž úplný seznam je její součástí.

Plzeň dne ....

podpis

PODĚKOVÁNÍ

Na tomto místě bych rád poděkoval všem, kteří mi pomohli cennými radami při psaní této práce. Zvláštní dík patří kolegům z Katedry kybernetiky.

Obsah

1 Úvod **10**

2 Obecná úloha rozpoznávání řeči **12**

2.1 Formulace úlohy 13

2.2 Dekompozice na podúlohy 14

3 Cíle disertační práce **18**

3.1 Dílčí cíle práce 19

4 Statistické jazykové modely **20**

4.1 Kvalita jazykového modelu 23

4.1.1 Perplexita 23

4.1.2 Accuracy, Correctness 25

4.2 Odhady pravděpodobností jazykového modelu 26

4.2.1 Metoda maximální věrohodnosti 27

4.2.2 Add-one smoothing 27

4.2.3 Held-out estimace 28

4.2.3.1 Good-Turing discounting 29

4.2.3.2 Absolute discounting 31

4.2.4 Witten-Bell discounting 32
4.3 Kombinování odhadů 33
4.3.1 Lineární interpolace 33
4.3.2 Backoff 34
4.3.3 Maximální entropie 37
4.4 Jazykové modely s třídami 38
4.4.1 Predictive clustering 42
4.4.2 Phrase class n-gramy 42
4.5 Modelování delších vzdáleností mezi slovy 44
4.5.1 Cache modely 44
4.5.2 Trigger modely 45
4.5.3 Skip n-gramové modely 46
4.6 Velikost n-gramových modelů a její redukce 46
4.6.1 Jazykové modely založené na neuronových sítích 49
4.6.2 Jazykové modely založené na rekurzivních neuronových sítích 49

5 Jazykové modelování češtiny 52

5.1 Snižování OOV v jazykových modelech 53
5.1.1 Morfémový jazykový model 53
5.1.2 Jazykový model s třídami. 54
5.1.2.1 Automaticky odvozené třídy - POS tagging 54
5.1.2.2 Sémantické třídy 54

6 Úloha a návrh řešení 57

6.1 Soustava soudů v České republice 59
6.2 Současné metody pro rozšíření slovníku 60
6.3 Návrh řešení 61
6.4 Zdroje dat pro sestavení tříd 62
6.5 Rozdělení pravděpodobnosti ve třídách 65
6.5.1 Uniformní rozdělení pravděpodobnosti ve třídách 65
6.5.2 Rozšířené rozdělení pravděpodobnosti ve třídách 65
6.6 Velikost třídového jazykového modelu 67

7 Experimenty a výsledky 68

7.1 Popis korpusu 68
7.2 Rozpoznávání s referenčními modely 69
7.3 Analýza chyb rozpoznávání 71
7.4 Třídy slov v trénovacím textu 72
7.5 Rozpoznávání s třídovými modely 73
7.6 Rozšíření slovníku jazykových modelů 75
7.6.1 Sestavení tříd obcí a ulic 76
7.6.1.1 Uniformní rozdělení ve třídách 77
7.6.1.2 Rozdělení pravděpodobnosti ve třídách obcí 77
7.6.1.3 Rozdělení pravděpodobnosti ve třídách ulic 78
7.7 Výsledky rozpoznávání s rozšířenými třídami 80

8 Závěr 82

Seznam tabulek

4.1 Počet parametrů modelu pro slovník 20 000 slov 22
5.1 Výsledky rozpoznávání hokejového komentáře 55
5.2 Výsledky rozpoznávání jmen osob v záznamu schůze PSP ČR 55
6.1 Čtyřčlánková soustava soudů v České republice 59
7.1 Korpusy soudních rozhodnutí 69
7.2 Základní testovací sady - text 70
7.3 Základní experimenty s rozpoznáváním 71
7.4 Označené místní názvy v korpusu 72
7.5 Označené obce podle pádu 73
7.6 Označené ulice podle pádů 73
7.7 Prořezání jazykového modelu z Plzeňského kraje 75
7.8 Počet obcí a ulic v soudních krajích ČR 76
7.9 Výsledky s třídami s uniformním rozložením pravděpodobnosti . . . 77
7.10 Korelační koeficienty mezi trénovacími daty a navrženou třídou obcí . 78
7.11 Korelační koeficienty mezi trénovacími daty a navrženou třídou ulic . 79
7.12 Výsledky s třídami s navrženým rozložením pravděpodobnosti 81
7.13 Srovnání výsledků rozpoznávání - průměrné hodnoty 81

Seznam obrázků

2.1 Akustický kanál 13

2.2 Blokové schéma rozpoznávání 15

4.1 Schéma neuronové sítě pro jazykové modelování 50

4.2 Schéma rekurentní neuronové sítě pro jazykové modelování 51

Kapitola 1

Úvod

Řešení problému komunikace člověka s počítačem má dlouhou historii a v současné době je jednou z nejaktuálnějších disciplín umělé inteligence. Nelze ovšem říci, že tato činnost souvisí pouze s tímto oborem, neboť v sobě zahrnuje i další vědní disciplíny, například akustiku, fonetiku, teorii informace, zpracování signálů, rozpoznávání obrazů a další až po matematickou lingvistiku a porozumění.

Tato práce se bude zabývat úlohou automatického rozpoznávání souvisle mluvené řeči. Jejím cílem je zpracovat řečový signál produkovaný řečníkem a přiřadit mu text odpovídající dané promluvě. Řešení této problematiky je složité z následujících důvodů:

- stejnou promluvu vysloví každý řečník jinak
- stejnou promluvu vysloví stejný řečník pokaždé jinak
- jednotlivé části promluvy mohou být vysloveny různě rychle
- nelze jednoduše stanovit začátek a konec jednotlivých slov
- současně s promluvou jsou snímány rušivé zvuky – šum

Celý systém rozpoznávání je složen z několika základních bloků. Tato práce se bude zabývat řešením pouze určité části tohoto systému, a to jazykovým modelováním. Nebude se zabývat obecným diktovacím systémem. Specializace diktovacích

systémů na konkrétní oblast umožňuje definovat a připravit systém vhodněji pro uživatele z pohledu použití i přesnosti rozpoznávání. Práce je součástí připravovaného specializovaného systému rozpoznávání řeči pro soustavu soudů v České republice pro diktování soudních rozhodnutí.

Disertační práce je rozdělena do několika částí. V první části je stručně popsán systém rozpoznávání řeči. Po samostatné kapitole, která je věnována cílům disertace, následují dvě části zabývající jazykovým modelováním. Pátá část obsahuje detailní popis řešené úlohy a navrhuje možný postup jejího řešení. Závěrečná část disertace popisuje soubor experimentů týkajících se použití navrženého postupu a diskutuje výsledky experimentů.

Kapitola 2

Obecná úloha rozpoznávání řeči

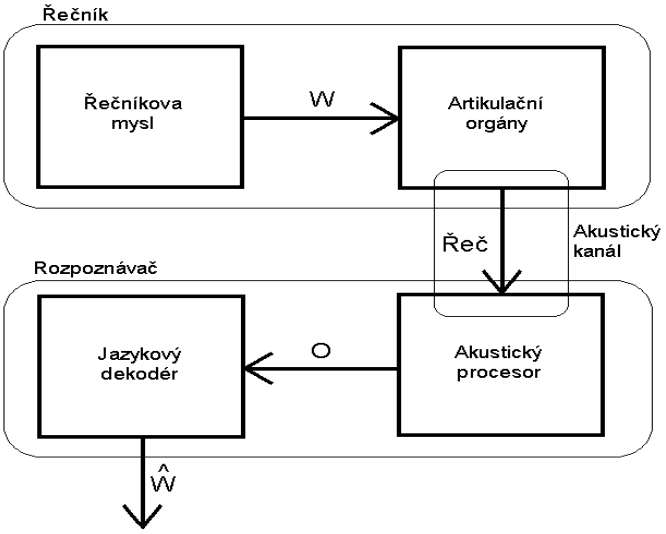
Pro úlohu rozpoznávání řeči bylo navrženo více způsobů řešení. Od padesátých let se uplatňovaly metody založené na porovnávání vzorů, tzv. template matching. Tato skupina metod se zejména uplatňovala v systémech pro rozpoznávání izolovaných slov.

Od osmdesátých let se v hojné míře využívá tzv. statistický přístup, který zatím přináší nejlepší výsledky. Prvním pokusy se statistickým přístupem jsou prezentovány od sedmdesátých let, ale až systém rozpoznávání řeči od společnosti IBM přinesl velký úspěch statistických metod.

Ve statistických metodách jsou slova a celé promluvy modelovány pomocí tzv. Markovových modelů. Nejčastěji jsou konstruovány skryté Markovovy modely subslovních jednotek a promluva je modelována zřetězením těchto subslovních modelů.

Pro použití statistických metod je potřeba velkého množství dat, ze kterého můžeme statisticky odvodit obecnější závěry, které se pak budou vztahovat nejen na dosud pozorovaná data, ale i na data nová, předložená pro rozpoznávání.

Úlohu rozpoznávání lze popsat jako systém tvořený řečníkem – člověkem a rozpoznávačem – počítačem, jež jsou spojeny akustickým kanálem, viz obrázek 2.1. Řečník ve své mysli generuje posloupnost slov, kterou následně vyslovuje pomocí artikulačních orgánů a je reprezentována zvukovým signálem. Funkcí rozpoznávače je zachytit daný zvukový signál, zpracovat ho akustickým procesorem na vektor příznaků O, který je potom převeden na předpokládanou posloupnost slov jazy-



Obrázek 2.1: Akustický kanál

kovým dekodérem.

2.1 Formulace úlohy

Označme $W = \{w_1, w_2, \ldots, w_N\}$ posloupnost slov $w_i, i = 1, \ldots, N$, vyslovených řečníkem patřících do pevného a známého slovníku V. Řečový signál odpovídající této posloupnosti slov je systémem automatického rozpoznávání nejprve konvertován na časovou posloupnost vektorů příznaků $O = \{o_1, o_2, \ldots, o_T\}$, kde o_t je vektor příznaků odpovídající t-tému mikrosegmentu promluvy. Cílem je nalézt posloupnost slov $\hat{W}$, která maximalizuje aposteriorní pravděpodobnost $P(W|O)$, tj. nejpravděpodobnější posloupnost slov pro danou akustickou informaci O.

$$\hat{W} = \arg\max_{W} P(W|O) \tag{2.1}$$

Použitím Bayesova pravidla dostaneme

$$\hat{W} = \arg\max_{W} \frac{P(W)P(O|W)}{P(O)} \tag{2.2}$$

kde $P(W)$ je apriorní pravděpodobnost, že řečník vysloví sekvenci slov W, $P(O|W)$ je pravděpodobnost, že při vyslovení posloupnosti W bude generována posloupnost

vektorů příznaků a $P(O)$ je pravděpodobnost, že bude pozorována posloupnost O. Protože $P(O)$ je pro daný akustický signál konstantní, můžeme místo maximalizace zlomku na pravé straně rovnice (2.2) maximalizovat jenom jeho čitatel. Takže snahou dekodéru je nalézt sekvenci slov, která splňuje:

$$\hat{W} = \arg\max_{W} P(W)P(O|W) \tag{2.3}$$

2.2 Dekompozice na podúlohy

Úlohu rozpoznávání lze dekomponovat na podúlohy, jak je znázorněno na obrázku 2.2. Blokové schéma naznačuje, jak na sebe jednotlivé části navazují.

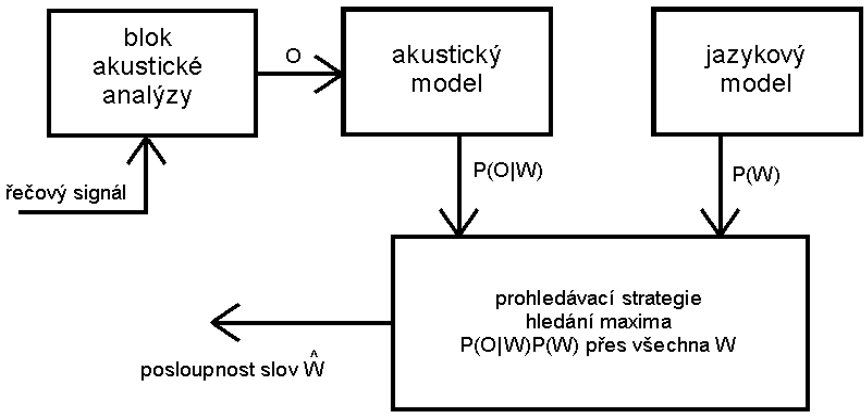
Akustická analýza

Na vstupu je akustický procesor, který převádí řečový signál na časovou posloupnost vektorů. Toto zpracování se provádí metodou melovských frekvenčních kepstrálních koeficientů, tzv. MFCC, nebo percepčních prediktivních koeficientů, tzv. PLP, které jsou odvozeny od subjektivního vnímání zvuku člověkem, nebo . Šířka slyšitelného pásma je rozdělena na různě dlouhé úseky. Pro každý tento úsek je použit pásmový filtr s různým zesílením. Obvykle se používá třináct filtrů. Takto získaná časová posloupnost vektorů jde na vstup jazykového dekodéru a dále je zpracována jazykovým dekodérem, který provádí vlastní rozpoznávání.

Akustický model

Cílem akustického modelu je poskytnout co nejpřesnější odhad pravděpodobnosti $P(O|W)$, tj. pravděpodobnosti pozorování posloupnosti O za předpokladu, že byla vyslovena posloupnost slov W. Je využíváno tzv. skrytých Markovových modelů.

Konstrukce klasifikátoru vychází z představy o vytváření řeči a je založena na modelování řečového signálu pomocí Markovova procesu. Při tomto procesu jsou generovány dvě vzájemně svázané časové posloupnosti náhodných proměnných, a to podpůrný Markovův řetězec, který je posloupností konečného počtu stavů a řetězec



Obrázek 2.2: Blokové schéma rozpoznávání

konečného počtu spektrálních vzorů. Pro jednotlivé spektrální vzory jsou vytvořeny náhodné funkce, které pravděpodobnostně ohodnocují vztah těchto vzorů ke všem stavům. V diskrétních časových okamžicích je proces v jediném stavu. Markovův řetězec pak mění stavy podle své matice pravděpodobností přechodu. Pozorovatel vidí jen výstup náhodných funkcí a nemůže pozorovat stavy podpůrného Markovova řetězce, odtud termín skrytý Markovův model (Hidden Markov Model – HMM).

Při modelování řeči se používají tzv. levo-pravé Markovovy modely, které jsou především vhodné pro modelování procesů, jejichž vývoj je spojen s postupujícím časem. Základní vlastností je, že s narůstajícím časem dochází k přechodům ze stavů s nižšími indexy do stavů s vyššími indexy nebo k setrvání ve stejném stavu. Průchod modelem je tedy zleva doprava.

Trénování těchto modelů probíhá většinou na trifónech, což je název pro foném s levým a pravým kontextem. Procesem trénování se nastaví hodnoty variance a střední hodnoty náhodných funkcí a matice přechodů mezi stavy. Využívá se Viterbiova a Baum-Welchova algoritmu. V závěru trénování se metodou shlukování pomocí rozhodovacích stromů svážou podobné stavy modelů trifónů a docílí se tím robustnějších odhadů parametrů.

Jazykový model

Úkolem jazykového modelu je nalézt apriorní pravděpodobnost $P(W)$, že řečník chce vyslovit posloupnost slov W. Takovou pravděpodobnost je nutné spočítat pro každou možnou posloupnost slov W. Tato práce se bude zabývat nejpoužívanějším typem jazykového modelování – statistickými jazykovými modely jejichž parametry jsou získávány z velkých textových korpusů.

$$P(w_1 \dots w_k) = P(w_1)P(w_2|w_1) \dots P(w_k|w_1 \dots w_{k-1}) = \prod_{k=1}^{K} P(w_k|w_1 \dots w_{k-1}) \quad (2.4)$$

Jazykovým modelováním se podrobně zabývají následující kapitoly této práce.

Prohledávací strategie

Prohledávací strategie neboli dekódování má za úkol z posloupnosti pozorování O a pravděpodobností $P(O|W)$ a $P(W)$ nalézt podle kritéria maximální aposteriorní pravděpodobnosti (MAP) takovou posloupnost slov $\hat{W}$, pro kterou součin $P(O|W)$ a $P(W)$ nabývá maximální hodnoty, viz rovnice (2.3). Pravděpodobnosti $P(O|W)$ a $P(W)$ jsou známé funkce, jejichž hodnoty je možné pro libovolnou posloupnost slov W určit výpočtem. Někdy je úloha dekódování zobecněna na úlohu nalezení více než jedné jedné posloupnosti slov $\hat{W}$. Pak mluvíme o hledání ***N* nejlepších** posloupností slov vyhovujících vztahu (2.3), tzv. ***N*–best**.

Dekódovací algoritmus, který by provedl úplné prohledání všech posloupností slov W, není možné provést z důvodu velké výpočetní náročnosti i pro relativně malé úlohy. Algoritmy pro dekódování vycházejí z teorie grafů. Akustický i jazykový model lze vyjádřit ohodnoceným orientovaným grafem a lze uplatnit princip *dynamického programování*. Nejčastěji se využívá Viterbiův algoritmus, který je vlastně problémem hledání cesty s nejmenší cenou v ohodnoceném grafu tvořeném sítí s konečným počtem stavů.

Obecně se dají dekodéry dělit podle několika kritérií. Při implementaci systému rozpoznávání řeči je vždy třeba zohlednit několik hledisek. Uvedeme jen některé, více v [35].

- **Kritérium hledání** se používá buď Viterbiho nebo MAP, které hledá skutečně nejpravděpodobnější posloupnost slov, jejíž pravděpodobnost je dána součtem pravděpodobností všech posloupností stavů, které posloupnost slov reprezentují.
- **Počet výsledných hypotéz** může být jedna nebo N, tedy jedna nejpravděpodobnější nebo seznam, případně graf, několika nejlepších hypotéz.
- **Počet průchodů hledání.** Při víceprůchodovém hledání se používá výsledek N nejlepších hypotéz prvního průchodu. Dále jsou zpracovány přesnějším jazykovým i akustickým modelem.
- **Velikost slovníku** je důležitá kvůli optimalizaci struktur stavového prostoru.

Základním cílem dekodéru je co nejvíce snížit počet výpočetních operací při řešení rovnice (2.3). Pokud dekódování zaručí nalezení nejpravděpodobnější posloupnosti slov $\hat{W}$, prohledávací strategie se nazývá strategií **optimální**. Ukazuje se ale, že není možné dosáhnout přijatelné doby řešení pro úlohy s velkými slovníky při dodržení požadavku optimálního řešení. Proto je prohledávací strategie dekódování téměř vždy **suboptimální**. Suboptimální prohledávací strategie používá k nalezení rychlého řešení vhodné heuristiky, nicméně nalezení globálně nejlepšího řešení nezaručuje. Hlavním cílem dekódovaní je pak dosáhnout co nejlepšího kompromisu mezi rychlostí řešení a přesností řešení.

Kapitola 3

Cíle disertační práce

Statistické jazykové modelování je jednou z důležitých součástí systémů pro rozpoznávání řeči. V moderních diktovacích systémech se vyžaduje maximální přesnost rozpoznávání, kterou lze výrazně ovlivnit zvoleným jazykovým modelem. Obecné diktovací systémy nejsou téměř používány a vždy se daný systém přizpůsobuje diktovanému oboru, například právnické a lékařské texty. Toto přizpůsobení má velký vliv na přesnost rozpoznávání a komfort používání diktovacích systémů. Přesto i při zvolení konkrétních oborů zůstávají části textu, pro které nelze statistiku věrohodně získat nebo odhadnout. V právnických textech jde o jména osob, názvy sídel a ulic nebo jména firem a společností. Tato slova se nacházejí v trénovacích textech, ovšem v příliš řídkém počtu, že nelze používanými metodami získat jazykový model pro správné rozpoznání této množiny slov. Většina slov z této množiny se v trénovacích textech vůbec nevyskytla, proto nejsou v jazykových modelech vůbec uvažována. Doplnění celé této množiny jazykového modelu a nastavení modelu pro správné rozpoznávání bude hlavním cílem této disertační práce. Cílem nebude zlepšení přesnosti rozpoznávání diktovacího systému jako celku, ale pouze zlepšení rozpoznávání zvolené omezené množiny slov při minimálním negativním vlivu na správnost rozpoznávání zbylého textu.

3.1 Dílčí cíle práce

1. Prostudovat a popsat standardní postupy tvorby jazykových modelů užívaných v systémech rozpoznávání řeči.

2. Detailně prostudovat a popsat metody jazykového modelování pro doplňování slovníku rozpoznávání. Z dostupných publikovaných výsledků zjistit jakých bylo dosahováno zlepšení oproti standardnímu postupu a na jakých úlohách.

