FAKULTA
APLIKOVANÝCH VĚD
ZÁPADOČESKÉ
UNIVERZITY
V PLZNI

AUTOREFERÁT

disertační práce

PLZEŇ, 2013

Ing. Jan Švec

Ing. Jan Švec

Diskriminativní model pro porozumění mluvené řeči

Autoreferát disertační práce k získání akademického titulu „Doktor"
v oboru
Kybernetika

Plzeň, 12. srpna 2013

Jan Švec

Discriminative model for spoken language understanding

Report on the doctoral thesis submitted in conformity with requirements for the degree of "Doctor of Philosophy"

in the field of

Cybernetics

Plzeň, August 12, 2013

Disertační práce byla vypracována v prezenčním/kombinovaném doktorském studiu na katedře kybernetiky fakulty aplikovaných věd ZČU.

Uchazeč: Ing. Jan Švec
Fakulta aplikovaných věd
Katedra kybernetiky
Univerzitní 8, 306 14 Plzeň

Školitel: Prof. Ing. Josef Psutka, CSc.
Fakulta aplikovaných věd
Katedra kybernetiky
Univerzitní 8, 306 14 Plzeň

Oponenti:

Autoreferát byl rozeslán dne: .....

Obhajoba disertační práce se koná dne ..... před komisí v oboru Kybernetika na FAV ZČU, Univerzitní 22, 306 14 Plzeň, v místnosti ..... v ..... hodin.

S disertační prací je možno se seznámit na studijním oddělení FAV ZČU, Univerzitní 22, UV 206.

Prof. Ing. Josef Psutka, CSc.
předseda oborové rady,
obor Kybernetika

Anotace

Předkládaná disertační práce je věnována problematice porozumění mluvené řeči. Práce prezentuje nový diskriminativní model určený pro tuto úlohu. Nejprve je popsána úloha porozumění řeči v kontextu hlasových dialogových systémů a jeho souvislost s rozpoznáváním řeči. Následuje přehled současného stavu řešené problematiky. Odstavce věnované tomuto tématu popisují jednak metody používané pro porozumění mluvené řeči, ale i metody z dalších oblastí zpracování řeči, které s prezentovaným modelem úzce souvisí. Dále jsou vytyčeny a odůvodněny cíle této disertační práce – především se jedná o vývoj nového diskriminativního modelu schopného zpracovat neurčitý vstup v podobě slovní nebo fonémové mřížky a následně vygenerovat více výstupních významových hypotéz. Jeden z podcílů je pak věnován výzkumu metody pro efektivní kombinaci znalostního a statistického přístupu k návrhu modulu porozumění. Porozumění mluvené řeči je dekomponováno do třech dílčích modelů – konceptového modelu, modelu detekce sémantických entit a modelu zarovnání. Zatímco konceptový model přiřazuje celé promluvě globální význam v podobě abstraktního sémantického stromu, model detekce sémantických entit označuje lokální dílčí významy pomocí jednotlivých sémantických entit. Následně model zarovnání provádí provázání těchto dvou dílčích významových reprezentací. Konceptový model je v této práci reprezentován hierarchickým diskriminativním modelem, který vznikl jako rozšíření existujícího statistického modelu založeného na klasifikátorech sémantických n-tic. Model detekce sémantických entit pak provádí hledání výskytů sémantických entit popsaných pomocí expertem definovaných bezkontextových gramatik. Po popisu těchto modelů následuje definice úlohy sestávající se z popisu dat použitých v experimentech a z popisu metodiky vyhodnocení. Součástí definice úlohy je i popis modelů a dekodéru pro automatické rozpoznávání řeči. Následuje experimentální ověření navržených modelů, přičemž jsou zdůvodněny konkrétní volby parametrů. Závěrečná kapitola shrnuje přínos navržené metody pro porozumění mluvené řeči. Rovněž popisuje splnění jednotlivých cílů disertační práce a předkládá další možné směry výzkumu navazující na tuto práci.