3. Navrhnout a ověřit metodu pro vhodné doplňování slov do rozpoznávacího slovníku a nastavení pravděpodobností modelu.

4. Provést experimenty s rozpoznáváním namluvené řeči na maximálním množství různých dostupných dat.

5. Výsledkem práce byl měl být postup pro obohacení jazykového modelu diktovacího systému o slova specifických skupin včetně co nejlepšího nastavení pravděpodobností pro jejich správné rozpoznávání při minimálním ovlivnění správnosti rozpoznávání celého systému.

Kapitola 4

Statistické jazykové modely

V této části budou popsány základní principy při tvorbě jazykových modelů založených převážně na n-gramech. V modelování jazyka je třeba uvážit, že každý jazyk má své zákonitosti a vlastnosti. Nejdůležitějšími jsou zřejmě používaný slovník daným jazykem a pravidla řetězení slov do větných celků. Každému slovu ve slovníku je možné přiřadit výslovnost nebo dokonce několik variant výslovnosti a tím jsou určeny posloupnosti fonémů v daném jazyce. Soubor těchto vlastností tedy přesně určuje jaký jazyk bude systémem pro rozpoznávání řeči přijímán a je zřejmé, že jazykový model není nikdy obecně nastaven na všechny možné situace, ve kterých se řečník může nacházet. Konkrétní jazykový model bere v úvahu celý kontext promluvy, který popisuje mnoha atributy, například jazyk, téma, úmysl řečníka význam sdělení, stav dialogu, řečníkovu náladu a další.

Principem jazykového modelování je snaha odhadnout pravděpodobnost $P(W)$ každé posloupnosti slov $W = (w_1, w_2, \ldots, w_k)$. Při využití dostatečně velkého textového korpusu lze parametry modelu odhadnout z četností jednotlivých posloupností slov v textu. Nejjednodušším přístupem by bylo přiřadit každému slovu stejnou pravděpodobnost. Tedy například při velikosti slovníku sto tisíc slov, by byla pravděpodobnost každého slova $\frac{1}{100\ 000}$. Cílem je ovšem odhadnout pravděpodobnost lépe. Je zřejmé, že člověk dokáže určit slovo nebo skupinu slov, která budou pravděpodobně následovat po slyšené sekvenci a naopak vyřadit z této skupiny slova málo pravděpodobná.

Předpokládejme kompletní posloupnost slov $W = (w_1, w_2, \ldots, w_k)$, alternativně

můžeme označit w_1^n. Za předpokladu, že každé slovo je na správném místě a je nezávislým jevem, lze zapsat

$$P(w_1, w_2, \ldots, w_k) \tag{4.1}$$

Použití pravidla pro zřetězení pravděpodobnosti:

$$P(w_1^n) = P(w_1, w_2, \ldots\ , w_k) = P(w_1)P(w_2|w_1)P(w_3|w_1^2)\ldots P(w_n|w_1^{n-1}) = \tag{4.2}$$

$$= \prod_{k=1}^{n} P(w_k|w_1^{k-1})$$

Tento rozklad je vhodný pro praktickou implementaci jazykového modelu v úloze dekódování. Umožní rozpoznávat posloupnost slov již v průběhu jejího vyslovování a určovat pravděpodobnosti $P(w_1^k)$ pro účely dekódování postupně pomocí již vypočtených $P(w_1^{k-1})$. V praxi se často zavádí kromě slov i symbol pro konec a začátek vět. Promluva W tedy může obsahovat celou promluvu s několika větami s informací o jejich začátku a konci.

Pro konstrukci jazykového modelu požadujeme znalost apriorních pravděpodobností $P(w_1^k)$ všech posloupností slov libovolné délky K Všechny tyto pravděpodobnosti je obtížné a téměř nemožné ocenit. Provádí se proto aproximace, kde všechny historie $w_1, \ldots, w_{i-2}, w_{i-1}$ které se shodují v posledních $n-1$ slovech jsou zařazeny do stejné třídy. Takové stochastické modely se nazývají stochastické **n-gramové modely**. n-gramem se rozumí posloupnost n za sebou jdoucích slov v pozorování jejich náhodného výběru. n-gramům s $n = 0$ se říká **zerogramy** a s $n = 1$ **unigramy**. Nejvíce používané pro češtinu jsou **bigramy** a **trigramy**, pro $n = 2$ a $n = 3$. Pro modelování jazykových závislostí by vhodný řád modelu měl být vyšší než $n = 3$. Z praktického hlediska se ovšem vyšší řády používají velmi málo.

Pro n=gramový mnodel je podmíněná pravděpodobnost slova na pozici k závislá pouze na $n-1$ předcházejících slovech. Aproximace $P(w_k|w_1^{k-1}) \approx P(w_k|w_{k-n+1}^{k-1})$. Platí

$$P(w_1^k) = \prod_{i=1}^{k} P(w_i|w_{i-n+1}1^{i-1}) \tag{4.3}$$

n-gram model	Počet parametrů
2	4×10^8
3	8×10^{12}
4	$1,6 \times 10^{17}$
5	$3,2 \times 10^{21}$

Tabulka 4.1: Počet parametrů modelu pro slovník 20 000 slov

Vlastnosti n-gramových modelů jsou vhodné zejména pro jazyky s relativně pevným pořádkem slov ve větě, kde jsou silné statistické závislosti výskytů následujících slov. Výhodný je rovněž poměrně snadný odhad pravděpodobnosti výskytu slov. Estimace jednotlivých pravděpodobností je založena na zjišťování relativní četnosti výskytů slov a jejich posloupností v textových trénovacích korpusech. Pro odhad trigramové pravděpodobnosti platí

$$\bar{P}(w_k|w_{k-2}wk-1) = \frac{N(w_{k-2}, wk-1, w_k)}{N(w_{k-2}, wk-1)}, \tag{4.4}$$

kde $N(w_{k-2}, wk-1, w_k)$ je počet kolikrát se vyskytl trigram $w_{k-2}, wk-1, w_k$ v trénovacích datech a $N(w_{k-2}, wk-1)$ kolikrát se ve stejných datech objevil bigram $w_{k-2}, wk-1$. Získání těchto počtů n=gramů ovšem v praxi není bez komplikací. Například uvažujme text, který obsahuje celkem 350 tisíc slov, z nichž je 20 tisíc různých – velikost slovníku. Pro takto velký slovník a zvolený řád n-gramového modelu je potřeba odhadnout počet parametrů uvedený v tabulce 4.1. Porovnáním jen pro bigramy, kde počet parametrů výrazně převyšuje počet všech bigramů v textu, vidíme, že data pro dokonalé nastavení parametrů budou vždy nedostatečná a musíme využít metody pro odhad pravděpodobnosti. Slovník s 20 tisíci slovy je v dnešních rozpoznávacích systémech spíše menším a typický spíše pro angličtinu. Pro český jazyk se používají slovníky o velikosti stovek tisíc slov.

4.1 Kvalita jazykového modelu

Tvorba jazykových modelů je široká úloha a pro různé modely vytvořené různými metodami je potřeba mít možnost dané modely porovnat a uvést míru, která ohodnotí kvalitu jazykového modelu. Nejlepším možným posouzením kvality jazykového modelu je provést celý proces rozpoznávání řeči a porovnat úspěšnost celého systému s různými jazykovými modely. Porovnávat můžeme modely mezi sebou, nebo například se systémem, kde nebyl použit žádný jazykový model, nebo s nějakým základním jazykovým modelem vůči němuž hledáme zlepšení. Nejčastěji se používají míry **Accuracy** a **Correctness** (přesnost a správnost), případně **WER** - Word Error Rate. Proces rozpoznávání je značně náročný a jazykový model může být vyhodnocen i odděleně od ostatních částí rozpoznávacího systému. Nejpoužívanější mírou je tzv. **perplexita**.

4.1.1 Perplexita

Definujme entropii pravděpodobnosti s distribuční funkcí $p(x)$

$$H(p) = -\sum_{x \in \Omega} p(x) \log_2 p(x), \tag{4.5}$$

kde Ω je stavový prostor dané pravděpodobnosti. Logaritmus o základu dva je používán pro získání entropie v bitech. Někdy je vhodné používat logaritmus o základu deset nebo přirozeného logaritmu – obvykle při minimalizaci kritéria optimality.

Dále definujme vzájemnou entropii dvou pravděpodobnostních rozdělení p a q

$$H_p(q) = -\sum_{x \in \Omega} p(x) \log_2 q(x) \tag{4.6}$$

Bylo dokázáno, že $H(p) \leq H_p(q)$ (s rovností), tehdy a jen tehdy, když $p(x) = q(x)$ pro všechna x.

Tyto dva vzorce mají svojí interpretaci v jazykovém modelování. Předpokládejme, že jazyk má pravděpodobnostní rozdělení p a máme dva jazykové modely s rozdělením p_{LM1} a p_{LM2}. Potom považujeme za lepší ten model, jehož pravděpodobnostní rozložení je bližší správnému rozdělení p. Tedy, $LM1$ je lepší než $LM2$, když $H_p(p_{LM1}) < H_p(p_{LM2})$.

Zůstává otázkou, jak určit správné rozložení pravděpodobnosti p. Může být simulována testovacími daty – tedy daty, která nebyla použita pro odhad pravděpodobností jazykového modelu. Potom můžeme definovat analogii vzájemné entropie – *logprob* definované jako

$$LP = -\frac{1}{K}\sum_{i=1}^{S} log_2 P_{LM}(w_i|\Phi(w_1, w_2, \ldots, w_{i-1})), \tag{4.7}$$

kde K je počet tokenů – slov v testovacích datech a $P_{LM}(w_i|\Phi(w_1, w_2, \ldots, w_{i-1}))$ je pravděpodobnost přiřazená slovu w_i jazykovým modelem. $\Phi(w_1, w_2, \ldots, w_{i-1}))$ je funkce „slovní historie“. Předpokládejme, že $P_{LM}(w_i|\Phi(w_1, w_2, \ldots, w_{i-1}))$ je rovno nule, pokud w_i není ve slovníku, tedy je ve skupině OOV slov.

Obvykle používaná míra pro kvalitativní hodnocení jazykových modelů je perplexita, která je odvozena od *logprob* jako

$$PP = 2^{LP}, \tag{4.8}$$

kde základ mocniny je stejný jako základ logaritmu u *logprob*.

Druhé možné vyjádření perplexity je pomocí ocenění apriorní pravděpodobnosti $P(W) = P(w_1 w_2 \ldots w_k)$ výskytu posloupnosti slov W, kterou poskytuje jazykový model. Odhad označme jako $\bar{P}(W)$. Čím větší hodnotu bude mít odhad $\bar{P}(W)$ pro dostatečně dlouhý korpus tím větší význam bude mít jazykový model pro úspěšnost rozpoznávání. V korpusu vyjádřeném jako posloupnost slov W obsahující K slov je v průměru pravděpodobnost posloupnosti K slov K-krát menší než pravděpodobnost posloupnosti délce jednoho slova. Je tedy vhodné odhad $\bar{P}(W)$ normalizovat vzhledem k počtu slov K funkcí příslušné odmocniny.

$$\sqrt[K]{\bar{P}(w_1 w_2 \ldots w_k)} \tag{4.9}$$

Díky této normalizaci lze srovnávat kvalitu jazykových modelů na různě dlouhých korpusech. Pro tuto normalizaci platí, že čím větší entropii bude mít daný korpus, tím menší hodnotu dostaneme. Definujme obrácenou hodnotu jako perplexitu PP

$$PP = frac1 \sqrt[K]{\bar{P}(w_1 w_2 \ldots w_k)} \tag{4.10}$$

jejím zlogaritmováním získáme

$$LP = log_2 PP = -\frac{1}{K}\sum_{i=1}^{K} log_2 \bar{P}(W) \tag{4.11}$$

Pro n-gramové modely můžeme zapsat ve tvaru

$$LP = -\frac{1}{K}\sum_{i=1}^{K} log_2 \bar{P}(w_i|w_{i-n+1}w_{i-n+1}\dots w_{i-1}), \tag{4.12}$$

který odpovídá 4.7.

Perplexitu je možno vysvětlit takto. Je-li složitost úlohy PPL, pak úloha rozpoznávání řeči je tak obtížná, jako kdyby jazyk měl PPL stejně pravděpodobných slov. Tedy čím nižší perplexita, tím lepší jazykový model.

Hodnota perplexity se počítá pouze pro slova obsažená ve slovníku, tj. ukazuje jak dobře jsou přeřazeny pravděpodobnosti v jazykovém modelu, ale nepostihuje míru pokrytí testovacích dat slovníkem. Pokud bude mnoho slov v testovacích datech neznámých bude perplexita nízká a mohlo by to vypadat, že jazykový model je dobrý. Z tohoto důvodu se kromě perplexity při porovnávání jazykových modelů uvádí hodnota OOV, která ukazuje, kolik slov z trénovacích dat není ve slovníku.

K porovnání kvality jazykových modelů potřebujeme znát OOV i perplexitu. Ke správnému porovnání modelů musíme konstruovat modely bez znalosti testovacích dat. Jakákoliv jejich znalost způsobí uměle nízké hodnoty perplexity i OOV. Dva jazykové modely jsou porovnatelné pouze tehdy pokud používají stejný slovník.

4.1.2 Accuracy, Correctness

Přesnost a správnost jsou míry používané při vyhodnocování systémů rozpoznávání řeči. Jazykový model značně ovlivňuje výsledek rozpoznávání. Nastavením jazykového modelu a porovnáním výsledků rozpoznávání můžeme modely srovnávat podle kvality. Přesnost – *Accuracy* se počítá následovně

$$Acc = \frac{N - D - S - I}{N} \cdot 100\%, \tag{4.13}$$

kde N značí počet slov v rozpoznávacím textu, vzor pro rozpoznávání. D, S a I jsou počty slov, která byla rozpoznávacím systémem vypuštěna (deletion), nahrazena (substitution) a vložena (insertion) ve výsledku rozpoznávání. Někdy je používána hodnota Správnost – *Correctness*, kde nejsou do výpočtu zahrnuty chyby vložení slov – insertions.

$$Corr = \frac{N - D - S}{N} \cdot 100\% \tag{4.14}$$

V literatuře se často uvádí míra Word Error Rate, tzv. WER. Jde o přepočet na chybná slova z míry Accuracy, tj.

$$WER = 100 - Acc \tag{4.15}$$

Rozdíl mezi přesností a správností je právě ve vložených slovech a jejich vlivu na výsledek rozpoznávání. V úlohách automatického diktátu jsou vložená slova v rozpoznávaném textu stejnou chybou jako vypuštění slova. Naopak v systémech s hledáním informací se preferuje neztracení žádného slova z rozpoznaného textu a vložená slova jsou tolerována. Tyto dvě míry porovnávají celý systém rozpoznávání řeči a jejich získání je časově náročné právě kvůli celému procesu rozpoznávání. Pro porovnání samotných jazykových modelů se používají další metody vycházející z teorie informace.

4.2 Odhady pravděpodobností jazykového modelu

Nejpoužívanějším typem jazykových modelů jsou n-gramové modely. Již bylo uvedeno, že jde o modely, které přidělují pravděpodobnost posloupnosti slov o délce n. na základě známé historie $n-1$ slov. Cílem navrhovaných modelů je co nejlepší přiblížení k reálnému jazyku a současně i malá paměťová a výpočetní náročnost v aplikacích. Problém spočívá ve velikosti čísla n. Volba n má velký vliv na věrohodnost modelu a množství potřebných parametrů modelu. Nejvíce se používají hodnoty $n = 2, 3, 4$ a modely se pak nazývají bigramové, trigramové a čtyřgramové. Problém s vyšším n je způsoben nedostatečností dat pro nastavení parametrů modelu.

V následujících částech budou popsány metody, které jsou schopny odhadnout pravděpodobnosti i z nedostatečných dat.

Pro srozumitelnější zápis budeme používat označení pro posloupnost slov

$$P(w_i|w_{i-n+1}, w_{i-n+2}, \ldots, w_{i-1}) = P(w_i|w_{i-n+1}^{i-1}) \tag{4.16}$$

4.2.1 Metoda maximální věrohodnosti

Metoda maximální věrohodnosti (Maximum likelihood estimation – MLE) neřeší přímo nedostatečnost dat. Je to přímá metoda odhadu pravděpodobností z velkého množství dat a další metody částečně modifikují tento základní odhad. Odhad pravděpodobnosti $P(w_i|w_{i-n+1}^{i-1})$ je počítán relativní frekvencí

$$P_{MLE}(w_i|w_{i-n+1}^{i-1}) = \frac{C(w_{i-n+1}^{i})}{\sum_{w_i \in V} C(w_{i-n+1}^{i})}, \tag{4.17}$$

kde $C(w_{i-n+1}^{i})$ je počet kolikrát se objevil n-gram w_{i-n+1}^{i} v trénovacích datech.

Odhad (4.17) počítá veškeré odhady pravděpodobností z trénovacích dat, tzn. že nezbývá žádná pravděpodobnost na události, které se v trénovací množině nevyskytly. Znamená to, že přiřadí nulovou pravděpodobnost všem neviděným n-gramům. To je nepřípustné v jazykovém modelování. Skutečnost, že se nevyskytly některé n-tice slov v trénovací množině neznamená, že se nemohou vyskytnout v běžném jazyce.

Je-li cílem přiřadit část pravděpodobnosti neviděným událostem, je třeba snížit odhady pravděpodobností určené metodou maximální věrohodnosti. Tyto metody jsou tzv. discounting metody – provádějí snižování pravděpodobností a proces, který je provádí se nazývá smoothing – vyhlazování. Metody pro snižování pravděpodobnosti budou popsány dále. Metodu MLE bez odpočítávání pravděpodobnosti lze použít v případech, kdy můžeme dostatečně jistě tvrdit, že řečník skutečně vysloví jenom data, která se vyskytují v trénovací množině.

4.2.2 Add-one smoothing

Tato vyhlazovací technika připočte jedničku ke každému počtu n-gramů. Někdy se též nazývá Laplaceovým pravidlem. Pravděpodobnost se počítá takto

$$P_{Lap} = (w_i|w_{i-n+1}^{i-1}) = \frac{C(w_{i-n+1}^{i}) + 1}{\sum_{w_i \in V} C(w_{i-n+1}^{i}) + |V|} \tag{4.18}$$

kde $|V|$ je velikost slovníku.

Přidání jedničky eliminuje nulové pravděpodobnosti neviděných událostí, ale prakticky se tato metoda příliš nepoužívá. Když $|V| > \sum_{w_i \in V} C(w_{i-n+1}^i)$ (a to je téměř vždy, protože $|V|$ je řádově v desítkách až stovkách tisíc), dává tento odhad pravděpodobnosti příliš velké množství pravděpodobnosti neviděným událostem, když nepřiměřeně snižuje pravděpodobnosti více frekventovaným.

4.2.3 Held-out estimace

Základní myšlenkou held-out estimace je rozdělení trénovacích dat na dvě části – vývojová data (development data) a odložená data (held-out data). Počty událostí jsou počítány z vývojových dat a jsou vyhlazovány pomocí odložených dat. Obvykle se volí 90% trénovacích dat jako vývojových. Zaveďme následující označení:

$C_1(x)$	počet výskytů události x ve vývojových datech
$C_2(x)$	počet výskytů události x v odložených datech
n_r	počet událostí, které se vyskytly ve vývojových datech právě r-krát $n_r = \sum_{x:C_1(x)=r} 1$
h_r	počet kolikrát se událost, která je ve vývojových datech, objevila v odložených datech $h_r = \sum_{x:C_1(x)=r} C_2(x)$
N_1	počet vzorků ve vývojových datech
N_2	počet vzorků v odložených datech
N	celkový počet vzorků $N = N_1 + N_2$

Mějme $n_r = 0$ pro všechna $r > R$ a dostaneme

$$\sum_{r=0}^{R} r n_r = N_1 \tag{4.19}$$

Platí následující podmínky:
Všechny události x vyskytující se r-krát ve vývojových datech mají stejnou pravděpodobnost P_r, tj.

$$P(x) = P_r \qquad x : C_1(x) = r \tag{4.20}$$

a pravděpodobnosti musí splňovat obvyklou podmínku

$$\sum_{r=0}^{R} n_r P_r = 1 \tag{4.21}$$

Cílem je nalézt P_r, která maximalizuje pravděpodobnost odložených dat, která může být vyjádřena pomocí P_r a h_r jako

$$P(held-out\ data) = \prod_{r=0}^{R} {P_r}^{h_r} \tag{4.22}$$

Po zlogaritmování dostaneme logaritmickou věrohodnostní funkci

$$L(P_0, P_1, \ldots, P_R) = \sum_{r=0}^{R} h_r \ln P_r \tag{4.23}$$

Začleněním normalizační podmínky a použitím metody Lagrangeových multiplikátorů máme maximalizovat

$$L(P_0, P_1, \ldots, P_R) = \sum_{r=0}^{R} h_r \ln P_r - \mu(\sum_{r=0}^{R} n_r P_r - 1) \tag{4.24}$$

Derivováním podle P_r a položením rovno nule, dostaneme

$$P_r = \frac{1}{\mu}\frac{h_r}{n_r} \qquad r = 0, 1, \ldots, R \tag{4.25}$$

kde μ získáme z normalizační podmínky

$$\sum_{r=0}^{R} n_r P_r = \frac{1}{\mu}\sum_{r=0}^{R} h_r = 1 \quad \Rightarrow \quad \mu = \sum_{r=0}^{R} h_r = N_2 \tag{4.26}$$

4.2.3.1 Good-Turing discounting

Tato metoda je modifikací held-out estimace, někdy se nazývá také *leaving-one-out*, neboli vynechávání jednoho slova. Základní myšlenkou je rozdělení trénovacích dat na vývojová a odložená data tak, že v odložených datech je jenom jedno slovo. Toto je provedeno N-krát, takže každé slovo je použito ve vývojových i odložených datech. V této metodě lze h_r vyjádřit pomocí r a n_r.