Klíčová slova: hlasové dialogové systémy; porozumění mluvené řeči; detekce sémantických entit; strojové učení; vážené konečné automaty

Annotation

The presented thesis is devoted to the spoken language understanding task. The thesis presents a new discriminative model for this task. First, the spoken language understanding is described in the context of spoken dialog systems and in relation to an automatic speech recognition. Then the state of the art is presented. The current methods for spoken language understanding are presented as well as methods related to the presented discriminative model. In the following chapter, the goals of the thesis are stated. The main goal is to develop a new discriminative model which is able to process uncertain input in the form of word-based or phoneme-based lattices and generate multiple output semantic hypotheses. One of the subgoals of this thesis is devoted to a research of method for effective combination of statistical and knowledge-based approaches to spoken language understanding. The spoken language understanding is decomposed into three partial models. A concept model assigns the global meaning of the utterance in the form of abstract semantic tree. A semantic entity detection tags the local parts of the meaning with the semantic entities. An alignment model links these two semantic representations and forms a discriminative spoken language understanding model. The concept model is represented by the hierarchical discriminative model which was developed as an extension of a statistical model based on semantic tuple classifiers. The semantic entity detection model performs the search for all occurrences of the semantic entities which are defined by knowledge-based context-free grammars. Then, the description of used data, recognition models, speech decoder, and evaluation methodology is presented. In the part devoted to experimental evaluation the values of specific parameters are selected and justified. The last chapter concludes the thesis and presents the overall performance of the presented method for spoken language understanding. It also describes the fulfilment of all goals of this thesis and presents the possible improvements and applications of the developed model.

Keywords: spoken dialog systems; spoken language understanding; semantic entity detection; machine learning; weighted finite state automata

Symbol	Popis (Poznámka)
$\mathcal{V}$, $\mathcal{T}$, $\mathcal{A}$, ...	Množina (kaligrafické písmo, velká písmena)
$\mathbf{o}$, $\mathbf{x}$, $\mathbf{w}$, ...	Vektor (tučný řez písma, malá písmena)
$\mathbf{Q}$, $\mathbf{M}$, $\mathbf{A}$, ...	Matice (tučný řez písma, velká písmena)
$\mathbf{A}^\intercal$	Transpozice matice $\mathbf{A}$
$\mathbf{x} \cdot \mathbf{w}$	Skalární součin vektorů $\mathbf{x}$ a $\mathbf{w}$
T, A, B, ...	Mimo jiné transducery a akceptory (základní řez písma, velká písmena)
O, U, T, ...	Náhodná proměnné (bezpatkové písmo, velká písmena)
$\mathrm{P}(A)$	Pravděpodobnost jevu A (symbol P je v základním řezu písma)
$\mathrm{P}(A\|B)$	Pravděpodobnost jevu A podmíněná jevem B
$\mathbb{R}$	Reálná čísla
$\mathbb{N}$	Přirozená čísla
$\mathbb{K}$	Polookruh $\mathbb{K} = (\mathcal{K}, \oplus, \otimes, \bar{0}, \bar{1})$
Θ	Množina sémantických konceptů
$\{\nu\}$	Speciální množina odpovídající chybějící lexikální realizaci
$\mathcal{V}$	Rozpoznávací slovník (množina slov)
$\mathcal{T}$	Trénovací množina (prvky jsou dvojice (*trénovací příklad*, *cílová třída*), nad prvky trénovací množiny existuje libovolné uspořádání pomocí indexů $i = 1, 2, \ldots l$)
$K(\cdot, \cdot)$	Jádrová funkce
$\mathrm{sgn}(x)$	Znaménková funkce
$\|a\|$	Absolutní hodnota a, pro řetězec a pak jeho délka
ϵ	Prázdný řetězec, u transducerů pak symbol se speciálním významem
$\mathcal{A}^*$	Kleeneho uzávěr množiny $\mathcal{A}$, množina $\mathcal{A}^*$ obsahuje všechny řetězce nad symboly z $\mathcal{A}$ včetně prázdného řetězce ϵ.
$\mathrm{cnt}(A, x)$	(Střední) počet výskytů jevu x ve struktuře A.
$W = (w_1, \ldots, w_N)$	Posloupnost W složená z prvků w_1, w_2 až w_N. Zkrácené zápisy: $W = w_1 w_2 \ldots w_N = (w_i)_{i=1}^N$
$\mathcal{A}$, $\mathcal{B}$	Vstupní, resp. výstupní abeceda transduceru
$\mathcal{Q}$	Množina stavů transduceru
$\mathcal{I}$, $\mathcal{F}$	Množina počátečních, resp. koncových stavů transduceru
$\mathcal{E}$	Množina přechodů transduceru
$\mathcal{P}$	Množina cest transducerem
ϵ, ρ, ϕ, σ	Symboly se speciálním významem
$\oplus$	Součet v daném polookruhu, sjednocení transducerů
$\otimes$	Součin v daném polookruhu, konkatenace transducerů
T^*, T^+	Kleeneho uzávěr, Kleeneho plus transduceru T
T^{-1}	Inverze transduceru T
$T_1 \circ T_2$	Kompozice transducerů T_1 a T_2

Symbol	Popis (Poznámka)
$\Pi_1(T)$, $\Pi_2(T)$	Projekce T na vstupní, resp. výstupní symboly
$\alpha[q]$, $\beta[q]$	Nejkratší vzdálenost z počátečních stavů do stavu q, resp. ze stavu q do koncových stavů transduceru
$F(T)$	Faktorový transducer nad transducerem T
rmeps, det, min	Optimalizační algoritmy nad transducery: odstranění ϵ-přechodů, determinizace, minimalizace
Acc, *Corr*	Slovní přesnost, správnost, četnost chyb
cAcc, *cCorr*	Konceptová přesnost, správnost
BH	Bezplatné Hovory (řečový korpus)
HDM	Hierarchical Discriminative Model, hier. diskr. model
HHTT	Human-Human Train Timetable (řečový korpus)
HMM	Hidden Markov Models, skryté Markovské modely
HVS	Hidden Vector State (parser), parser se skryt. vekt. stavem
NLP	Natural Language Processing, zpracování přiroz. jazyka
OOT	Out-of-topic, (věty) mimo téma
OOV	Out-of-vocabulary, (slova) mimo slovník
STC	Semantic Tuple Classifiers, klasifikátory sémantických n-tic
SVM	Support Vector Machines
TIA	Telefonní Inteligentní Asistentka (řečový korpus)

Obsah

1 Úvod

Již v roce 1950 se Alan Turing ve své práci „Computing Machinery and Intelligence" [1, str. 442] zamýšlel nad otázkou, zda stroje mohou přemýšlet. Z této práce pochází i následující pasáž:

> *... at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted.*

> *... na konci století budou mluva a poučené obecné mínění pozměněné natolik, že bude možno mluvit o myslících strojích, aniž by se mluvčí musel obávat nesouhlasu.*

Jak je podotknuto v knize [2], i přes dlouhá desetiletí výzkumu v oblasti porozumění mluvené řeči není stále možné považovat současný stav za konečný. V reakci na výše zmíněný citát Alana Turinga její autor Gokhan Tür uvádí:

> *Yet, now we are well past the year 2000, and we wonder whether he meant the end of 21st century when machines will be able to „understand" us.*

> *Nyní, kdy je rok 2000 již delší dobu za námi, se musíme ptát, zda nebyl myšlen konec 21. století jako doba, kdy nám stroje budou schopny „porozumět".*

Hlavním tématem disertační práce je strojové porozumění mluvené řeči. Práce popisuje nový model pro porozumění řeči v omezené doméně za účelem použití v hlasových dialogových systémech. Výsledný model umožňuje efektivně kombinovat jak expertní znalost o dané úloze, tak znalosti (a modely) získané statistickými metodami z trénovacích dat. V průběhu řešení bylo nutné zkombinovat celou řadu různých přístupů.