Předpokládejme, že jednotlivé slovo je vzato jako odložená data. Jestliže se vyskytovalo $(r+1)$-krát v trénovacích datech, tak po jeho vyjmutí se ve vývojových datech vyskytuje pouze r-krát a je mu přiřazena pravděpodobnost P_r. Tento princip se použije na všechna slova, která se vyskytují $(r+1)$-krát v trénovacích datech a jsou vybrána jako held-out část. Taková situace nastane přesně $((r+1)n_{r+1})$-krát.

Dostaneme

$$h_r = (r+1)n_{r+1} \qquad r = 0, 1, \ldots, R-1 \tag{4.27}$$

a dosadíme do logaritmické věrohodnostní funkce

$$L(P_0, P_1, \ldots, P_R) = \sum_{r=0}^{R-1}(r+1)n_{r+1} \ln P_r - \mu(\sum_{r=0}^{R} n_r P_r - 1) \tag{4.28}$$

Po maximalizaci přes P_r dostaneme odhad pravděpodobnosti metodou vynechání jednoho slova (leaving-one-out probability estimate).

$$P_r = \frac{1}{\mu}\frac{(r+1)n_{r+1}}{n_r} \qquad r = 0, 1, \ldots, R \tag{4.29}$$

Pravděpodobnost P_R nemůže být odhadnuta touto metodou a musí být získána jinak, například relativní frekvencí.

Lagrangeův multiplikátor μ je vypočítán z normalizační podmínky (4.21)

$$\sum_{r=0}^{R} n_r P_r = \frac{1}{\mu}\sum_{r=0}^{R-1}(r+1)n_{r+1} + n_R P_R = \frac{1}{\mu}\sum_{r'=1}^{R'} r' n_r + n_R P_R = \frac{1}{\mu}N + n_R P_R = 1$$

$$\Rightarrow \qquad \frac{1}{\mu} = \frac{1 - n_R P_R}{N} \tag{4.30}$$

Většinou lze hodnotu $n_R P_R$ zanedbat a získáme odhad pravděpodobnosti ve formě

$$P_r = \frac{1}{N}\frac{(r+1)n_{r+1}}{n_r} \tag{4.31}$$

Tato rovnice je známa jako Good-Turingův odhad pravděpodobnosti (Good-Turing probability estimate)

Pravděpodobnost přidělená neviděným slovům je dána

$$n_0 P_0 = \frac{n_1}{N} \tag{4.32}$$

Nyní se pokusíme porovnat Good-Turingův odhad s relativní frekvencí. Definujme modifikovaný počet $r^\star$ a snižující (discount) koeficient d_r.

$$r^\star = \frac{(r+1)n_{r+1}}{n_r} \qquad d_r = \frac{r^\star}{r} \tag{4.33}$$

Potom můžeme pravděpodobnost vyjádřit pomocí d_r jako

$$P_r = d_r \frac{r}{N} \tag{4.34}$$

Vrátíme-li se k jazykovým modelům, tak Good-Turingův odhad podmíněné pravděpodobnosti $P(w_i|w_{i-n+1}^i)$ je definován

$$P_{GT}(w_i|w_{i-n+1}^i) = d_{C(w_{i-n+1}^i)} \frac{C(w_{i-n+1}^i)}{\sum_{w \in V} C(w_{i-n+1}^i)} \tag{4.35}$$

kde

$$d_{C(w_{i-n+1}^i)} = \frac{(C(w_{i-n+1}^i)+1)n_{C(w_{i-n+1}^i)}}{C(w_{i-n+1}^i)n_{C(w_{i-n+1}^i)}} = \frac{C^\star(w_{i-n+1}^i)}{C(w_{i-n+1}^i)} \tag{4.36}$$

Koeficient d_r může nabývat různých tvarů a od nich jsou potom odvozeny další metody snižování pravděpodobnosti.

4.2.3.2 Absolute discounting

Tato metoda snižování pravděpodobnosti uvažuje koeficient d_r ve tvaru

$$d_r = \frac{r-b}{r} \qquad 0 < b = const(r) < 1 \tag{4.37}$$

kde $b = const(r)$ znamená, že b je konstanta závislá na r. Tohle snižování pravděpodobnosti mnohem více ovlivňuje vysoké počty slov než malé.

Pravděpodobnost, která zbude pro neviděné události je

$$n_0 P_0 = 1 - \sum_{r=1}^{R} n_r P_r = 1 - \sum_{r=1}^{R} n_r \frac{r-b}{N} = \frac{b}{N} \sum_{r=1}^{R} n_r = b\frac{X-n_0}{N} \tag{4.38}$$

kde X je počet všech možných slov. Pravděpodobnost P_r je definována

$$P_r = \begin{cases} \frac{r-b}{N} & r > 0 \\ \\ b\frac{X-n_0}{Nn_0} & r = 0 \end{cases} \tag{4.39}$$

Logaritmická pravděpodobnostní funkce je

$$L(b) = n_1 \ln b + n_1 \ln \frac{X-n_0}{N} + \sum_{r=2}^{R} r n_r \ln(r-1-b) \tag{4.40}$$

Hledáme řešení pro $0 < b < 1$. Zderivujeme podle b, položíme rovno nule a vytkneme výraz pro $r = 0$ ze sumy po roznásobení $b(1-b)$

$$\frac{n_1}{b} - \frac{2n_2}{1-b} = \sum_{r=3}^{R} \frac{r n_r}{r-1-b}$$

$$n_1 - b(n_1 + 2n_2) = \frac{b(1-b)}{2-b} \sum_{r=3}^{R} r n_r \frac{2-b}{r-1-b} \tag{4.41}$$

Protože je pravá strana rovnice (4.41) pozitivní pro všechna $0 < b < 1$, můžeme napsat horní mez b_0 pro b.

$$b \leq b_0 = \frac{n_1}{n_1 + 2n_2} < 1 \tag{4.42}$$

V rovnicích jsme předpokládali, že n_1, n_2 jsou kladná, což je splněno pro všechna reálná trénovací data. Pro volbu parametru b je většinou dostačující, položíme-li $b = b_0$.

4.2.4 Witten-Bell discounting

Witten-Bellova metoda je založena na jednoduché, ale chytré myšlence, týkající se událostí s nulovým výskytem. Uvažujme, že slova (případně n-gramy) s nulovým výskytem jsou události, které se doposud nestaly. Pokud se objeví, bude to poprvé, kdy uvidíme takovou událost. Takže pravděpodobnost prozatím neviděných n-gramů lze modelovat pomocí pravděpodobnosti n-gramů viděných jednou. Je to rekurzivní princip ve statistickém jazykovém modelování a používá se vlastně i při Good-Turingově metodě.

Pravděpodobnost, že bude n-gram spatřen poprvé, vyčíslíme počtem, kolikrát jsme viděli takové n-gramy, které se v trénovacích datech objevily jedenkrát. Jde tedy o počet různých n-gramů.

Pro pravděpodobnost dosud neviděných událostí platí

$$n_0 P_0 = \frac{T}{N+T} = \frac{T}{\sum_{w \in V} C(w_{i-n+1}^{i}) + T}, \tag{4.43}$$

kde N je počet událostí a T je počet různých událostí v trénovací množině.

Rovnici (4.43) je možno chápat jako MLE odhad pravděpodobnosti výskytu dosud neviděné události.

Pravděpodobnost z rovnice (4.43) musíme ubrat z pravděpodobností všech viděných n-gramů, pro ně dostaneme

$$P_{WB}(w_i | w_{i-n+1}^{i-1}) = \frac{C(w_{i-n+1}^{i})}{\sum_{w \in V} C(w_{i-n+1}^{i}) + T} \tag{4.44}$$

Witten-Bellův vztah vypadá pro unigramy velmi podobně jako metoda *add-one smoothing*. Podíváme-li se na bigramy uvidíme velký rozdíl. Nyní jsou totiž události závislé na historii. Pro odhad pravděpodobnosti bigramu $w_{i-1}w_i$, který jsme dosud neviděli použijeme pravděpodobnost, že uvidíme nový bigram, který začíná w_{i-1}. Zajistíme tím odhady specifické pro danou historii. Slova vyskytující se v menším počtu bigramů poskytnou menší odhad než slova, která jsou zastoupená více.

Tento fakt způsobí. že hodnota T v rovnici 4.44 je závislá na historii slov pro daný n-gram w_{i-n+1}^{i}. Je to přesně počet různých slov, které následují historii w_{i-n+1}^{i-1} v trénovací množině.

Stejně jako u ostatních metod lze vyčíslit tzv. *discount koeficient*

$$d_{C(w_{i-n+1}^{i-1})} = \frac{\sum_{w \in V} C(w_{i-n+1}^{i})}{\sum_{w \in V} C(w_{i-n+1}^{i}) + T} \tag{4.45}$$

Žádná z uvedených metod snižování pravděpodobnosti není používána samostatně, ale odhady pro jednotlivé n-gramy jsou používány v dalších konstrukcích, například back-off.

4.3 Kombinování odhadů

Snižováním pravděpodobnosti, které jsme dosud popisovali, jsme se snažili vyřešit problém neviděných n-gramů. Můžeme ovšem použít i další znalosti. Pokud nemáme žádný případ jednotlivého trigramu $w_{n-2}w_{n-1}w_n$, tak pro počítání pravděpodobnosti $P(w_n|w_{n-2}w_{n-1})$ můžeme použít pravděpodobnost $P(w_n|w_{n-1})$. Podobně pokud neznáme $P(w_n|w_{n-1})$ zaměříme se na unigram $P(w_n)$.

4.3.1 Lineární interpolace

Tato metoda, někdy též nazývána *deleted interpolation*, kombinuje rozdílné řády n-gramů lineární iterpolací všech tří modelů, když počítáme trigramy. Odhadujeme tedy pravděpodobnost $P(w_n|w_{n-2}w_{n-1})$ smícháním pravděpodobností unigramů, bigramů i trigramů. Každá z těchto pravděpodobností je ohodnocena váhou λ_i

$$\hat{P}(w_n|w_{n-2}w_{n-1}) = \lambda_3 P(w_n|w_{n-2}w_{n-1}) + \lambda_2 P(w_n|w_{n-1}) + \lambda_1 P(w_n) \tag{4.46}$$

pro koeficienty λ_i platí: $\sum_i \lambda_i = 1$.

V praxi se nepoužívají jenom konstantní koeficienty λ_i, ale často se mění každé λ_i na funkci historie trigramů. Pokud máme přesné počty jednotlivých bigramů, předpokládáme, že počty trigramů založených na bigramech budou více důvěryhodné, a proto můžeme koeficienty pro tyto trigramy zvýšit a tím dát trigramům větší váhu v interpolaci. Rovnice (4.46) se změní na

$$\hat{P}(w_n|w_{n-2}w_{n-1}) = \lambda_3(w_{n-2}^{n-1})P(w_n|w_{n-2}w_{n-1})+\lambda_2(w_{n-2}^{n-1})P(w_n|w_{n-1})+\lambda_1(w_{n-2}^{n-1})P(w_n) \tag{4.47}$$

Trénovací data pro interpolovaný model musí být rozdělena na vývojovou (development) a odloženou (held-out) část. Odhady maximální věrohodnosti jsou odvozeny z vývojových dat a následně optimální váhy λ_i minimalizací *logprob* funkce z odložených dat.

$$LP(HO) = -\frac{1}{N_2}\sum_{i=1}^{N_2} \ln \hat{P}(w_n|w_{n-2}w_{n-1}) \tag{4.48}$$

Pro minimalizaci se využívá tzv. EM algoritmu. Jeho detailní popis lze nalézt v [2].

Lineární interpolace je používána pro svoji obecnost. Interpolovaný model nebude nikdy horší než nějaká z jeho součástí.

4.3.2 Backoff

Druhou metodou, jak kombinovat jazykové modely je backoff, kterou poprvé popsal Katz [17] v roce 1987. V backoff modelu stavíme n-gramový model z $(n-1)$-gramového, stejně jako u lineární interpolace. Rozdíl je, že pokud máme nenulový počet výskytů trigramu, spoléháme se pouze na něj a neiterpolujeme jej počtem bigramů a unigramů. K n-gramům nižších řádů se vracíme (backoff) pouze, když je nulový počet n-gramů vyšších řádů.

Trigramová verze backoff modelu může být reprezentována takto

$$\hat{P}(w_i|w_{i-2}w_{i-1}) = \begin{cases} P(w_i|w_{i-2}w_{i-1}), & C(w_{i-2}w_{i-1}w_i) > 0 \\ \alpha_1 P(w_i|w_{i-1}) & C(w_{i-2}w_{i-1}w_i) = 0 \ \wedge \ C(w_{i-1}w_i) > 0 \\ \alpha_2 P(w_i) & jinak \end{cases} \tag{4.49}$$

Je to pouze orientační zápis, který dále upřesníme.

Zatím jsme předpokládali, že neviděné události, jimž přiřazujeme pravděpodobnost, kterou jsme pro ně odečetli od viděných událostí, jsou stejně pravděpodobné a lze tuto část rozdělit rovnoměrně mezi ně. Nyní můžeme discounting kombinovat s *backoff* algoritmem pro „chytřejší“ přiřazení pravděpodobností. Discounting použijeme pro získání pravděpodobnosti neviděných událostí, kterou rozdělíme *backoff* algoritmem.

Koeficienty $\alpha_1,\ \alpha_2$ v rovnici (4.49) jsou zavedeny kvůli podmínce, která říká, že pravděpodobnost slova w_n přes všechny historie je rovna jedné:

$$\sum_{i,j} P(w_n|w_i w_j) = 1 \tag{4.50}$$

Pokud máme případ, kdy „se vracíme“ k modelu s nižším řádem a zbylá pravděpodobnost je nulová, tak po přidání další pravděpodobnosti do rovnice dostaneme pravděpodobnost slova větší než jedna. To nám říká, že na jazykový backoff model musí být aplikována metoda odpočítávání.

Označme $\tilde{P}$ pravděpodobnost vzešlou z odpočítávání, čímž jsme uchovali nějakou pravděpodobnost pro n-gramy nižších řádů. Nechť P je označení pravděpodobnosti spočítané přímo z počtů v trénovacích datech. Rovnici (4.49) lze vyjádřit v rekurzivní formě:

$$\hat{P}(w_i|w_{i-n+1}^{i-1}) = \tilde{P}(w_i|w_{i-n+1}^{i-1}) + \theta(P(w_i|w_{i-n+1}^{i-1}))\alpha(w_{i-n+1}^{i-1})\hat{P}(w_i|w_{i-n+2}^{i-1}) \tag{4.51}$$

kde θ je binární funkce, která vybere model s nižším řádem jen pokud model vyššího řádu dává nulovou pravděpodobnost.

$$\theta(x) = \begin{cases} 1, & x = 0 \\ 0, & x \neq 0 \end{cases} \tag{4.52}$$

Pro rozdělení pravděpodobnosti nižšímu řádu n-gramů, musíme nastavit váhy α. Definujme pravděpodobnost zbylou pro n-gramy nižších řádů jako funkci β závislou

na historii w_{i-n+1}^{i-1}. Ta může být spočítána odečtením celkové pravděpodobnosti přiřazené n-gramům s danou historií od jedničky.

$$\beta(w_{i-n+1}^{i-1}) = 1 - \sum_{w_i : C(w_{i-n+1}^{i}) > 0} \tilde{P}(w_i | w_{i-n+1}^{i-1}) \tag{4.53}$$

Je to pravděpodobnost, kterou budeme rozdělovat mezi $(n-1)$-gramy. Každý $(n-1)$-gram dostane zlomek této pravděpodobnosti, takže musíme normalizovat β celkovou pravděpodobností všech $(n-1)$-gramů, kterými začíná nějaký n-gram. Tím získáme funkci α, která říká kolik pravděpodobnosti rozdělit od n-gramu pro $(n-1)$-gramy.

$$\alpha(w_{i-n+1}^{i-1}) = \frac{1 - \sum_{w_i : C(w_{i-n+1}^{i}) > 0} \tilde{P}(w_i | w_{i-n+1}^{i-1})}{1 - \sum_{w_i : C(w_{i-n+1}^{i}) > 0} \tilde{P}(w_i | w_{i-n+2}^{i-1})} \tag{4.54}$$

Funkce $\alpha(w_{i-n+1}^{i-1})$ je tzv. **backoff váha**.

Pokud dáme dohromady vzorce 4.49–4.54, dostaneme rekurzivní formuli

$$P_{bo}(w_i | w_{i-n+1}^{i-1}) = \begin{cases} d_{C(w_{i-n+1}^{i})} \frac{C(w_{i-n+1}^{i})}{\sum_{w_i \in V} C(w_{i-n+1}^{i})} & C(w_{i-n+1}^{i}) > 0 \\ \\ \alpha(w_{i-n+1}^{i-1}) P_{bo}(w_i | w_{i-n+2}^{i-1}) & C(w_{i-n+1}^{i}) = 0 \end{cases} \tag{4.55}$$

kde $d_{C(w_{i-n+1}^{i})}$ je discount koeficient z metod uvedených výše.

Hodnota $P_{bo}(w_i | w_{i-n+1}^{i-1})$ je počítána rekurzivně pro všechna n až do požadovaného řádu modelu, začíná se u unigramů. Protože není možné provádět „backoff“ z unigramů je pravděpodobnost β rozdělena rovnoměrně mezi neviděná slova.

V praxi se pro backoff modely používají metody Witten-Bell, absolute a Good-Turing discounting při nichž se ignorují n-gramy, které se vyskytly jenom jednou, jsou uvažovány, jako by se nikdy nevyskytly.

Používá se též modifikace Good-Turingova vztahu, kdy se velké počty $C(w_{i-n+1}^{i})$ považují za věrohodné a formálně se položí

$$d_{C(w_{i-n+1}^{i})} = 1 \qquad C(w_{i-n+1}^{i}) > T \tag{4.56}$$

kde T je zvolený práh. Mění se nám tím koeficient $d_{C(w_{i-n+1}^{i})}$, který je potom dán vztahem

$$d_{C(w_{i-n+1}^{i})} = \frac{\frac{C^{\star}(w_{i-n+1}^{i})}{C(w_{i-n+1}^{i})} - \frac{(T+1)n_{T+1}}{n_1}}{1 - \frac{(T+1)n_{T+1}}{n_1}} \tag{4.57}$$

Pozn.: Použití tohoto vzorce přináší někdy problémy. Většinou potřebujeme $d_r > 0$ pro všechna r, a to klade podmínky na relativní počty $n_1, n_2, \ldots, n_{T+1}$. Tato omezení bývají splněna přirozeně se vyskytujícími daty, ale uměle modifikovaná data je mohou porušit.

4.3.3 Maximální entropie

Využití maximální entropie je trochu rozdílné od metod uvedených v předešlých podkapitolách. Místo kombinací několika modelů se pokouší sestavit jeden model využitím několika informačních kanálů a z nich získat maximum možné informace.

Každý informační zdroj přináší určité podmínky a výsledný model, rozdělení pravděpodobnosti, musí splňovat všechna omezení z informačních zdrojů. Výsledkem může být mnoho pravděpodobnostních rozdělení, proto hledáme takové, které má největší entropii.

Každý informační zdroj i je asociován s výběrovou funkcí $f_i(h, w)$, která má obvykle tvar

$$f_i(h,w) = \begin{cases} 1 & \text{když se } (h,w) \text{ objevilo v trénovacích datech} \\ \\ 0 & \text{jinak} \end{cases} \tag{4.58}$$

a také požadovanou pravděpodobností d_i. Pravděpodobnost každé funkce $f_i(h, w)$ pro dané požadované rozdělení $P(h, w)$ je poté položena rovnosti s žádanou pravděpodobností d_i

$$E_p(f_i(h,w)) = \sum_{h,w} P(h,w) f_i(h,w) = d_i \tag{4.59}$$

Protože chceme model, který splňuje D+1 podmínek (d_0 je normalizační podmínka, která zaručí , že $P(h, w)$ je pravděpodobnostní rozdělení – proto definujeme $f_0(h, w) = 1$ pro všechny (h, w) a $d_0 = 1$) ve vzorci 4.59 a má maximální možnou entropii $H(P(h, w))$. Použitím metody Langrangeových multiplikátorů minimalizujeme funkci

$$\begin{gathered} L(P(h,w)) = H(P(h,w)) - \textstyle\sum_{i=0}^{D} \lambda_i (E_p(f_i(h,w)) - d_i) = \\ = \textstyle\sum_{h,w} P(h,w) \ln P(h,w) - \sum_{i=0}^{D} \lambda_i ((\sum_{h,w} P(h,w) f_i(h,w)) - d_i) \end{gathered} \tag{4.60}$$

Zderivujeme parciálně podle $P(h, w)$ a položíme rovno nule.

$$\begin{aligned} 1 + \ln P(h, w) - \textstyle\sum_{i=0}^{D} \lambda_i f_i(h, w) = 0 \\ \ln P(h, w) = \textstyle\sum_{i=0}^{D} \lambda_i f_i(h, w) + \lambda_0 - 1 \end{aligned} \tag{4.61}$$

Výsledný model je

$$P(h, w) = \frac{1}{Z} \exp(\sum_{i=1}^{D} \lambda_i f_i(h, w)) \tag{4.62}$$

kde $Z = \exp(1 - \lambda_0)$ je normalizační faktor. Pro nalezení hodnot λ_i je používán generalized iterative scaling algoritmus.

Výhodou maximální entropie v jazykovém modelování je možnost kombinace více zdrojů do jednoho modelu. Získání modelu pomocí maximální entropie je velice výpočetně náročné a proto není příliš široce využíváno.

4.4 Jazykové modely s třídami

Dosud jsme pojednávali převážně o n-gramových modelech. Jejich jednoduchost vedla k širokému uplatnění v rozpoznávání řeči. N-gramové modely mají výhodné vlastnosti:

- Pravděpodobnosti jsou získány z dat. Čím více, tím lépe.
- Parametry jsou automaticky získány z korpusů.
- Zahrnují lokální syntaxi, sémantiku a pragmatiku.
- Velmi dobře se integrují do prohledávacích algoritmů jako Viterbi a A*
- Mnoho jazyků má velmi striktní pořádek slov.

Ovšem mají i negativní vlastnosti:

- Nepostihují omezení jazyka na delších vzdálenostech slov.
- Pro jazyky s volným pořádkem slov se příliš nehodí.

- Složitě se přizpůsobují:
  - novým slovům ve slovníku
  - jiným tématům a oborům
  - dynamickým změnám řeči
- Nedosahují kvalit člověka v:
  - identifikaci a korekci chyb rozpoznávání
  - predikci následujících slov (znaků)
- Vůbec nepostihují smysl pro porozumění řeči.

Modely založené na třídách rozdělují statistiku mezi slova stejné kategorie a jsou proto schopné zevšeobecnění slovních typů, které se nevyskytly v trénovacích datech. Rozdělení skupin slov může být provedeno například podle slovních druhů, Part-Of-Speech (zkr. POS), nebo podle významu (dny v týdnu, města, ...). Vytvořením vztahů mezi skupinami slov, kategoriemi, získáme obecnější popis jazyka a popíšeme posloupnosti slov, které se v trénovacích datech nevyskytla.

Rozdělení tříd rozlišujeme na dva typy. Slovo může být prvkem právě jedné třídy, tzv. hard clustering, nebo slovo může být součástí několika tříd, tzv. soft clustering. začlenění slov do několika tříd vychází z jazyka, kde jedno slovo může vystupovat jako několik slovní druhů.

Skupiny slov místo jednotlivých slov umožňují redukci počtu parametrů jazykového modelu. Tím se zmenší i paměťová náročnost pro ukládání jazykových modelů

Základní modely založené na třídách definují podmíněnou pravděpodobnost slova w_n založenou na historii jako součin dvou činitelů: pravděpodobnost třídy a pravděpodobnost jednotlivého slova dané třídy.

$$P(w_i|w_{i-n+1}^{i-1}) = P(w_i|c_i)P(c_n|c_{i-n+1}^{i-1}) \tag{4.63}$$

Pro modelování třídami slov je třeba vyřešit dva základní problémy. První z nich je přiřazení jednotlivých slov do tříd, tzv. *word-to-class mapping*

$$G : c_i = G(w_i) \tag{4.64}$$

které přiřazuje každé slovo do jedné nebo více kategorií c_i. Každé slovo může být součástí pouze jedné třídy, potom mluvíme o deterministickém mapování, nebo může být v několika třídách současně, tzv stochastické rozdělení. Někdy se tato rozlišení vyjadřují jako *many-to-one* a *many-to-many* rozdělení.

Druhý mapování Φ seskupuje slovní historie $h_i = w_i^{i-1}$ ekvivalenčních tříd (stavů) s_i

$$\Phi : s_i = \Phi(h_i) \tag{4.65}$$

V obecném případě, kde G i Φ jsou stochastická mapování, je pravděpodobnost slova jako

$$P(w_i|h_i) = \sum_{c_i} P(w_i|c_i)(\sum_{s_i} P(c_i|s_i)P(s_i|h_i)) \tag{4.66}$$

Nejpřirozenějším výběrem mapování tříd ekvivalence pro historii slov jsou právě předchozí třídy slov. Je to analogie n-gramového modelu slov.

$$s_i = \Phi(h_i) = \{c_{i-n+1}, c_{i-n+2}, \dots c_{i-1}\} = \{c_{i-n+1}^{i-1}\} \tag{4.67}$$

s pravděpodobností $P(s_i|h_i) = 1$. Rovnice 4.66 se zjednoduší na

$$P(w_i|h_i) = \sum_{c_i} P(w_i|c_i)P(c_i|c_{i-n+1}^{i-1}) \tag{4.68}$$

A v případě deterministického rozdělení slov do tříd se pravděpodobnost slova v modelu s třídami dostaneme

$$P(w_i|h_i) = P(w_i|c_i)P(c_i|c_{i-n+1}^{i-1}) \tag{4.69}$$

Hlavním problémem zůstává funkce G, tedy přiřazení slov do tříd. Rozdělení do tříd může být provedeno manuálně, což je omezeno pouze na relativně malé úlohy, nebo automaticky.

Shlukování podle jazykových znalostí

Nejpoužívanějším typem je tzv. part-of-speech (POS) shlukování. POS se snaží zachytit syntaktické znalosti a soustředí na vztah mezi syntaktickou funkcí slov. Lze navrhnout automatický rozdělovač do tříd a s ním rozdělení provést. Má ovšem

jisté problémy. Především není stoprocentně přesný a jednotlivá slova mohou patřit do více tříd, což lze vyřešit vícenásobným členstvím ve třídách, ale někdy to nelze použít.

Deterministické sémantické shlukování

Toto shlukování do tříd je založeno na sémantické podobnosti slov. Například jména dnů, měsíců, měst se chovají podobně vzhledem k jejich kontextu. Problémem je, že tuto klasifikaci je nutno provést ručně.

Datově řízené shlukování

V datově řízeném shlukování je využíváno velkého množství dat k automatickému odvození tříd pomocí statistiky. Snahou je najít předpis pro přiřazení do tříd maximalizací věrohodnostní funkce. Původní a nejpoužívanější algoritmus je popsán v [2]. Předpokládejme, že máme bigramový model s třídami ve tvaru (4.69),

Cílem je najít word-to-class mapování $G : c_i = G(w_i)$, které maximalizuje věrohodnostní funkci

$$L(G) = \frac{1}{N}\sum_{i=1}^{N} \ln P(w_i|c_i)P(c_i|c_{i-n+1}^{i-1}) \tag{4.70}$$

Tuto funkci můžeme upravit následovně

$$L(G) = \frac{1}{N}\sum_{i=1}^{N} \ln P(w_i|c_i)P(c_i|c_{i-n+1}^{i-1}) = \tag{4.71}$$

$$= \frac{1}{N}\sum_{i=1}^{N} \ln P(w_i|c_i)P(c_i)\frac{P(c_i|c_{i-1})}{P(c_i)} =$$

$$= \frac{1}{N}\sum_{i=1}^{N} \ln P(w_i, c_i) + \frac{1}{N}\sum_{i=1}^{N} \ln \frac{P(c_i|c_{i-1})}{P(c_i)} =$$

$$= \frac{1}{N}\sum_{i=1}^{N} \ln P(w_i) + \frac{1}{N}\sum_{i=1}^{N} \ln \frac{P(c_i|c_{i-1})P(c_{i-1})}{P(c_{i-1})P(c_i)} =$$

$$= -H(w) + \frac{1}{N}\sum_{i=1}^{N} \ln \frac{P(c_i, c_{i-1})}{P(c_{i-1})P(c_i)} =$$

$$= -H(w) + \frac{1}{N} \sum_{c_1,c_2 \in C} P(c_1, c_2) \ln \frac{P(c_1, c_2)}{P(c_1)P(c_2)} =$$
$$= -H(w) + I(c_1, c_2)$$

kde $H(w)$ je entropie unigramového rozložení slov a $I(c_1, c_2)$ je tzv. průměrná mutual information dvou tříd. Protože $H(w)$ nezávisí na rozdělení tříd G, je maximalizace věrohodnostní funkce $L(G)$ stejná jako maximalizace průměrné mutual information dvou tříd. Bohužel neexistuje žádná metoda pro maximalizaci $I(c_1, c_2)$ přes všechna možná rozdělení. Brown aj. [2] představil algoritmus, který se snaží minimalizovat ztrátu mutual information v průběhu spojování.

Další možností kromě minimalizace ztráty průměrné mutual information je minimalizace vzrůstu podmíněné entropie. Metrika pro minimalizaci je následující

$$H = -\frac{1}{N} \sum_{c_1,c_2 \in C} P(c_1, c_2) \ln P(c_1|c_2) \tag{4.72}$$

4.4.1 Predictive clustering

Prediktivni rozdělení využívá trochu odlišný model odhadu pravděpodobnosti slov. Zatímco v třídách slovních n-gramů se používá pro odhad slova v závislosti na historii h_i posloupnost slovních tříd (4.69), tj.

$$P(w_i|h_i) = P(w_i|c_i)P(c_i|c_{i-n+1}^{i-1}) \tag{4.73}$$

pro predictive clustering je tvar

$$P(w_i|h_i) = P(w_i|h_i c_i)P(c_i|h_i) \tag{4.74}$$

Historie h_i může být rozdělena různě pro dvě různé třídy. Tímto způsobem má model více parametrů a je tedy větší než model postavený na slovech. Velikost je další důležitá vlastnost u modelů, viz další kapitoly. V [8] byla tato metoda spojena s prořezáváním a uvedeny byly dobré výsledky v poměru perplexita/velikost.

4.4.2 Phrase class n-gramy

Phrase class n-gram modely jsou speciálním druhem jazykového modelu, který je konstruován na základě automatického odvození bezkontextové gramatiky z textu.

Algoritmus byl navržen pro systém ATIS [32], který je jazykově omezený na prostředí letecké dopravy. Algoritmus má dvě fáze. Na počátku jsou neoznačené trénovací věty a první fáze automaticky odvodí bezkontextovou gramatiku pro modelování struktury tréninkových vět. Druhá fáze je integrace této gramatiky do phrase class n-gram (PCNG) formalismu. Pro přiřazení pravděpodobnosti testovacích vět. PCNG model má dvě komponenty – stochastickou bezkontextovou gramatiku a n-gramovou složku. Gramatika má následující strukturu. Pro každou větu má jedno pravidlo $S \Rightarrow s_i$, kde s_i jsou jednotlivé trénovací věty. Nazývají se *pravidla vět*.Pravidla vět obsahují na pravé straně jen slova. Gramatika tato slova nahrazuje neterminálními symboly, které představují jednotky jako třídy slov nebo fráze. *Fráze* obsahuje sekvenci slov a/nebo neterminálů. Každý neterminál je definován frázovým pravidlem $NT_i \Rightarrow x_1|x_2|\ldots|x_N$, kde x_i je jedna z alternativ na pravé straně, sekvence slov a/nebo neterminálů.

Každé pravidlo z gramatiky má přiřazenou pravděpodobnost odpovídající pravděpodobnosti, že se pravidlo aplikuje pro daný neterminální symbol na levé straně. Vždy při použití pravidla (nepoužívají se *pravidla vět*) k redukci věty je pravděpodobnost vynásobena pro získání tzv. síťové prostorové pravděpodobnosti věty. Sada pravidel není kompletní, a proto výsledkem je kombinace slov a neterminálních symbolů. Na tento řetězec je použit n-gramový model pro přiřazení dočasné pravděpodobnosti věty. Pravděpodobnost pravidla se získává standardním maximálním odhadem pravděpodobnosti.

$$P(NT_i \Rightarrow \alpha) = \frac{C(NT_i \Rightarrow \alpha)}{\sum_{NT_i \Rightarrow \beta} C(NT_i \Rightarrow \beta)} \tag{4.75}$$

kde α a β označují sekvenci slov a neterminálů. Standardní n-gramový model je poté trénován na redukovaných tvarech trénovacích vět.

Pokud neobsahuje gramatika žádná pravidla, PCNG model je jednoduchý n-gramový model. Když je gramatika zapsána, tak že je každá věta kompletně rozložena, dostaneme tzv. SCFG model (stochastická bezkontextová gramatika) [14]. Poslední zajímavá gramatika je s pravidly, která mají na pravá straně vždy jeden symbol – slovo nebo neterminál. Potom je model ekvivalentem n-gramového modelu s třídami.

Vylepšené class phrase modely byly představeny v [37]. Jejich použití již není omezeno na malé úlohy jako ATIS a jsou optimalizované pro trigramy.

4.5 Modelování delších vzdáleností mezi slovy

Modely založené na n-gramech mají typickou vlastnost, že postihují vztahy pouze mezi velmi blízkými slovy do hodnoty n. Je zřejmé, že člověku při porozumění textu značně pomáhá znalost vyslovených slov nejen těsně předcházejících, ale i více vzdálených. Většinou se bloky textu týkají stejného tématu a snaha zahrnout do pravděpodobnostních modelů delší vzdálenost slov než jen těsně sousedící je odůvodnitelné. Definujme jednoduchý model kombinací komponent
$LM_1, LM_2, \ldots, LM_k$.

$$P(w_i|w_1^{n-1}) = \sum_{j=1}^{k} \lambda_j P_{LM_j}(w_i|w_1^{n-1}) \tag{4.76}$$

kde λ_j jsou interpolační váhy splňující

$$\sum_{j=1}^{k} \lambda_j = 1 \tag{4.77}$$

Tento model je obecným zápisem cache a trigger modelů, jejichž popis následuje.

4.5.1 Cache modely

Cache modely jsou založeny na myšlence, že slova vyskytující se v dokumentu mají větší pravděpodobnost, že se v něm vyskytnou několikrát. Je to obdoba cache pamětí v počítačích. Krátká historie slov je zaznamenávána a těmto slovům je přiřazena větší pravděpodobnost.

Obvykle je cache model kombinován lineární interpolací s n-gramovým modelem.

$$P(w_i|w_1^{n-1}) = \lambda P_{cache}(w_i|w_1^{n-1}) + (1-\lambda) P_{n-gram}(w_i|w_{i-n+1}^{i-1}) \tag{4.78}$$

Cache je realizována jako historie slov o délce K slov. Obvykle dosahuje délky několika stovek. Pravděpodobnost P_{cache} je obvykle počítána jako n-gramová z textu v cache. Často se používá unigramová pravděpodobnost.

$$P_{cache}(w_i|w_1^{n-1}) = \frac{N_{cache}(w_i)}{K} \tag{4.79}$$

kde $N_{cache}(w_i)$ je počet výskytů slova w_i v cache textu.

Původně [15] byl cache model interpolován s trigramovým modelem založeným na třídách podle part-of-speech značek. Váha λ byla počítána pro každou značku zvlášť. Dále bylo navrženo několik dalších rozšíření. Největší přínos byl dosažen při interpolaci cache modelu s trigramovým modelem slov, nikoliv modelem s třídami.

Pravděpodobnost cache modelu nemusí být počítána jen z izolovaných slov jako unigramová, ale lze využít i n-gramy vyšších řádů. Ovšem počítání vyšších řádů naráží na nedostatečnost dat způsobenou jen krátkou historií cache.

Další rozšíření bylo provedeno myšlenkou, že následující slova jsou více ovlivněna slovy z blízké historie než-li slovy vzdálenějšími. Pravděpodobnost je tedy exponenciálně snižována pro slova, která jsou v historii více vzdálena od slova, které aktuálně předpovídáme.

Cache modely výrazně snižují perplexitu standardních modelů a navržená rozšíření přinesla další zlepšení. Bohužel tato zlepšení již nelze pozorovat u výsledků rozpoznávání řeči, které se zlepšily jen velmi nevýrazně. Pravděpodobně je to způsobeno tím, že pokud má slovo zvýšit svoji pravděpodobnost musí být správně rozpoznáno při prvním výskytu. Ovšem pokud je slovo rozpoznáno bez příspěvku cache modelu, lze očekávat, že bude rozpoznáno i v budoucnosti a cache model vlastně ztrácí smysl.

4.5.2 Trigger modely

Trigger modely využívají dočasnou historii slov stejně jako cache modely. Narozdíl od cache modelů jenom některá slova se využívají pro cache. Používají se tzv. trigger páry. Jestliže je sekvence A dobrře korelována se sekvencí B potom $(A \rightarrow B)$ je označen jako *trigger pár*, kde A je *trigger* (spouštěč) a B *triggered sequence* (spouštená sekvence). Pokud se A objevilo v historii, zvyšuje pravděpodobnost výskytu B

Pro nalezení nejvíce "užitečných"párů je třeba použít algoritmus. Oproti bigramům vyskytujícím se v textu, jejichž počet je jen malý zlomek $|V|^2$, počet párů, které se vyskytly ve stejném dokumentu je již celkem významná podmnožina $|V|^2$. Pro nalezení párů $(A \rightarrow B)$ se využívá tzv. mutual information, která ocení očekávaný

zisk A při předpovídání B

$$I(A,B) = P(A,B)\ln\frac{P(A,B)}{P(A)P(B)} + P(A,\bar{B})\ln\frac{P(A,\bar{B})}{P(A)P(\bar{B})} + \\ + P(\bar{A},B)\ln\frac{P(\bar{A},B)}{P(\bar{A})P(B)} + P(\bar{A},\bar{B})\ln\frac{P(\bar{A},\bar{B})}{P(\bar{A})P(\bar{B})} \tag{4.80}$$

Bylo ukázáno, že self-triggery, tedy páry, kde slovo predikuje samo sebe, a páry, ve kterých jsou slova se stejným kořenem jsou robustní a efektivní. Modely jsou používány v kombinaci v n-gramovými modely buď lineární interpolací nebo častěji využitím maximální entropie.

4.5.3 Skip n-gramové modely

Modelování delších historií a závislostí mezi slovy přineslo mírné zlepšení v rozpoznávacích systémech. Velké srovnání a jejich kombinaci těchto technik je diskutováno v [7] Zlepšení je prezentováno v perplexitě jednotlivých modelů ovšem už není dosahováno zlepšení v chybovosti rozpoznávacích systémů.

4.6 Velikost n-gramových modelů a její redukce

Pro zlepšování kvality n-gramových jazykových modelů se v praktických úlohách využívají dvě metody. První je použití n-gramů vyšších řádů a druhá zvýšení objemu trénovacích dat z různých zdrojů. Obě metody mají za následek velký nárůst počtu parametrů modelu a celková velikost modelu je velký problém v reálných aplikacích. Potřeba omezit počet parametrů vedla k návrhu několika metod, které se snaží minimalizovat parametry modelu při zachování jeho kvality. Prvním pokusem byl model s proměnnou délkou n-gramů [18], a poté byly navrženy postupy založené na teorii informace. Metody by měly splňovat tři základní podmínky

1. **Bezvadnost:** Kritérium by mělo optimalizovat správnou míru z informační teorie, která hodnotí kvalitu modelu.

2. **Efektivnost:** Algoritmus by měl být dostatečně rychlý. tj. úměrně k počtu n-gramů.

3. **Samoobsažitelnost:** Prořezávaní by mělo vycházet pouze z informací v původním jazykovém modelu.

Obvykle n-gramový jazykový model reprezentuje pravděpodobnostní rozdělení slov w podmíněné posloupností $n-1$ slov, neboli historií h. Ovšem jen malá část n-gramů (w,h) má v modelu uvedenu pravděpodobnost přímo v modelu. Zbytek n-gramů má přiřazenou pravděpodobnost rekurzivním backoff pravidlem

$$P(w|h) = \alpha(h)p(w|h') \tag{4.81}$$

kde h' je historie h zkrácená o první slovo (to nejvzdálenější od w) a $\alpha(h)$ je backoff váha přidružená k historii h, nastavené tak, aby $\sum_w p(w|h) = 1$.