Již v 50. letech 20. století se zformovala dvě paradigma pro zpracování přirozeného jazyka: přístup založený na automatech a přístup pravděpodobnostní [3]. První paradigma bylo reprezentováno osobnostmi jako Alan Turing (položil základy moderní informatiky), Warren McCulloch a Walter Pitts (popsali zjednodušený model neuronu – perceptron), Stephen Cole Kleene (objevitel regulárních výrazů) nebo Avram Noam Chomsky (např. popsal Chomského hierarchii formálních jazyků). Druhé paradigma pak těžilo především z teorie zašuměného kanálu, se kterou přišel Claude

Elwood Shannon. Touto teorií je proces porozumění řeči modelován jako přenos jejího významu pomocí akustického komunikačního kanálu.

Na konci 50. let a v průběhu 60. let 20. století se zpracování řeči a přirozeného jazyka velmi čistě rozdělilo do dvou směrů: symbolického a stochastického. K symbolickému přístupu přispěla jednak práce Chomského a dalších v oblasti teorie formálních jazyků a dále rozvoj symbolické umělé inteligence využívající modely usuzování a formální logiku. Symbolický přístup vedl k prvním modelům pro porozumění mluvené řeči [3]. Stochastický přístup používal Bayesovské metody pro modelování charakteristik pozorovaných jevů. Byl použit nejprve pro rozpoznávání textů (optical character recognition), v 70. letech však byl stochastický přístup použit pro rozpoznávání řeči. Modely pro automatické rozpoznávání řeči využívající skryté Markovské modely spolu s modelem zašuměného kanálu byly využívány především na pracovišti Thomas J. Watson Research Center firmy IBM. Tento přístup k rozpoznávání řeči je nerozlučně spjat se jménem českého emigranta Bedřicha Jelínka (v USA začal používat jméno Frederick Jelinek) a s jeho prací *Continuous speech recognition by statistical methods* [4]. Jeho přístup byl v tomto směru naprosto inovativní, neboť již v roce 1957 A. N. Chomsky napsal, že [5]:

> *We are forced to conclude that grammar is autonomous and independent of meaning, and that probabilistic models give no insight into the basic problems of syntactic structure.*
>
> *Jsme nuceni konstatovat, že gramatika je autonomní a nezávislá na významu a že pravděpodobnostní modely nedávají žádný náhled do základní problematiky syntaktických struktur.*

Přestože v 70. letech byl tento názor Chomského brán jako axiom komputační lingvistiky, F. Jelinek dokázal přijít se statistickým modelem, který je, i přes velkou snahu celé komunity vědců věnující se rozpoznávání řeči, stále nejrozšířenějším typem modelu [5].

V 80. letech 20. století se objevují aplikace konečně stavových modelů v oblasti zpracování přirozeného jazyka. Na základě úspěchů v rozpoznávání řeči jsou statistické metody používány i v dalších oblastech komputační lingvistiky.

V 90. letech 20. století a na počátku 21. století pak dochází k aplikaci teorie vážených konečných automatů v oblasti rozpoznávání a porozumění řeči. Jmenujme například práce, jejímiž autory jsou Mehryar Mohri, Fernando C. N. Pereirra a Michael Riley, popř. Cyril Allauzen [6, 7, 8]. Vážené konečné transducery působí jako jednotící prvek spojující symbolické a statistické paradigma. Například – znalosti v podobě regulárních nebo bezkontextových gramatik speciálního typu je možné

převést do podoby váženého konečného transduceru. Vážené konečné transducery také umožňují reprezentovat generativní modely (typicky skryté Markovské modely) [9] a s využitím teorie racionální jádrových funkcí je lze použít i v modelech diskriminativních [10].