Cílem prořezávání je odstranit z modelu odhady $p(w|h)$ pro zmenšení počtu parametrů při současné minimální ztrátě kvality modelu. Vyřazením části n-gramů se nezmění pravděpodobnost zbývajících, ale musí být přepočítány backoff váhy. Z toho důvodu je prořezávání nezávislé na zvolené vyhlazovací metodě.

Pro porovnání modelů se nejčastěji používá *relativní entropie* nebo *Kullback-Lieblerova vzdálenost*. Relativní entropie mezi dvěma modely je

$$D(p||p') = \sum_{w_i,h_j} p(w_i,h_j)[\log p'(w_i|h_j) - \log p(w_i|h_j)] \tag{4.82}$$

kde $p(\cdot|\cdot)$ je podmíněná pravděpodobnost původního modelu a $p'(\cdot|\cdot)$ pravděpodobnost prořezaného modelu.

Nyní minimalizací $D(p||p')$ vybereme n-gramy pro prořezání. Při předpokladu, že jednotlivé n-gramy ovlivňují relativní entropii nezávisle, můžeme spočítat $D(p||p')$ pro každý n-gram zvlášť. Potom lze vybrat takové n-gramy, které minimálně zvýšili relativní entropii.

Výpočetní náročnost se dá zjednodušit, pokud použijeme vzorců pro perplexitu. Perplexita původního modelu

$$PP = e^{\sum_{h,w} p(w|h) \log p(w|h)} \tag{4.83}$$

a perplexita prořezaného modelu s původním rozdělením pravděpodobnosti

$$PP' = e^{\sum_{h,w} p(w|h) \log p'(w|h)} \tag{4.84}$$

Relativní změna perplexity prořezaného modelu a původního modelu jako relativní entropie

$$\frac{PP' - PP}{PP} = e^{D(p||p')} - 1 \tag{4.85}$$

Výpočet prořezávání je pak velmi jednoduchý. Spočítáme přírůstek relativní entropie při odstranění jednotlivých n-gramů a odstraníme z modelu takové, u kterých změna přesáhla zvolený práh θ. Ve výsledném modelu přepočítáme backoff váhy. Přesný algoritmus lze nalézt v [40].

Byla navrženy i jiná kritéria pro prořezávání. Nejjednodušším je tzv. *count cutoff*. Metoda, která počítá pravděpodobnosti jazykového modelu jen s některými n-gramy. Například při zvoleném prahu 2 jsou uvažovány jen n-gramy, které se vyskytly v datech alespoň třikrát. Nejsou vůbec zohledněny n-gramy z dat s výskytem jedenkrát nebo dvakrát.

Následně byl uveden vážený rozdíl (weighted difference) [39]

$$N(w,h)[\log p(w|h) - \log p'(w|h)] \tag{4.86}$$

kde $N(w,h)$ je počet kolikrát byl n-gram (w,h) v trénovacích datech. Případně podobný tvar

$$p(w,h)[\log p(w|h) - \log p'(w|h)] \tag{4.87}$$

, kde $p(w,h)$ je vyhlazená pravděpodobnost v původním modelu. Je to obdoba relativní entropie, kde se ovšem nezahrnuje vliv jiných n-gramů než jednoho zkoumaného. tedy změny v backoff váhách $\alpha(h)$

Dalším zajímavým kritériem je tzv. **Rank**. Základem je *rank* slova w, který je definován jako pozice slova v seřazeném seznamu n-gramových pravděpodobností $p(w|h)$, kde $w \in V$ a V je slovník. Nejvíce pravděpodobné slovo má hodnotu jedna a nejméně pravděpodobné $|V|$, tedy rovnu velikosti slovníku. Ztrátová funkce vypadá takto

$$\sum_{w_i,h_j} \{\log[R'(w_i|h_j) + k] - \log R(w_i|h_j)\} \tag{4.88}$$

kde $R(w_i|h_j)$ je rank pozorovaného bigramu $P(w_i|h_j)$ před prořezáváním a $R'(w_i|h_j)$ je rank po prořezávání a sumace je přes všechny n-gramy (w_i,h_j). Konstanta k zaručí, že $\log[R'(w_i|h_j) + k] - \log R(w_i|h_j) \neq 0$. Kritéria lze spolu kombinovat pro vylepšení

výsledků, viz [6]. Lze nalézt i další podoby ztrátové funkce, např. ditribution based pruning [5].

4.6.1 Jazykové modely založené na neuronových sítích

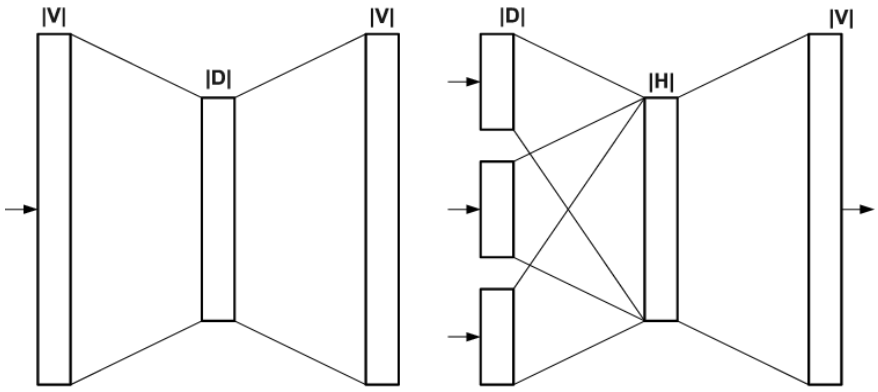
Jazykové n-gramové modely jsou svými vlastnostmi částečně omezeny na modelování krátkých historií slov a ustálený pořádek slov ve větě. Modelování jazyka pomocí neuronových sítí představil [1] a první výsledky s rozpoznáváním řeči publikoval [38]. Struktura modelování pomocí neuronových sítí je obvykle dělená na trénování dvou sítí. Nejdříve kroku se trénuje bigramová neuronová síť, kde pro dané slovo w ze slovníku V se odhaduje pravděpodobnostní rozdělení následujícího slova v textu. Obvykle je používána neuronová síť se vstupní a výstupní vrstvou velikosti $|V|$ a jednou skrytou vrstvou s několika desítkami neuronů. Výstupní vrstva obvykle používá aktivační funkci *softmax*, která zajistí, že suma výstupů neuronů je v sumě rovna jedné.

Pro modelování delších závislostí mezi slovy než bigram se nepoužívá na vstupu $(n-1)*|V|$ neuronů, ale využívá se natrénovaná skrytá vrstva z bigramového modelu první neuronové sítě. Skrytá vrstva je projekcí slova ze slovníku V do spojité dimenze $|D|$. Spojením více těchto projekcí získáme historii slov podobně jako u n-gramového modelu. Výstupní vrstva je opět o velikosti $|V|$ se softmax aktivační funkcí. Obrázek 4.1 popisuje tuto strukturu.

Náročnost použití neuronových sítí není jenom v trénování relativně velkých sítí, ale i implementace do rozpoznávacích systémů. Nejvíce jsou tyto modely používané pro reskórování výsledků získaných při rozpoznávání se standardním n-gramovým modelem.

4.6.2 Jazykové modely založené na rekurzivních neuronových sítích

Rozšířenou variantou použití neuronových sítí jsou rekurzivní sítě. Nevýhodou dopředných neuronových sítí je omezenost kontextu, který je pevně nastaven ve vstupní vrstvě sítě před začátkem trénování. Je to podobné omezení jako n-gramové



Obrázek 4.1: Schéma neuronové sítě pro jazykové modelování

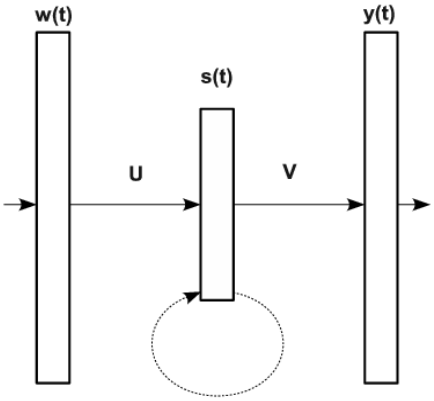
modely s omezenou délkou historie. Rekurzivní sítě takové omezení nemají a kontext udržený v sítí může být libovolný. Pro jazykové modely byla použita nejjednodušší varianta rekurentní sítě s jednoduchou implementací a trénováním. Síť se vstupní vrstvou x, skrytou vrstvou s (nazývanou též kontextová nebo stavová vrstva) a výstupní vrstvou y je zobrazena na obrázku 4.2. Vstupní vektor $x(t)$ je je tvořen spojením vektoru w, který reprezentuje současné slovo a výstupem neuronů skryté vrstvy v čase $t-1$.

$$x(t) = w(t) + s(t-1) \tag{4.89}$$

$$s_j(t) = f\left(\sum_i x_i(t) u_j i)\right) \tag{4.90}$$

$$y_k(t) = g\left(\sum_j s_j(t) v_k j)\right) \tag{4.91}$$

kde aktivační funkce sigmoid $f(z)$ ve skryté a softmax $g(z)$ ve výstupní vrstvě jsou:



Obrázek 4.2: Schéma rekurentní neuronové sítě pro jazykové modelování

$$f(z) = \frac{1}{1 + e^{-z}}, \qquad g(z_m) = \frac{e^{z_m}}{\sum_k e^{z_k}} \tag{4.92}$$

Pro trénování se používá standardní algoritmus backpropagation. K optimalizaci výpočtu se používá nastavení pravděpodobnosti slov, které se nevyskytují v textu častěji než zvolený práh, společná pravděpodobnost pro tato slova.

Experimenty s rekurentními neuronovými sítěmi prezentoval [26], [27] a [19]. Bylo dosaženo výrazného snížení hodnoty perplexity a zvýšené přesnosti rozpoznávání oproti n-gramovým modelům. Navíc lineární interpolací s n-gramovými modely bylo dosaženo nejlepších výsledků.

Kapitola 5

Jazykové modelování češtiny

Čeština patří do skupiny slovanských jazyků, kde jsou například i ruština, polština, slovenština a další. Tato skupina jazyků se vyznačuje určitými znaky, odlišnými od germánských a románských, které způsobují problémy při jazykovém modelování.

- Vysoká míra flexe, tedy skloňování, časování a ohýbání slov. Pro jednotlivé sloveso je v českém jazyce až tři sta odvozených tvarů, pro přídavná jméno až dvě stovky a podstatné jméno dvacet.
- Používání odvozených slov, tzv. derivace, pomocí předpon a přípon.
- Volný pořádek slov ve větě.

Tyto tři důvody mají za následek dvě problematické oblasti při modelování

1. Velikost slovníku $|V|$ roste velice rychle při větších trénovacích datech. Je to dáno tím, ze v rozpoznávání řeči se uvažuje každý tvar slova, jako unikátní výraz ve slovníku. Proto flexe a derivace slov ve slovanských jazycích způsobuje velký rozsah slovníků. Systémy pro rozpoznávání řeči jsou většinou omezené na určitou velikost slovníku z důvodu výpočetní náročnosti, z toho důvodu slovník nedokáže dostatečně pokrýt reálná data a způsobuje vysoký poměr OOV slov.
2. Volný pořádek slov ve větě snižuje vhodnost použití n-gramových modelů, v nichž se modeluje právě pořadí slov, jak jsou za sebou.

Pořádek slov ve větě a horší schopnost n-gramů postihnout tuto skutečnost je z pohledu velkého *OOV* méně důležitou součástí a proto se většina snah o zlepšení modelů ubírá směrem snižování *OOV rate*. Zvětšením slovníku v rozpoznávacích systémech lze redukovat míru OOV, ovšem ani maximalizace množství dostupných trénovacích dat nezaručuje nízkou a v žádném případě nulovou hodnotu OOV.

5.1 Snižování OOV v jazykových modelech

5.1.1 Morfémový jazykový model

Pro rozpoznávání s velkým slovníkem byly představeny tzv. *morpheme-based* jazykové modely, které jako základní modelovací jednotku používají části slov - morfémy. Ověřeným dělením slov je na kmen slova a koncovku. Kmen slova je ekvivalentem lemmatu, které nese hlavní část informace o slovu a není změněn pravidly odvozování. Počet koncovek je relativně malý (v češtině asi 700) a velikost slovníku příliš nezvýší oproti původnímu slovníku slov.

V základním přístupu morpheme-based modelů jsou všechny části brány jako ekvivalent slov. Například trigramová pravděpodobnost i-tého morfému m_i je

$$P(m_i|h_i) = P(m_i|m_{i-2}m_{i-1}) \tag{5.1}$$

Tento model při použití kmenů a koncovek odhaduje pravděpodobnosti dvěma různými cestami. Pokud je morfém i koncovka, tak její predikce je postavena na odpovídajícím kmenu a předchozí koncovce. Tento způsob přesně odpovídá češtině, kde ke kmenu přísluší jen omezená část koncovek a po sobě jdoucí slova dodržují shodu v rodu, číslu a pádu. Druhým případem je predikce kmenu slova, která závisí na předchozí koncovce a předchozím slovu. Ve skutečnosti jde o bigram a závislost kmen-kmen, protože koncovka neovlivní následující kmen. bigramový model má omezenou možnost predikce oproti trigramovému a proto se použila následující modifikace morfémového modelu

Modifikovaný morfémový model rozlišuje, zda je predikované slovo koncovka nebo kmen slova. Pro kmen slova se používá

$$P(s_i|h_i) = P(s_i|s_{i-2}s_{i-1}) \tag{5.2}$$

, tedy predikce kmenu je závislá na předchozích kmenech. To odpovídá skutečné trigramové pravděpodobnosti slov. Vzorec pro koncovky je stejný jako v nemodifikované verzi

$$P(e_i|h_i) = P(e_i|e_{i-1}s_i) \tag{5.3}$$

5.1.2 Jazykový model s třídami.

5.1.2.1 Automaticky odvozené třídy - POS tagging

Při modelování pomocí tříd slov pro inflektivní jazyky se nejvíce používají automaticky vytvořené třídy pomocí Part-Of-Speech. značek. Je to ekvivalent slovních druhů, ovšem počet tříd je podstatně větší. Pro češtinu se používají poziční POS značky [10, 9]. Značkování, tzv. Part-Of-Speech tagging, je přiřazení ke každému slovu jedné (nebo více) gramatické značky. Přiřazení značky není jednoznačné a vybrání správné značky je problém. Nejlepší taggery pro angličtinu dosahují úspěšnosti až 98%. V češtině. která je značně typograficky odlišná je nejlepší úspěšnost kolem 94%.

Rozpoznávání češtiny s modelem tříd na základě POS značkování bylo zkoumáno v [3]. Jazykový model byl vytvořen z velkého korpusu a otagován stochastickým taggerem. Rozpoznávání provedli na systému vyvinutém na Technické univerzitě v Liberci [30] Další práce [13] s modely založenými na POS třídách se snažila ověřit vliv použití tříd na zlepšení rozpoznávání. Nejprve byla řeč rozpoznána n-gramovým modelem, kde výstupem byla mřížka pravděpodobných slov, která byla následně reskórována modelem založeným na třídách. Přepočet novým modelem přinesl snížení chybovosti. Třídy jazykového modelu byly odvozeny automaticky. Byl proveden i test, kdy byl text označkován ručně a následně vytvořen model s třídami z těchto značek. Tento model nepřinesl zlepšení jako automaticky otagovaný text.

5.1.2.2 Sémantické třídy

Dalším způsobem vytvoření tříd je možnost využití tříd pro slova s podobným významem. Například křestní jména, příjmení, názvy dnů v týdnu, měsíců v roce, měst, států a další. Nalezení těchto slov v korpusu nelze jednoduše algoritmizovat. Přiřazení do tříd se proto provádí převážně ručně a lze ho udělat jen u relativně

dec.	**AT&T**		**ERIS**			
	Corr	Acc	Corr	Acc	OOV rate	PPL
bigramy	69,23 %	58,27 %	64,71 %	55,73 %	7,4 %	587
třída jmen	71,74 %	63,07 %	67,02 %	60,61 %	4,3 %	587
třídy tvarů jmen	74,12 %	65,78 %	69,48 %	63,66 %	4,4 %	446
národnosti	74,39 %	65,92 %	69,75 %	64,15 %	4,3 %	458

Tabulka 5.1: Výsledky rozpoznávání hokejového komentáře

	BLM		**ALM**	
test	OOVr	WER jmen	OOVr	WER jmen
1	2,35 %	4,17 %	2,35 %	4,17 %
2	2,50 %	20,00 %	2,50 %	0,00 %
3	2,46 %	16,13 %	2,46 %	12,90 %
4	3,51 %	16,95 %	3,42 %	5,08 %
5	0,47 %	0,00 %	0,47 %	0,00 %
1-5	**2,39 %**	**17,68 %**	**2,37 %**	**5,13 %**

Tabulka 5.2: Výsledky rozpoznávání jmen osob v záznamu schůze PSP ČR

malých úloh.

Třídy jmen a příjmení byly prezentovány v pracích [11, 34, 36] , které za zabývají rozpoznáváním komentářů hokejových utkání. Nejprve byly vytvořeny třídy jmen, jejichž cílem bylo hlavně snížit vysoký *OOV rate* a později byly třídy rozděleny podle gramatických vlastností na menší, což výrazně zlepšilo úspěšnost rozpoznávání, zvláště u sledované skupiny jmen a příjmení.

V tabulce 5.1 výsledky experimentů s rozpoznáváním komentáře hokejových zápasů. K rozpoznávání byly použity dekodéry *AT&T* a *ERIS*. V jednotlivých řádcích jsou vidět výsledky nejprve pro bigramový jazykový model. Při použití jedné třídy jmen pro všechna jména, která byla vytvořena ze soupisky jednotlivých rozpoznávaných zápasů. Tím se výrazně snižila *OOV rate* a zlepšila se správnost rozpoznávání. Dalším krokem bylo rozdělení třídy jmen podle gramatických kategorií, což přineslo zpřesnění rozpoznávání jmen ve správných tvarech. Posledním krokem bylo vytvoření podobných tříd jako u jmen pro státy a národnosti.

Podobný postup byl také nasazen pro rozpoznávání schůzí Poslanecké sněmovny Parlamentu České Republiky. Na pěti různých záznamech byl zkoumán vliv na úspěšnost rozpoznávání jmen při použití tříd slov pro jména podle gramatických kategorií, podobně jako v hokejových zápasech. V tabulce 5.2 jsou uvedeny výsledky rozpoznávání pro sledovanou skupinu jmen s se základním bigramovým modelem – *BLM* a modelem s třídami jmen – *ALM*. Průměrná perplexita testů byla 294. Celkové Accuracy a Correctness rozpoznávání s *ALM* modelem bylo 84,17 % a 88,17 %.

Kapitola 6

Úloha a návrh řešení

Systémy rozpoznávání řeči převádějící mluvený projev do textové podoby jsou užívány v širokém spektru úloh. Dále se nebudeme zabývat zařízeními, které zpracovávají hlasové příkazy z omezené množiny povelů. Systémy pro zpracování souvislého mluveného projevu lze rozdělit například na skupiny:

- Hledání klíčových slov
- Indexování archivů
- Titulkování
- Diktovací systémy

V této práci se budeme zabývat systémy pro diktování textu, které mají svá specifika a potřeby kvalitního výstupu. V mnoha oborech lidské činnosti se pořizují psané záznamy, které vznikají přímým zápisem diktovaného textu zapisujícím osobám. Úkolem těchto osob je přesný záznam diktovaného textu do podoby psané zprávy. Systém rozpoznávání řeči pro diktování by měl být schopen nahradit tuto činnost a zautomatizovat záznam mluveného projevu.

Výstup z diktovacího systému je omezen vlastnostmi diktovacího systému, které lze rozdělit na akustické a jazykové. Akustické podmínky jako správná výslovnost bez vad řeči, tiché prostředí nebo prostředí akusticky znečištěné apod. jsou záležitosti, kterými se zabývají metody akustického modelování, kterými se zabývat v této práci

nebudeme. Z jazykového pohledu jde zejména o správný zápis textu bez chybného rozpoznání jednotlivých slov a podle pravidel pravopisu. Systém rozpoznávání řeči má omezený slovník a nelze očekávat, že rozpozná správně libovolné vyslovené slovo. Způsob vytváření jazykových modelů z trénovacích textů zaručuje, že slovní zásoba všeobecně používaná při diktování a tvorbě větných celků bude systémem zpracovávána v dobré kvalitě. Omezení nastává při diktování slov, která se v trénovacích textech neobjevila nebo je jejich zastoupení obecně velmi nízké. Obvykle jde o jmenné entity, jako jsou jména osob, geografické názvy - oblasti, sídla, místní názvy.

Budeme se zabývat zejména možnostmi rozšíření jazykového modelu o nová slova. Zvolili jsme rozšíření o seznam měst, obcí, osad a názvů ulic. Tato data jsou velmi rozdílná v jednotlivých regionech. Obvyklý postup tvorby jazykového modelu je zpracování dat ze stejné oblasti, pro kterou je diktovací systém navrhován. Na Katedře kybernetiky jsme řešili návrh diktovacího systému, který by mohl být využíván u soudů v České republice. Na všech úrovních soudů je vytvářeno mnoho dokumentů z různých částí soudních procesů. Při zpracování dat z několika soudů jsme po analýze dat zjistili, že data z jednotlivých soudů si jsou velmi podobná, ale přesto je nelze použít pro vytvoření univerzálního modelu. Jazykové modely se lišily nejvíce právě ve jmenných entitách zmíněných výše. Každý regionální soud se zabývá nejvíce množinou jmenných entit a jazykové modely nejsou univerzálně přenositelné. Nejvíce rozdílnou množinou slov byly místní názvy - názvy sídel a ulic.

Typický postup tvorby jazykového modelu by byl, že pro každý soud, který by měl zájem používat diktovací systém pro dokumenty a byly v něm zahrnut místní názvy obcí a ulic, by bylo potřeba zpracovat soudní rozhodnutí z každého soudu podle regionu a z těchto dat nastavit diktovací systém. Takový postup není příliš efektivní.

Výsledkem této práce bude vytvoření postupu pro úpravu jazykového modelu z dat, například jediného regionu, a použití tohoto modelu v jiném regionu s úpravou pro místní názvy. Jedním ze základních rozdílů nebude jen možnost správného rozpoznání místních názvů, ale například i v modelu pro původní region nebudou obsaženy všechny místní názvy. Získání dat, která by obsahovala kompletní množinu místních názvů není možné prakticky. Například při uvažování přejmenování ulic ve městech, výstavbě nových ulic nebo růstu aglomerací se tato data mění v běhu času.

Nejvyšší soud ČR	Nejvyšší správní soud ČR
Vrchní soud v Praze	Vrchní soud v Olomouci
Krajské soudy – 9 soudů	
Okresní soudy – 85 soudů	

Tabulka 6.1: Čtyřčlánková soustava soudů v České republice

I tento problém se pokusíme vyřešit, kdy do obecného modelu bude možnost dodat libovolnou sadu místních názvů pro rozpoznávání.

Zvýší se tím komfort pro uživatele diktovacího systému, kdy uživatel je obvykle předem informován, že v systému nejsou obsažena všechna slova, která může vyslovit a chybovost systému bude muset opravit ručně. Uživatel systému s přidanou množinou místních názvů bude mít znalost, že systém místním názvů „rozumí“, a tím se zvýší použitelnost systému. Další výhodou by měl být lepší výstup systému z pohledu českého pravopisu, kde správně rozpoznaný místní název by měl být zapsán podle pravidel českého pravopisu.

6.1 Soustava soudů v České republice

Pro experimenty jsme zvolili tématickou oblast soudních rozhodnutí. Rozhodnutí soudu je dokument, který uzavírá soudní proces a je v něm popsáno celé soudní jednání a soudce v něm odůvodňuje rozsudek. V České republice je čtyřčlánková soustava soudů, viz tabulka 6.1. Jsou to nejvyšší, vrchní, krajské a okresní soudy. Stranou těchto soudů stojí Ústavní soud, který v ústavním soudnictví rozhoduje o souladu právních předpisů i rozhodnutí s ústavou. Obecné soudnictví sestává z civilní a trestní větve. Soudní řízení v obecném soudnictví je dvoustupňové a zabývají s jím okresní a krajské soudy.

V České republice s nástupem eGovernmentu vznikl požadavek na elektronické zpracování dokumentů. Jedním požadavkem byl i vývoj diktovacích systémů pro soudce pro zápis soudních rozhodnutí. Diktovací systémy byly vyvinuty a testovány soudci a na základě jejich připomínek dále rozvíjeny. Funkčnost systémů pro obecný

text byla na dobré úrovni. Velké zlepšení bylo požadováno v diktování jmenných entit, zvláště jmen osob a místopisných názvů, jejichž možnost rozpoznávání je v diktovacím systému velmi omezená. Teoreticky bychom měli pro každý okres a kraj zpracovat korpus z dané oblasti, aby v trénovacích datech byly zastoupeny místopisné názvy. Přesto ani ve velkém korpusu bychom nezískali plný slovník všech názvů obcí, měst, osad a názvů ulic.

6.2 Současné metody pro rozšíření slovníku

V současné době neexistují metody, které by obecně řešily rozšiřování slovníku rozpoznávacího systému. Standardní postup s použitím velkého množství trénovacích dat předpokládá velké množství textů, které obsahují kompletní informaci o možném vstupu celého rozpoznávacího systému. Takové obrovské množství textu ovšem není dostupné téměř pro žádnou větší úlohu.

Kombinace více jazykových modelů

Přizpůsobení systému rozpoznávání řeči lze provést kombinací několika jazykových modelů včetně jejich různé váhy. Základní obecný model může být vytvořen z velkého množství dat. Doplňující modely pak mohou přinášet právě rozšíření slovníku o nové tvary. Tyto modely mohou být sestavovány z menšího objemu dat a zle je tedy vhodně obměňovat dle potřeb. Problematické je zvláště správné nastavení váhy modelů a získávání dat pro modely.

Jazykové modely s POS třídami

Označení slovních druhů a využití této informace pro kvalitnější jazykové modely je další možnost. Označování slov POS tagy – slovní druhy – není dokonalé a kombinace modelu postaveného na slovech a modelu s třídami slovních druhů je problematické. Přesto v některých úlohách se tento postup osvědčil a byl prezentován.

Modely subslovních jednotek

Byly prezentovány systémy, které kombinují jazykový model postavený na slovech s modelem, který obsahuje části slov. Části mohou být krátké slabiky nebo předpony, kořeny a přípony. Tyto metody byly využity u flektivních jazyků jakým je například čeština. Rozdělením slov do jednotek se ztrácí informace, kterou mají n-gramové modely a to mezislovní vazba. V trigramovém modelu jsou uloženy pravděpodobnosti až trojic po sobě následujících slov. Pokud slova rozdělíme na dvě jednotky sníží se vazba mezi slovy pouze na jedno slovo a jednu koncovku nebo začátek slova.

Návrh řešení rozšíření slovníku jazykového modelu, který budeme dále prezentovat není obecný postup. Zaměříme se na skupiny slov – geografické názvy, které lze relativně dobře popsat a v trénovacích textech nelze výskyt celé množiny pozorovat.

6.3 Návrh řešení

V předchozí kapitole je zmíněno doplnění jazykového modelu o nová slova. Konkrétně jde o rozšíření jazykových modelů o třídy osobních jmen a příjmení. V pracích [11, 34, 36] byl postup použit pro aktualizaci jazykových modelů o vlastní jména, jejichž množinu bylo možné před nasazením jazykového modelu a rozpoznávače odhadnout. Druhým významným přínosem zvláště u úlohy rozpoznávání hokejových komentářů bylo, že pomocí označení jmen v trénovacích datech a jejich nahrazení novou množinou došlo ke zmenšení slovníku o jména, která by bez označení byla v jazykovém modelu. Přitom by mohlo jít o jména hráčů, která pocházela z trénovacích dat pořízených před velkou dobou. V jazykovém modelu by se přenášela jména hokejistů, kteří již ukončili kariéru. Model by se musel průběžně aktualizovat o nová data, což lze pouze novou anotací zápasů, která je velmi náročná a neřeší úplnost množiny jmen v jazykovém modelu.

V prezentované práci byla z počátku použita jedna třída jmen a uniformní pravděpodobností uvnitř třídy. Po vytvoření třídy jmen všech tvarů podle morfologických kategorií došlo k problému s velmi malou pravděpodobností všech jmen po začlenění třídy do jazykového modelu. V případě, že byla třída jmen menší a pravděpodobnosti ve třídě větší, fungoval rozpoznávací systém podle předpokladů a jména se objevila

ve výsledcích rozpoznávání. Malá třída jmen ovšem nedokázala pokrýt potřebné množství jmen. Řešením s dobrým výsledkem bylo použití více tříd, jejichž rozdělení bylo provedeno na základě morfologické kategorie. Pravděpodobnost ve třídách byla i dále rozdělena uniformně.

Myšlenka rozšíření jazykových modelů pro diktování soudních rozhodnutí o místní názvy je postavena na předchozích výsledcích. Prvním krokem by tedy mělo být v trénovacích textech označit místní názvy a následně navrhnout třídy místních názvů. Určitě bude vhodné použít rozdělení podle morfologické kategorie, ale zřejmě to bude nedostatečné, protože velikost tříd bude příliš veliká. Podívejme se na dostupná data.

6.4 Zdroje dat pro sestavení tříd

Pro vytvoření tříd obsahující kompletní seznam obcí a ulic v daném regionu je potřeba takové seznamy získat. Dostupné seznamy jsou z několika zdrojů. Jde o veřejně dostupné seznamy obcí a ulic, jak je poskytuje státní správa ČR, statistický úřad a další organizace.

Databáze obcí a ulic Ministerstva vnitra

Na webových stránkách Ministerstva vnitra jsou dostupné seznamy obcí ulic a adres ve formátu XML. Tyto seznamy jsou pravidelné aktualizovány, ale bohužel přibližně od roku 2011 už neobsahují počty obyvatel v obcích. Seznamy jsou rozděleny podle krajů a okresů a jsou označeny obce s rozšířenou působností a pod ně spadající obce. Seznam ulic a adres je rozčleněn podle obcí a je uveden seznam čísel popisných v dané ulici. Seznamy jsou pravidelně aktualizovány v průběhu roku. Seznam obcí s počtem obyvatel v nich je další registr poskytovaný Ministerstvem vnitra. V něm je uveden seznam obcí v ČR členěný stejně jako druhý seznam podle krajů a navíc s počtem obyvatel v každé obci. Počet obyvatel je aktualizovaný jednou ročně na začátku roku.

Databáze UIR-ADR Ministerstva práce a sociálních věcí

Databáze UIR-ADR vytváří ministerstvo práce a sociálních věcí ve spolupráci s obecními úřady a jde o seznam všech stavebních objektů, které mají číslo domovní. Adresy neobsahují žádné údaje o osobách ani organizacích. Česká pošta poskytuje pro

adresy platná poštovní směrovací čísla. Registr je využíván pro potřeby státní sociální podpory a úřadů práce. Za spolupráce obcí jsou průběžně doplňovány chybějící adresy, zaznamenávány změny názvů, případně označeny zrušené stavební objekty. Používání registru zajišťuje jednotné a správné psaní názvů a umožňuje kontrolu existence adresy, a tak lze zpřesnit a zrychlit doručování zásilek a zajistit další funkce závislé na přesné a platné adrese. Data jsou aktualizována.

Seznam obcí má podmnožinu měst, která budou s velkou pravděpodobností ve všechny soudních rozhodnutích velmi frekventovaná. Jsou to města, kde sídlí okresní krajské soudy. V každém bývalém okresním městě a v případě pobočky okresního soudu i jiné město, sídlí okresní soud. V krajských městech (ne ve všech) sídlí krajské soudy, ale tato města jsou obvykle shodná s lokací okresních soudů. Proto seznam okresních soudů pokrývá tuto množinu.

Seznam katastrálních úřadů

V textech soudních rozhodnutí se objevují názvy katastrálních území. Obvykle je název katastrálního území shodný s názvem obce, ale neplatí to vždy, protože názvy jednotlivých území nesmějí být stejné. Proto jsou některé lehce upravené. Obvykle je k názvu přidán název větší obce nebo města, kde obec leží, například Borek u Rokycan.

Jednotný telefonní seznam

Jednotný telefonní seznam vedený společností MEDIATEL, spol. s r. o. je obecně známý jako Zlaté stránky. Jsou v něm zveřejněna telefonní čísla zákazníků všech telefonních operátorů v ČR. Zákazníci musí dát ke zveřejnění v tomto seznamu souhlas. Z tohoto seznamu nelze získat seznam všech obcí a ulic, ale může být užitečný pro získání statistik o četnosti zastoupení obcí a ulic v soudních rozhodnutích, které se často týkají soukromých občanů. Přeneseně by se dalo uvažovat, že získáme počet telefonů na dané adrese případně v jedné ulici.

Registr ekonomických subjektů

Registr ekonomických subjektů je databáze spravovaná a vydávána Českým statistickým úřadem, která obsahuje ekonomické subjekty což je každá právnická osoba a fyzická osoba s postavením podnikatele a organizační složka státu, která je účetní jednotkou. Podobně jako u jednotného telefonního seznamu nejde o seznam

všech obcí a ulic, ale lze získat statistiky právnických osob na dané adrese, v ulici, v obci.

Seznam advokátů

Rejstřík, který udržuje Advokátní komora o svých členech, advokátech a koncipientech, včetně adres advokátních kanceláří může přinést další informace pro statistiku názvů obcí a ulic. V každém soudním řízení je advokát přítomen a jeho jméno i adresa kanceláře je uváděno v soudním rozhodnutí.

Základní registry veřejné správy

Nastupující eGovernment (elektronická veřejná správa) v České republice, přinesl vývoj takzvaných základní registrů veřejné správy, které by měly v budoucnu nahradit všechny dosavadní registry a vyřešit jejich nejednotnost, multiplicitu a neaktuálnost. Některé výše uvedené registry budou nahrazeny novým systémem. Navržená struktura základních registrů je ze čtyř částí.

- *Registr obyvatel* – obsahující základní údaje o občanech a cizincích s povolením k pobytu, mezi tyto údaje patří: jméno a příjmení, datum a místo narození a úmrtí a státní občanství.
- *Registr práv a povinností* – obsahující referenční údaje o působnosti orgánů veřejné moci, mj. oprávnění k přístupu do k jednotlivým údajům, informace o změnách provedených v těchto údajích apod.
- *Registr osob* –obsahující údaje o právnických osobách, podnikajících fyzických osobách, orgánech veřejné moci i o nekomerčních subjektech, jako jsou občanská sdružení a církve.
- *Registr územní identifikace, adres a nemovitostí* – spravující údaje o základních územních a správních prvcích.

Splnění hlavních cílů základních registrů bude velký přínos pro správu dat. Získávání strukturovaných dat ze současných registrů je velmi problematické a skutečně jsou v registrech zanesena data, která neodpovídají skutečnosti. Zpracování současných registrů bylo komplikované a obzvláště řešení vícenásobných údajů, nejednotně uvedené údaje a chybně zadaná data bylo velkým problémem.

6.5 Rozdělení pravděpodobnosti ve třídách

6.5.1 Uniformní rozdělení pravděpodobnosti ve třídách

V předchozí práci s hokejovými zápasy a parlamentními schůzemi bylo použito uvnitř tříd uniformní rozdělení pravděpodobnosti. Je zřejmé, že toto rozdělení neodpovídá skutečnému rozdělení pravděpodobnosti jednotlivých slov ve třídě. Například v hokejovém zápase nejsou na hrací ploše všichni hráči rovnoměrně nebo jeden z týmů má obvykle „více ze hry“ a najdou se i další nerovnoměrnosti v pravděpodobnostním rozložení možnosti vyslovení jména komentátorem. Mohutnost tříd v úlohách s hokejovými komentáři a parlamentních schůzí je obvykle v desítkách až maximálně kolem dvou set prvků v jedné třídě.

Seznamy místních názvů dosahují podstatně větších hodnot i pro malý region. Analogicky jako u prvních experimentů s hokejovými komentáři lze očekávat horší výsledky rozpoznání prvků tříd z důvodu velkého počtu prvků třídy z důvodu velmi malých pravděpodobností. V HMM modelech získají větší pravděpodobnost akusticky podobná slova mimo třídu. Jeden z experimentů s místními názvy provedeme s rovnoměrným rozdělením pravděpodobnosti uvnitř tříd.

6.5.2 Rozšířené rozdělení pravděpodobnosti ve třídách

Odhad lepšího rozložení pravděpodobnosti uvnitř tříd se pokusíme navrhnout z trénovacích dat a dostupných rejstříků místních názvů. Nalezení lepšího rozložení pravděpodobnosti nelze provádět na základě rozpoznávacích experimentů kvůli velké výpočetní náročnosti. Zkusme navrhnout rozložení pravděpodobnosti na základě podobnosti s rozložením pravděpodobnosti v trénovacích datech. Četnosti jednotlivých slov třídy v textu lze považovat za náhodný vektor. Pro srovnání dvou náhodných proměnných lze využít tzv. Pearsonův korelační koeficient, který pro dvě náhodné proměnné X, Y lze vypočítat jako

$$\rho = \frac{cov(X,Y)}{\sigma_X \sigma_Y} \tag{6.1}$$

kde $cov(X.Y)$ je kovariance proměnných X, Y, což je střední hodnota součinu

odchylek od středních hodnot.

$$cov(X, Y) = E\left(E(X - \mu_X)E(Y - \mu_Y)\right) \tag{6.2}$$

a σ jednotlivých proměnných je směrodatná odchylka, což je odmocnina rozptylu veličiny.

$$\sigma_X = \sqrt{E(X^2) - E^2(X)} \tag{6.3}$$

$E(.)$ označuje střední hodnotu, v našem případě aritmetický průměr.

Hodnota korelačního koeficientu může nabývat hodnot v intervalu $< -1, 1 >$, kdy nulová hodnota znamená, že veličiny jsou nezávislé. kladné hodnoty značí kladný lineární vztah veličin až po úplnou závislost při $\rho = +1$, přímá úměrnost. Opačně při záporných hodnotách zápornou lineární závislost až nepřímou úměru při $\rho = -1$.

Budeme porovnávat vektor složený z četností názvů obcí, které se vyskytly v trénovacích datech, s naším odhadem, který budeme dělat nezávisle na znalosti hodnot z trénovacích dat. Pro srovnání použijeme třídu obcí v prvním pádu. Zastoupení prvního pádu je v textu nejčastější a v ostatních lze očekávat stejné rozložení pravděpodobnosti.

První z logických úvah při odpovídání na otázku „*Jaká města a obce se budou nejčastěji vyskytovat v soudních rozhodnutích?*“ bude, že půjde o „*větší města a obce*“. Tato myšlenka se dá zřejmě vyjádřit podle počtu obyvatel. Další rozvíjení myšlenky by mohlo být, že nejčastěji se objeví názvy okresních soudů a tedy i bývalých okresních měst a z těchto měst nejčastěji město, kde je sídlo soudu, kde se píše dané rozhodnutí. Názvy katastrálních území nejsou stejné jako názvy obcí. Pro krajský soud je zřejmě vhodné použít seznam obcí z daného kraje a k tomuto seznamu přidat množinu dalších, například zmíněná okresní města a katastrální úřady. Detailněji se návrhem tříd a korelací s daty budeme věnovat dále.

6.6 Velikost třídového jazykového modelu

Velikost jazykového modelu, počet parametrů, je důležitá vlastnost pro systém rozpoznávání řeči. Použitím velkých tříd slov dochází k exponenciálnímu růstu počtu parametrů modelu. V navrhovaném řešení použijeme „prořezávání“ jazykového modelu pro snížení počtu parametrů. Nebudeme využívat výše zmíněné metody snižování počtu parametrů, ale zaměříme se, zda nelze využít vlastnosti dat a model zmenšit na základě znalostí z dat.

V trénovacím textu se podíváme bigramy, které mají na pozici slovní historie geografický název a také bigramy, které odhadují pravděpodobnost geografického názvu. V modelu ponecháme jenom nejvíce používané slovní historie a následníky. Předpokládáme, že tím dojde k výraznému snížení počtu parametrů modelu a současně nedojde k ovlivnění kvality rozpoznávání řeči.

Kapitola 7

Experimenty a výsledky

7.1 Popis korpusu

Pro experimenty jsme měli k dispozici kopie soudních rozhodnutí z několika soudů v České republice. Data z jednotlivých soudů byla získána jako kompletní dokumenty soudních rozhodnutí z jednotlivých případů z několika roků. Soubory jsme zpracovali pro použití v experimentech v několika krocích. Z jednotlivých dokumentů bylo třeba správně převést čistý text, očištěný od záhlaví a zápatí jednotlivých stran, včetně úvodních hlaviček a závěrečných odstavců s poučením. Druhým krokem bylo označení specifických slov a slovních spojení, které bylo provedeno na základě pravidel. Například označení jednacích čísel, číslovek, jména a příjmení osob, názvy firem, zkratek typických pro právnické texty a soudní rozhodnutí atd. Tato pravidla byla dělána ručně. Pro experimenty v této práci jsme nejvíce využili nalezení názvů obcí, měst, osad a názvů ulic. Úspěšnost tohoto značení je kolem 95%. V tabulce 7.1 jsou uvedeny části korpusu z jednotlivých soudů po zpracování na čistý text.

Dostupná data jsme rozdělili na část pro trénování jazykových modelů část pro nastavení rozpoznávacího systému pro nejlepší dosažené výsledky a část testovací, na které budeme provádět všechny experimenty. Z každého soudního kraje jsme přibližně 95% dat použili jako trénovací část. Zbylá část jsou vývojová a testovací data. Pro testování rozpoznávání řeči jsme pořídili nahrávky celých soudních rozhodnutí. Celý text soudních rozhodnutí obsahuje v poměru ke své délce velmi málo místních názvů,

Zdroj dat	velikost	počet tokenů (v milionech)
Krajský soud v Praze	339 MB	56,72
Krajský soud v Českých Budějovicích	74 MB	12,23
Krajský soud v Plzni	197 MB	33,21
Krajský soud v Ústí nad Labem	65MB	10,80
Celkem	675 MB	112,96

Tabulka 7.1: Korpusy soudních rozhodnutí

na které se zaměřují naše testy. Proto jsme vybrali i jednotlivé věty, ve kterých jsou místní názvy a pro testování jsme do nich doplnili názvy obcí a ulic pro otestování reprezentativní skupiny místních názvů a jejich správné rozpoznání včetně pravidel pravopisu s velkými písmeny. Pro každou oblast máme trénovací, vývojová i testovací data. Testovací sady jsme rozdělili na několik skupin.

$Test1$ je tvořen z textů soudních rozhodnutí z jednotlivých krajů. Bylo vybrána vždy část z každého soudního kraje. V tabulce 7.2 jsou jednotlivé části poměřeny k jazykovým modelům z jednotlivých krajů. Sloupec označený T_{t1} označuje počet tokenů v testovaném souboru. V_{t1} je počet různých slov v testu. OOV_r je OOV rate slov v textu, která nebyla viděna v trénovacích datech uvedená v procentech a PPL je perplexita. Velmi nízká perplexita a OOV na úrovni $0,3-1\%$ ukazují na relativně velké množství trénovacích dat.

$Test2$ je podmnožina testovacího textu $Test1$. Jsou vybrány věty, které obsahovaly místní názvy. Množiny dat se zmenšily přibližně na 40%. Perplexita se zvýšila nejvýrazněji u dat ze středočeského kraje a nejméně u dat z Ústí nad Labem. OOV rate se výrazně zvýšila až na dvojnásobné hodnoty. Výpis jednotlivých OOV slov ukazuje na výrazné zastoupení místních názvů.

7.2 Rozpoznávání s referenčními modely

Všechny experimenty s rozpoznáváním jsme provedli pomocí rozpoznávacího systému vyvinutého na Katedře kybernetiky na ZČU v Plzni. Pro trénování akustického

	Test 1				Test 2			
Soudní kraj	$\|T_{t1}\|$	$\|V_{t1}\|$	OOVr	PPL	$\|T_{t2}\|$	$\|V_{t2}\|$	OOVr	PPL
Praha	992k	22 505	1,03	16,86	430k	16 753	2,19	25,21
Č. Budějovice	251k	8 258	0,5	12,99	86k	5 006	0,73	14,43
Plzeň	249k	10 317	0,28	12,37	109k	7 233	0,46	13,82
Ústí n. Labem	273k	11 144	0,47	14,31	120k	7 581	0,82	15,06

Tabulka 7.2: Základní testovací sady - text

modelu jsme použili vysoce kvalitní řečový signál. Jednalo se o čtený korpus, kde jednotlivé věty byly vybrány z elektronických verzí nejčtenějších českých novin, a to tak, aby co nejlépe reprezentovaly statistické rozložení českých trifónů, vzhledem k tomu jak se objevují v běžné řeči. Nahráli jsme 800 různých řečníků - 384 mužů a 419 žen. Korpus byl pořízen v tichém prostředí s použitím close-talking mikrofonu (Sennheisser HMD410-6). Celkem bylo nahráno přes 220 hodin řeči.

Řečový signál jsme digitalizovali s frekvencí 22kHz s 16ti bitovým rozlišením. Front-end procesor byl založen na parametrizaci řeči metodou PLP. Na základě několika experimentů bylo zvoleno nastavení 27 spektrálních filtrů a 12 statických kepstrálních koeficientů. Výsledný příznakový vektor jsme pak doplnili o delta a delta-delta koeficienty, takže dimenze obrazového prostoru byla 36. Při zpracování jsme aplikovali také on-line CMN, abychom eliminovali případné změny přenosového kanálu na testovacích datech.

Elementární řečovou jednotkou v našem systému je tří-stavový HMM se spojitou výstupní pravděpodobností spojenou s každým stavem. Vzhledem k tomu, že je počet všech možných českých trifónů příliš velký, použili jsme ke svázání akusticky podobných stavů fonetický rozhodovací strom. Optimální počet stavů a s tím spojený optimální počet složek GMM svázaných s jedním stavem jsme zjistili na základě rozsáhlého množství experimentů. Výsledkem je tedy model, který má 4922 stavů a kde je každý stav reprezentován 16ti složkovým GMM. K tvorbě akustického modelu jsme použili HTK-Toolkit verze 3.4.

Jazykové modely jsme natrénovali různými metodami a podle vývojových dat vybrali vyhlazování Knesser-Ney. Váha jazykového modelu vůči akustickému v de-

		Test 1 – audio				Test 2 – audio			
Soudní kraj	$\|V_{LM}\|$	OOVr	PPL	Corr	Acc	OOVr	PPL	Corr	Acc
Praha	249k	3,5	200	89,32	86,68	5,3	320	84,30	81,1
Č. Budějovice	127k	3,9	239	88,38	85,19	4,8	256	76,85	73,9
Plzeň	217k	2,9	151	90,56	87,92	3,0	190	85,42	81,2
Ústí n Labem	92k	5,0	295	86,81	83,28	5,5	280	80,71	77,3

Tabulka 7.3: Základní experimenty s rozpoznáváním

kodéru byla nastavena na hodnotu 8.

V tabulce 7.3 jsou uvedenu základní experimenty testovacích soudních rozhodnutí s jazykovými modely, které jsou natrénovány z dat příslušného soudního kraje. $|V|$ je velikost slovníku daného modelu, $OOVr$ je hodnota OOV rate slov v testu, PPL je perplexita testovacích dat proti příslušnému modelu, $Corr$ je správnost a Acc je přesnost rozpoznání testovacích dat. Použili jsme opět dvě testovací sady. První sada označená **Test 1 - audio** jsou celá soudní rozhodnutí. Ve druhé sadě **Test 2 - audio** je výběr vět, které obsahovaly místopisné údaje.

Perplexita testovacích dat je větší než u předchozí sady. To je z důvodu, že jsme v sadě použili soudní rozhodnutí nejen ze zmíněných soudních krajů, ale i dalších a dva záznamy odůvodnění rozsudku Nejvyššího soudu. Tyto výsledky budeme dále uvažovat jako referenční.

7.3 Analýza chyb rozpoznávání

Výsledky experimentu rozpoznávání testovacích dat s modely z jednotlivých soudních krajů jsou uvedeny v tabulkách 7.2 a 7.3. Úroveň rozpoznávání je vysoká a započítané chyby nelze ovlivnit jenom nastavením rozpoznávacího systému. Část chyb je způsobena špatnou výslovností.

Celá soudní rozhodnutí testovaná proti modelům natrénovaných z jednotlivých krajů jsou rozpoznávána relativně dobře s mírným poklesem přesnosti u soudních rozhodnutí, která nejsou přímo z daného kraje. Oproti tomu vybrané věty s místopisnými

Korpusová sada	Obce_T	$\|obce\|$	Ulice_T	$\|ulice\|$	$\|Obce\_R\|$	$\|Ulice\_R\|$
Praha	490 200	8 958	111 469	7 890	20 824	23 747
Č. Budějovice	82 615	3 352	15 908	2 369	14 467	14 124
Plzeň	212 456	5 512	43 378	4 164	13 701	11 295
Ústí nad Labem	105 446	3 621	38 383	3 493	7 780	18 245

Tabulka 7.4: Označené místní názvy v korpusu

údaji vykazují výrazné snížení přesnosti v rozpoznávání. Zvýšené OOV rate je v důsledku geografických údajů a snížení obecného textu v tetovací sadě.

Analýzou chyb u testu jsme došli k závěrům: Velký význam na snížení úspěšnosti mají OOV slova v testech jednotlivých vět. Slova jsou z velké míry geografické údaje, které se v trénovacích datech z jiného kraje nemohly objevit. V některých případech je správně rozpoznaná část názvu obce či ulice, ovšem není správně velikost písmen. Například rozdíl v názvu ulice `U nádraží` a `u nádraží` a další.

7.4 Třídy slov v trénovacím textu

Třídy jsme v textu měli označeny v trénovacím textu předchozím zpracováním a rozděleny na názvy obcí a ulic. V tabulce 7.4 jsou uvedeny počty označených obcí (Obce_T) a ulic (Ulice_T) v korpusu včetně počtu různých slov v těchto skupinách. Jde o relativně mohutné třídy, které ovšem nepokývají všechna sídla v daném kraji, jak ukazují poslední sloupce v tabulce ($|Obce\_R|$, $|Ulice\_R|$). Počty názvů obcí a ulic jsme získali z rejstříků uvedených výše. Vyskloňované tvary z těchto rejstříků jsme získali částečně automaticky a následnou ruční kontrolou. Uvedené počty jsou souhrny všech tvarů. Slova označená v textech nejsou jenom z množiny geografických názvů pro daný kraj. V soudních rozhodnutích jsou i názvy mimo příslušný region. Například adresy osob a podniků bývají z jiného kraje, místo narození osob bývá mimo kraj, velké společnosti mají jediné sídlo a působí celoplošně v republice a pod.

Skupiny slov zastupující obce i ulice jsou ještě rozděleny podle morfologické kategorie pád, kterých je v českém jazyce sedm, z nichž pátý pád používaný pro

Kraj	Praha		Č. Bud.		Plzeň		Ústí n/L.	
Obce	T	$\|W\|$	T	$\|W\|$	T	$\|W\|$	T	$\|W\|$
1. pád	231 179	4917	31 086	1 907	95 799	2 876	54 608	2 267
2. pád	13 465	1040	2948	406	8 800	738	3977	300
3. pád	3 014	216	182	73	838	157	215	53
4. pád	7 809	158	628	107	1 653	161	184	32
6. pád	233 269	2372	47 488	764	104 757	1 422	46 193	905
7. pád	1 465	255	283	95	649	158	269	64

Tabulka 7.5: Označené obce podle pádu

Kraj	Praha		Č. Bud.		Plzeň		Ústí nL.	
Ulice	T	$\|W\|$	T	$\|W\|$	T	$\|W\|$	T	$\|W\|$
1.pád	108572	7139	14 470	1 981	39 002	3 439	38 082	3 352
2.pád	257	92	118	56	426	127	27	15
3.pád	15	9	36	21	40	17	0	0
4.pád	3	2	3	2	33	6	0	0
6.pád	2604	635	1 270	301	2 820	547	274	126
7.pád	18	13	11	8	57	28	0	0

Tabulka 7.6: Označené ulice podle pádů

oslovování se v textu nevyskytuje. Tabulky 7.5 a 7.6 zobrazují počty označených obcí a ulic rozdělené podle pádů. Sloupec T značí počet označených tokenů v daném pádu a sloupec $|W|$ počet různých tvarů v daném pádu. Zajímavé je zastoupení pádů u názvů ulic téměř výhradně prvním a šestým pádem.

7.5 Rozpoznávání s třídovými modely

Označením tříd jsme získali třídový model. Použití tříd znamená pro rozpoznávač vytvořit tzv. expandovaný model, který pro každé slovo ze tříd získá jeho pravděpodobnost vynásobením pravděpodobnosti třídy a pravděpodobnosti slova uvnitř třídy

podle vzorce 4.69. Tato expanze vede u velkých modelů a tříd, které mají větší rozměr k růstu velikosti modelu nad možnosti zpracování, proto jsme přistoupili k optimalizaci modelu, kde jsme museli omezit bigramy, kde jsou zastoupeny třídy. Základní myšlenkou je vyřazení bigramů s malou četností, které obsahují třídu, z modelu. Expanze takových bigramů přidává velké množství parametrů do výsledného modelu.

V trénovacích textech se vyskytuje 22 526 různých bigramů, které obsahuji některou ze tříd obcí na místě slovní historie a týká se jich expanze modelu. Každý z těchto bigramů má určitou četnost. Vybereme-li 14 bigramů, které se objevily v datech nejčastěji, součet jejich výskytů je 70 % všech výskytů z této sady bigramů. Pro pokrytí 80 % je potřeba 33 nejčetnějších. Druhá část bigramů - 10 137 různých, které se týká expanze, má jednu ze tříd obcí na druhém místě a jde o slovní historie, které předchází označení obce. 18 nejčetnějších pokrývá 80 % a k 90% pokrytí stačí 44 bigramů.

U bigramů s třídami ulic je situace podobná. Počet různých bigramů, kde je ulice na místě slovní historie je 3920. Nejčetnějších 33 těchto bigramů zastupuje 95 % všech výskytů a pro získání 98 % stačí 77 bigramů. Počet slov, která uvozují název ulice, tedy různé historie vyslovení názvu je 3004 různých a 12 nejčetnějších tvoří 95 % četnosti všech těchto bigramů. Pro pokrytí 98 % je dostatečných 34 bigramů-

Uvedené možnosti ukazují jednoduché prořezání bigramů s třídou pro zmenšení modelu, který již nebude příliš výpočetně náročný. Experimenty s nastavením prořezávání jsme provedli na datech z Plzeňského soudního kraje. Model s označením tříd obcí a ulic má slovník velký asi 400 tisíc slov a počet parametrů modelu je asi 12 milionů. Při expanzi všech možných bigramů s třídou obcí a ulic dostaneme jednoduchým vynásobením počet parametrů expandovaného modelu kolem 650 milionů a to pouze v případě, že se nebude provádět expanze bigramů obsahující dvě třídy jdoucí po sobe. V takovém případě je nárůst počtu parametrů exponenciální.

Tabulka 7.7 ukazuje nárůst parametrů modelu při prořezání uvedeném výše. Sloupec Obce-H obsahuje informace o prořezání bigramů, kde je název obce na pozici slovní historie, a Obce-W prořezání bigramů, kde je je název obce jako slovo po historii. Číslo uvádí počet nejčetnějších bigramů, které jsme v modelu ponechali a v závorce je uveden procentní zastoupení výskytů těchto bigramů.

Obce-H	Obce-W	Ulice-H	Ulice-W	Přírůstek parametrů
14 (70 %)	18 (80 %)	33 (95 %)	12 (95 %)	1 191k
14 (70 %)	18 (80 %)	77 (98 %)	34 (98 %)	1 959k
33 (80 %)	44 (90 %)	77 (98 %)	34 (98 %)	1 808k
33 (80 %)	44 (90 %)	33 (95 %)	12 (95 %)	2 575k
51 (85 %)	44 (90 %)	77 (98 %)	34 (98 %)	2 055k
51 (85 %)	44 (90 %)	77 (98 %)	34 (98 %)	2 822k

Tabulka 7.7: Prořezání jazykového modelu z Plzeňského kraje

Pro další experimenty jsme vybrali prořezávání, které je uvedeno na čtvrté řádce tabulky z důvodu výrazného snížení velikosti modelu. Tedy $33 + 44$ expandovaných bigramů s obcemi obce a $33 + 12$ expandovaných bigramů s ulicemi. Ostatní bigramy, které obsahují třídu obcí a ulic z modelu vyřadíme. Tímto prořezáním došlo také k tomu, že třídy obcí a ulic v jednotlivých pádech se redukovaly pouze na první a šestý pád. Je to celkem očekávané prořezání podle četností uvedených v tabulkách 7.5 a 7.6

7.6 Rozšíření slovníku jazykových modelů

V kapitole o jazykovém modelování češtiny jsme již uvedli možný postup s rozšířením slovníku. Použili jsme jazykový model s třídami v úloze rozpoznávání komentáře hokejového zápasu. Třídy byly sestaveny ze soupisky hráčů zápasu a bylo možné třídy nastavit podle aktuální soupisky zápasu. Třídy byly připraveny pro jména aktérů zápasu podle mluvnické kategorie - pádu. Pravděpodobnost slov uvnitř třídy byla nastavena rovnoměrně mezi všechna slova. Použití těchto tříd přineslo výrazné snížení OOV slov v nových zápasech a zlepšení výsledků rozpoznávání. Použité řešení předpokládalo každé slovo v jedné třídě kromě jmen, která tvořila třídy. U komentáře hokeje bylo toto řešení vhodné i proto, že mohutnost tříd v trénovacím textu odpovídala mohutnosti nové třídy a relativně stejně pravděpodobný výskyt slov ze třídy.

Počet obcí a ulic v jednotlivých soudních krajích je uveden v tabulce 7.8. Získali

Krajský soud	obce	ulice	obce - data	ulice - data
Praha	2625	8346	4917	7139
České Budějovice	2360	7068	1907	1981
Plzeň	2387	3208	2876	3439
Ústí nad Labem	1355	7529	2267	3352

Tabulka 7.8: Počet obcí a ulic v soudních krajích ČR

jsme ho z rejstříků, které jsou uvedeny níže. V dalších sloupcích je uveden počet obcí a ulic v prvním pádu, které se vyskytly v trénovacích textech. Geografické názvy z trénovacích textů z textů jednotlivých krajů neobsahují názvy jenom z konkrétního soudního kraje. Naopak množina získaných obcí a ulic z textů některých krajů přesahuje mohutnost množiny dostupné z rejstříků. Obce jsou v rejstřících často uvedeny jedním zástupcem a například městské části nebo osady pařící k obci je třeba dohledávat podrobněji.

Dalším faktorem je rozvrstvení obyvatel v sídlech v jednotlivých krajích. Například v Plzeňském kraji, který je málo osídlen v pohraničních oblastech, nejsou v obcích určeny názvy ulic. Dále je v testech uveden výrazný počet obcí a ulic z ostatních krajů. Je to dáno například tím, že soudy si mezi sebou převádějí podle místní příslušnosti, proto se objevuje hodně názvů okresních soudů. Dalšími příklady jsou vyjmenování sídel společností, místa narození a další.

7.6.1 Sestavení tříd obcí a ulic

Tato kapitola se bude zabývat sestavením tříd obcí a ulic s rozšířeným slovníkem oproti původnímu slovníku z trénovacích dat. V jazykovém modelu i případném diktovacím systému budou obsažena všechna slova týkajících se místních názvů. Pro uživatele systému je tato znalost velkou výhodou proti modelu, který je sestaven jenom z trénovacích dat, protože si může být jistý, že systém tato slova zná.

	Test 1 – audio				Test 2 – audio			
Soudní kraj	OOVr	PPL	Corr	Acc	OOVr	PPL	Corr	Acc
Praha	3,4	306	84,20	82,38	2,8	382	82,00	78,33
Č. Budějovice	3,6	288	83,81	81,26	2,2	356	74,85	70,38
Plzeň	2,7	201	87,01	83,15	0,9	290	80,42	75,70
Ústí n Labem	4,5	351	85,98	80,49	1,5	420	76,10	73,23

Tabulka 7.9: Výsledky s třídami s uniformním rozložením pravděpodobnosti

7.6.1.1 Uniformní rozdělení ve třídách

První test jsme udělali podobně jako výše zmíněný systém na rozpoznávání hokejových komentářů. Všechna slova ve třídách budou mít stejnou pravděpodobnost a dekodér bude vybírat slova pouze na základě akustické informace. Třídy obcí i ulic ve všech pádech byly naplněny příslušnými tvary slov. Víceslovné názvy jsou reprezentovány jedním slovem jako sousloví. Vyskloňované tvary obcí a ulic jsme získali částečně automaticky s důkladnou ruční kontrolou u nepravidelných tvarů.

V tabulce 7.9 jsou uvedeny výsledky rozpoznávání s třídami s uniformní pravděpodobností všech slov. Výsledky jsou horší oproti původním modelům. Zvláště druhá testovací sada, která je zaměřena na místní názvy, dosáhla výrazně horších výsledků. Při detailním pohledu do výsledků docházelo i k chybám rozpoznání u velkých měst, která v předchozím testu měla minimální chybovost. Je zřejmé, že uniformní rozložení pravděpodobnosti uvnitř tříd není vhodné. Pokusíme se nalézt lepší rozložení pravděpodobnosti.

7.6.1.2 Rozdělení pravděpodobnosti ve třídách obcí

V předchozí kapitole jsme navrhli porovnání rozložení pravděpodobnosti ve třídách se skutečným rozložením pravděpodobnosti v trénovacích textech.

V tabulce 7.10 jsou uvedeny korelační koeficienty mezi daty z trénovacích dat a vektory, které jsme navrhli následovně:

Krajský soud	**A**	**B**	**C**	**D**
Plzeň	0	0,68	0,69	0,69
Praha	0	0,62	0,64	0,64
České Budějovice	0	0,71	0,74	0,75
Ústí nad Labem	0	0,65	0,66	0,66

Tabulka 7.10: Korelační koeficienty mezi trénovacími daty a navrženou třídou obcí

A množina obcí z daného kraje, uniformní rozdělení[1]

B množina obcí z daného kraje, rozdělení podle počtu obyvatel

C množina obcí z daného kraje + okresní města, rozdělení podle počtu obyvatel

D množina obcí z daného kraje + okresní města + katastrální území, rozdělení podle počtu obyvatel

Nejvyšší korelační koeficienty jsme dostali pro odhady **C** a **D**. Lze usuzovat, že právě takto navržená třída obcí je nejlepším odhadem výskytu názvů měst a obcí v textech soudních rozhodnutí. Rozšíření třídy o názvy katastrálních území je vhodné zejména pro přesně definované názvy katastrálních území. které jsou jedinečné v celé republice. Korelačního koeficientu $\rho = +1$ nelze dosáhnout z důvodu, že námi navržený vektor obcí je delší o obce, které se v textech nevyskytly.

7.6.1.3 Rozdělení pravděpodobnosti ve třídách ulic

Podobně jako jsme navrhli rozložení pravděpodobnosti ve třídě obcí budeme postupovat u ulic. Bohužel u ulic nemáme rejstřík, který by udával počet obyvatel. Z dostupných rejstříků lze zjistit počet čísel popisných, který by mohl pomoci v odhadu pravděpodobnosti, ale v případě velkých bytových bytových domů je nepřesný. V telefonním seznamu pevných linek jsou uvedeny telefonní čísla a adresy. Počet telefonních linek by mohl být trochu lepší. Dalším registrem je Registr ekonomických subjektů, který se týká právnických osob. Právnické osoby jsou v soudních řízení často

[1]Konstatntní hodnota náhodné veličiny je specifická a korelace pro ni není definována. Nulou značíme, že proměnné jsou nekorelované, lineárně nezávislé.

Krajský soud	**A**	**B**	**C**	**D**	**E**
Plzeň	0	0,59	0,69	0,51	0,75
Praha	0	0,44	0,64	0,52	0,69
České Budějovice	0	0,51	0,71	0,48	0,78
Ústí nad Labem	0	0,52	0,64	0,48	0,72

Tabulka 7.11: Korelační koeficienty mezi trénovacími daty a navrženou třídou ulic

přítomné a proto jejich statistika by mohla odhadu pravděpodobnosti také pomoci. Specifickou skupinou právnických osob jsou advokáti, kteří se účastní soudních řízení. V soudním rozhodnutí jsou vždy uváděna celé jména advokátů a jejich působiště.

Opět jsme provedli srovnání pomocí korelačního koeficientu mezi vektorem ulic v prvním pádu, které se objevily v trénovacích textech, a navrženou třídou. Jednotlivé návrhy tříd jsou uvedeny v tabulce 7.11 s následujícím označením:

A množina ulic z daného kraje, uniformní rozdělení[2]

B množina ulic z daného kraje, rozdělení podle počtu čísel popisných

C množina ulic z daného kraje, rozdělení podle počtu právnických osob v ulici

D množina ulic z daného kraje, rozdělení podle advokátů působících v ulici

E kombinace C + D

Z tabulky vyplývá, že nejvyšší korelační koeficient dosáhla kombinace dat z rejstříku ekonomických subjektů a advokátů. Ekonomické subjekty lze odůvodnit jejich vyšší účastí v soudních sporech a koncentrace sídel v ulicích mimo bytovou zástavbu, například v průmyslových zónách. Seznam advokátů má velký přínos zejména pro zmíněnou účast advokátů v soudním řízení, ale například i pro sídla v ulicích s kancelářskými budovami. Kombinace těchto dvou hodnot dávala nejlepší výsledky v poměru $D + 0.3 * E$. Pro každý kraj byl poměr mírně odlišný a tato hodnota vycházela v průměru nejlépe.

[2]Konstatntní hodnota náhodné veličiny je specifická a korelace pro ni není definována. Nulou značíme, že proměnné jsou nekorelované, lineárně nezávislé.

Zkoušeli jsme použít i další ze jmenovaných rejstříků – Jednotný telefonní seznam. Nakonec jsme ho použili jenom při vývoji a na datech z Plzeňského kraje. Čísla v telefonním seznamu jsou velmi korelovaná s hodnotou počtu čísel popisných. K celkové korelaci neměla tato hodnota velký přínos a proto jsme telefonní seznam pro další kraje už nepoužili. Bylo velmi obtížné získat strukturovanou podobu telefonního seznamu z dostupných dat a proto jsme je dále nepoužili.

7.7 Výsledky rozpoznávání s rozšířenými třídami

Navrhli jsme rozložení pravděpodobnosti uvnitř tříd obcí a ulic, které nejvíce odpovídá původnímu rozložení pravděpodobnosti v trénovacích datech. Provedli jsme experimenty s rozpoznáváním, kde jsme použili navržené třídy. Výsledky jsou uvedené v tabulce 7.12. Pro rozložení pravděpodobnosti obcí jsme použili seznam obcí v kraji s přidanými okresními městy a katastrálními územími ohodnocené počtem obyvatel. Třídy ulic byly sestaveny z ulic daného kraje a ohodnoceny počtem ekonomických subjektů a počtem působících advokátů v ulici.

Výsledky na testovací sadě **Test 1** obsahující celá soudní rozhodnutí jsou téměř shodné jako při použití modelu, který vznikl přímým natrénováním z trénovacích dat. Sada **Test 2** je složená zvláště pro otestování rozpoznávání jmenných entit - obcí a ulic. Výsledky s novými modely přinesly výrazné snížení OOV rate díky rozšíření slovníku o obce a ulice celého kraje. Současně se správně tyto názvy obcí a ulic rozpoznaly včetně korektních velkých a malých písmen podle pravidel českého pravopisu.

Srovnání výsledků je uvedeno s tabulce 7.13. V řádcích jsou uvedeny průměrné hodnoty z předchozích tabulek. Označení *Baseline* je pro původní systém rozpoznávání. Ve druhém řádku *Uniform* jsou hodnoty z testu s modelem s třídami obcí a ulic, které měli uniformní rozložení pravděpodobnosti uvnitř tříd. Závěrečné výsledky s prezentovaným rozšířeným rozložením pravděpodobnosti jsou označeny *Extended*.

Tyto výsledky potvrdily, že návrh na rozšíření slovníku pomocí tříd je správný. Uvnitř tříd je nutné pro dobré výsledky dodržet správné rozložení pravděpodobnosti. Těmito experimenty nedošlo ke zlepšení celkové správnosti rozpoznávání, ale důležité

	Test 1 – audio				Test 2 – audio			
Soudní kraj	OOVr	PPL	Corr	Acc	OOVr	PPL	Corr	Acc
Praha	3,4	250	83,60	85,84	2,8	282	82,00	83,37
Č. Budějovice	3,6	263	87,90	84,67	2,2	312	74,85	76,81
Plzeň	2,7	210	88,54	84,65	0,9	260	80,42	82,05
Ústí n Labem	4,5	315	86,22	80,90	1,5	354	76,10	77,92

Tabulka 7.12: Výsledky s třídami s navrženým rozložením pravděpodobnosti

	Test 1 – audio				Test 2 – audio			
Testy	OOVr	PPL	Corr	Acc	OOVr	PPL	Corr	Acc
Baseline	3,83	221	88,77	85,77	4,65	262	81,82	78,38
Uniform	3,55	287	85,25	81,82	1,85	362	78,34	74,41
Extended	3,55	260	86,57	84,02	1,85	302	83,56	80,04

Tabulka 7.13: Srovnání výsledků rozpoznávání - průměrné hodnoty

je, že nedošlo ke zhoršení použitelnosti v diktovacích systémech a naopak ke zlepšení rozpoznávání konkrétních skupin slov. Pro uživatele diktovacího systému je důležité, že systém rozeznává všechny místní názvy, protože nemusí při diktování přemýšlet, zda dané slovo vyslovit s případnou ruční opravou nebo ho dopsat do dokumentu přímo.

Pro uživatele diktovacího systému je důležité, že systém rozeznává všechny místní názvy. Při diktování si je uživatel relativně jistý, že geografické názvy může vyslovovat s částečnou jistotou správného rozpoznání. Minimalizuje se tím čas případných ručních oprav a dopisování nerozpoznaných slov do dokumentu.

Kapitola 8

Závěr

Disertační práce se věnuje oboru rozpoznávání řeči a detailně zpracovává jazykové modelování pro diktovací systém, který byl připravován pro použití v soustavě soudů České republiky. Zabývá se oblastí nedostatečnosti trénovacích dat pro skupiny slov pokrývající geografické údaje.

Cílem práce nebylo vylepšení diktovacího systému jako celku a zlepšení přesnosti rozpoznávání k maximu. Snahou bylo nalézt možnost, jak obohatit slovní zásobu systému tak, aby systém byl schopen nová slova rozpoznávat a zároveň se nezhoršily jeho parametry jako celku.

Byla navržena metoda rozšíření jazykového modelu o slova nevyskytující se v trénovací množině. Byl ukázán přístup, který byl částečně používán v autorových předchozích pracích, ale je nedostačující pro systémy rozpoznávání řeči s třídami slov, které mají velký rozdíl mohutnosti množiny slov, která má být v původních datech nahrazena. Byl navržen postup úpravy pravděpodobností uvnitř tříd jazykového modulu, který spolehlivě vloží do jazykového modelu nová slova a systém je následně schopen slova rozpoznávat.

Prezentované výsledky na široké množině dat z čtyř soudních krajů potvrzují správnost postupu rozšíření slovníku jazykového o nová slova, která nelze získat z trénovacích dat. Jedním z nejnáročnějších úkolů, bylo označení geografických názvů v textech a jeho následné zpracování. Podobně náročné bylo získání strukturovaných dat z veřejných rejstříků a příprava tříd do jazykového modelu. Data z rejstříků

byla nekonzistentní a obsahovala velké množství chyb. V budoucnu se nabízí využití nových základních registrů veřejné správy, které mají tyto problémy vyřešit pro celou statní správu.

V disertační práci jsou postupně naplněny cíle, které byly stanovené ve třetí kapitole. Autor věří, že získané poznatky budou v krátké době využity v systémech rozpoznávání řeči, které jsou využívány při řešení projektů na našem pracovišti a že disertační práce přinese i nové podněty a nápady, které bude potřeba dále rozpracovat a ověřit.

Resumé

Tato práce se zabývá statickými modely přirozeného jazyka a jejich speciálním druhem jazykovými modely postavené na třídách. Tyto modely se používají v systémech rozpoznávání řeči.

Práce se nezabývá obecným zlepšováním jazykového modelování, ale zaměřuje na oblast specifických skupin slov, které nelze dobře modelovat z dostupných trénovacích dat. Nedostatečnost dat je limitující faktor pro jazykové modelování současných metod. Přidání slov do systému není triviální problém z důvodu požadavku, aby systém tato slova dokázal rozpoznat, budou-li vyslovena a současně se nezhoršila obecná přesnost rozpoznávání.

Jelikož tyto třídy jsou otevřené, vyskytuje se často legitimní potřeba tato slova přidávat do již hotového ASR systému. Toto přidání ovšem není jednoduchá záležitost a často vede ke snížení přesnosti rozpoznávání. Cílem práce je připravit postup rozšíření slovníku systému rozpoznávání řeči a zajistit správné nastavení systému pro správné rozpoznávání těchto slov při současném nezhoršení parametrů celého systému.

Resumé

This thesis aims at research into the static natural language models, especially the language models build on word classes. Such language models are (widely, commonly) used in automatic speech recognition systems.

This work doesn't aim at general research into improving the aforementioned language models. Instead, the focus of the thesis is laid on specific word classes(, i.e. specific set of words syntactically or semantically similar). The words from these classes cannot be modeled reliably from the training data. The main obstacle here is the insufficient amount of training data.

Since these classes are open, there is often legitimate need to introduce these words into an already existing ASR system. This is not an simple task, however, and often hurts the ASR system recognition accuracy.The purpose of the research was to develop an approach for augmenting the vocabulary of an ASR system and to determine the correct settings of the ASR system resulting in correct recognition of the problematic word classes, while the recognition accuracy remains the same.

Literatura

[1] Yoshua Bengio, Duchar Réjean, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, March 2003.

[2] Brown P. F., Della Pietra, V. J., deSouza P. V., Lai J. C., Mercer R. L. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, 1992.

[3] Drábková J. *Tvorba jazykového modelu založeného na třídách.* PhD thesis, Technická univerzita v Liberci, 2005.

[4] Gao J., Goodman J. T., Cao G., Li Hang. Exploring asymmetric clustering for statistical language modeling. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 183–190, Morristown, NJ, USA, 2001. Association for Computational Linguistics.

[5] Gao J., Microsoft. Distribution-based pruning of backoff language models, 2000.

[6] Gao J., Zhang M. Improving language model size reduction using better pruning criteria. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 176–182, Morristown, NJ, USA, 2001. Association for Computational Linguistics.

[7] Joshua T. Goodman. A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403 – 434, 2001.

[8] Gao J. Goodman J. Language model size reduction by pruning and clustering. In *ICSLP'00*, Beijing, China, 2000.

[9] Hajič J. *Disambiguation of Rich Inflection (Computational Morphology of Czech).* Karolinum, Charles University Press, Prague, Czech Republic, 2004.

[10] Hajič J., Hajičová E., Rosen A. Formal Representation of Language Structures. *TELRI Newsletter*, (3):12–19, 1996.

[11] Hoidekr J., Psutka J. V., Pražák A., Psutka J. Benefit of a class-based language model for real-time closed-captioning of tv ice-hockey commentaries. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006.

[12] Ircing P. *Large vocabulary continuous speech recognition of Highly Inflectional Language (Czech).* PhD thesis, Západočeská univerzita v Plzni, 2001.

[13] Ircing P., Hoidekr J., Psutka J. Exploiting linguistic knowledge in language modeling of czech spontaneous speech. In *Proceedings of the 5th International Conference on Language Resources and Evaluation*, 2006.

[14] Jelinek F., Lafferty J. D., Mercer R. L. Basic methods of probabilistic context free grammars. In *Speech Recognition and Understanding. Recent Advances, Trends, and Applications*, volume F75, pages 345–360, Springer Verlag, 1992.

[15] Jelinek F., Merialdo B., Roukos S., Strauss M. A dynamic language model for speech recognition. In *HLT '91: Proceedings of the workshop on Speech and Natural Language*, pages 293–295, Morristown, NJ, USA, 1991. Association for Computational Linguistics.

[16] Martin. J. H. Jurafsky D. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.* Prentice Hall, January 2000.

[17] Katz S. Estimation of probabilities from sparse data for the language model component of a speech recognizer, 1997.

[18] Kneser R. Statistical language modeling using a variable context length. In *Proc. ICSLP '96*, volume 1, pages 494–497, Philadelphia, PA, 1996.

[19] Stefan Kombrink, Tomáš Mikolov, Martin Karafiát, and Lukáš Burget. Recurrent neural network based language modeling in meeting recognition. In *Proceedings of Interspeech 2011*, volume 2011, pages 2877–2880. International Speech Communication Association, 2011.

[20] Kuhn R., De Mori R. A cache-based natural language model for speech recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(6):570–583, 1990.

[21] Manning C. D., Schütze H. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999.

[22] McCandless M., Glass J. Empirical acquisition of word and phrase classes in the ATIS domain. In *Third European Conference on Speech Communication and Technology, Berlin, Germany*, September 1993.

[23] McCandless M., Glass J. Empirical acquisition of language models for speech recognition. In *Proc. ICSLP '94*, Yokohama, Japan, 1994.

[24] Tomáš Mikolov. Language modeling of czech using neural networks. In *Proc. 13th Conference STUDENT EEICT 2007*, pages 1–3. Faculty of Electrical Engineering and Communication BUT, 2007.

[25] Tomáš Mikolov. Language models for automatic speech recognition of czech lectures. In *Proc. STUDENT EEICT 2008*, pages 1–5. Faculty of Electrical Engineering and Communication BUT, 2008.

[26] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010)*, volume 2010, pages 1045–1048. International Speech Communication Association, 2010.

[27] Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. Extensions of recurrent neural network language model. In *Proceedings of the 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2011*, pages 5528–5531. IEEE Signal Processing Society, 2011.

[28] Tomáš Mikolov, Jiří Kopecký, Lukáš Burget, Ondřej Glembek, and Jan Černocký. Neural network based language models for highly inflective languages. In *Proc. ICASSP 2009*, page 4. IEEE Signal Processing Society, 2009.

[29] Nouza J. Strategies for developing a real-time continuous speech recognition system for czech language. In *TSD '02: Proceedings of the 5th International Conference on Text, Speech and Dialogue*, pages 189–196, London, UK, 2002. Springer-Verlag.

[30] Nouza J., Drábková J. Combining lexical and morphological knowledge in language model for inflectional (czech) language. pages 494–497, Philadelphia, PA, 2002.

[31] Pražák A., Psutka J. V., Hoidekr J., Kanis J., Müller L., Psutka J. Automatic online subtitling of the czech parliament meetings. In *Text, Speech and Dialogue, 9th International Conference*, 2006.

[32] Price P. J. Evaluation of spoken language systems: the atis domain. In *HLT '90: Proceedings of the workshop on Speech and Natural Language*, pages 91–95, Morristown, NJ, USA, 1990. Association for Computational Linguistics.

[33] Psutka J. *Komunikace s počítačem mluvenou řečí*. Academia, Praha, 1995.

[34] Psutka J., Hoidekr J., Ircing P., Psutka J. V. Automatic transcription of tv ice-hockey commentary. In *Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics SCI?2003*, pages 419–423, 2003.

[35] Psutka J., Müller L., Matoušek J., Radová V. *Mluvíme s počítačem česky* . Academia, Praha, 2006.

[36] Psutka J., Psutka J. V., Ircing P., Hoidekr J. Recognition of spontaneously pronounced tv ice-hockey commentary. In *Proceedings of the ISCA&IEEE Workshop on Spontaneous Speech Processing and Recognition SSPR 2003*, pages 83–86, 2003.

[37] Ries K., Buo F. D., Waibel A. Class phrase models for language modelling. In *Proc. ICSLP '96*, volume 1, pages 398–401, Philadelphia, PA, 1996.

[38] Holger Schwenk and Jean-Luc Gauvain. Training neural network language models on very large corpora. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 201–208, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics.

[39] Seymore K., Rosenfeld R. Scalable backoff language models. In *Proc. ICSLP '96*, volume 1, pages 232–235, Philadelphia, PA, 1996.

[40] Stolcke, A. Entropy-based pruning of backoff language models. In *Proceedings of the ARPA Workshop on Human Language Technology.*, 1998.

Autorovy publikace

[1] J. Hoidekr, A. Pražák, J. Psutka, and Z. Tychtl. Systém automatického vyhledávání klíčových segmentů v rozsáhlém audiovizuálním archivu hokejových zápasů, 2007.

[2] J. Hoidekr, Psutka Josef V., A. Pražák, and J. Psutka. Benefit of a class-based language model for real-time closed-captioning of tv ice-hockey commentaries. pages 2064–2067, Paris, 2006. ELRA.

[3] P. Ircing, J. Hoidekr, and J. Psutka. Exploiting linguistic knowledge in language modeling of czech spontaneous speech. pages 2600–2603, Paris, 2006. ELRA.

[4] P. Ircing, D. Oard, and J. Hoidekr. First experiments searching spontaneous czech speech. pages 835–836, New York, 2007. ACM Press.

[5] P. Ircing, P. Pecina, D. Oard, J. Wang, R. White, and J. Hoidekr. Information retrieval test collection for searching spontaneous czech speech. *Lecture Notes in Artificial Intelligence*, pages 439–446, 2007.

[6] A. Pražák, J. Psutka, J. Hoidekr, J. Kanis, L. Müller, and J. Psutka. Adaptive language model in automatic online subtitling. pages 479–483, Anaheim, 2006. ACTA Press.

[7] A. Pražák, Psutka Josef V., J. Hoidekr, J. Kanis, L. Müller, and J. Psutka. Automatic online subtitling of the czech parliament meetings. *Lecture Notes in Artificial Intelligence*, pages 501–508, 2006.

[8] J. Psutka, J. Hoidekr, P. Ircing, and Psutka Josef V. Recognition of spontaneous speech - some problems and their solutions. pages 169–172, Orlando, 2006. IIIS.

[9] J. Psutka, J. Psutka, P. Ircing, and J. Hoidekr. Recognition of spontaneously pronounced tv ice-hockey commentary. pages 83–86, Tokyo, 2003. Tokyo Institute of Technology.

[10] Psutka Josef V., J. Hoidekr, P. Ircing, and J. Psutka. Automatic transcription of tv ice-hockey commentary. pages 419–423, Orlando, 2003. International Institute of Informatics and Systemics.

[11] Jan Švec, Jan Hoidekr, Daniel Soutner, and Jan Vavruška. Web text data mining for building large scale language modelling corpus. 6836:356–363, 2011.