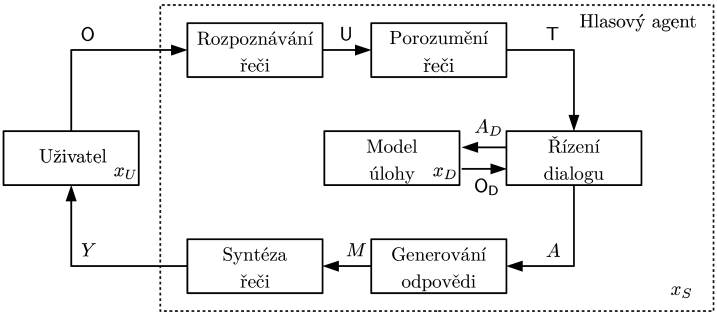
Modely pro porozumění řeči je možné rozdělit do tří kategorií [2]. Systémy náležející do první z těchto kategorií se ve skutečnosti o porozumění ani nesnaží, pouze jej předstírají. Typickým zástupcem této třídy je systém ELIZA [11], který pomocí jednoduchých transformačních pravidel aplikovaných na uživatelův vstup generoval svůj výstup. Druhá kategorie vychází z teorie (symbolické) umělé inteligence. Tyto systémy jsou založeny na formální reprezentaci znalostí a na formální sémantické interpretaci. Provádějí mapování věty na její reprezentaci ve zvolené formální logice. Ukázalo se však, že tyto systémy jsou vhodné pouze pro velmi omezené domény.

Systémy spadající do třetí kategorie redukují porozumění řeči na problém zpracování přirozeného jazyka založený zpravidla na jeho statistickém zpracování. V současné době je snaha o úspěšné globální porozumění mluvené řeči otevřený problém, nicméně porozumění řeči v dané problémové oblasti (doméně) je řešitelné. Je však nutné poznamenat, že porozumění mluvené řeči není pouze jediná technologie, ale existuje celá řada přístupů vhodných pro konkrétní nasazení. Z důvodů chyb při automatickém rozpoznávání řeči není zpravidla možné použít obecný rozpoznávač řeči a jeho textový výstup použít v obecných algoritmech zpracování přirozeného jazyka. Je nutné brát v úvahu neurčitost vznikající při rozpoznávání řeči a tuto neurčitost převést i v neurčitost významových hypotéz.

2 Hlasové dialogové systémy

Z kybernetického úhlu pohledu můžeme na hlasový dialogový systém pohlížet jako na systém o dvou subsystémech, přičemž řeč tvoří komunikační prostředek mezi těmito dvěma subsystémy. Jeden z těchto subsystémů nazývejme *uživatel* a druhý *hlasový agent*. Uživatel, zpravidla člověk, užívá služeb hlasového agenta pro vykonání svých cílů a proto jej instruuje (řídí) tak, aby těchto cílů dosáhl. Na druhou stranu hlasový agent pro splnění cílů často musí řídit uživatele takovým způsobem, aby od něj získal požadované informace a mohl splnit jeho cíl.

Pro účely modelování hlasového dialogového systému definujme *dialogový akt* (angl. *dialogue act*) jako základní jednotku hlasového dialogu. Celý dialog se skládá z výměny jednotlivých dialogových aktů mezi komunikujícími stranami. Dialogový akt jedné strany je zpravidla následován dialogovým aktem druhé strany a naopak, nicméně mohou existovat i dva a více zřetězených dialogových aktů jedné komunikující strany, navíc bez explicitního odlišení nebo oddělení jednotlivých dílčích dialogových aktů.



Obrázek 1: Model hlasového dialogového systému.

Model hlasového dialogu mezi uživatelem a agentem v nejjednodušší podobě je zobrazen na obrázku 1. V tomto modelu uvažujeme existenci následujících podsystémů – modulu *rozpoznávání řeči* a modulu *porozumění řeči*, modulu *generování odpovědi* a *syntézy řeči*, modulu *řízení dialogu* a *modelu úlohy*. Zatímco rozpoznávání a porozumění řeči převádí uživatelův akustický řečový signál o do strojové reprezentace dialogového aktu t, generování odpovědi zpracovává výstupní dialogové akty agenta A na akustický řečový signál Y. Modul řízení dialogu generuje na základě stavu úlohy x_D a stavu agenta nový dialogový akt agenta x_S. Zároveň interaguje s modelem úlohy, odkud získává informace potřebné ke splnění cíle dialogu [12, 13].

Předpokládejme, že hlasový dialog je zahájen agentem prostřednictvím dialogového aktu A, který je v modulu *generování odpovědi* převeden na slovní realizaci M a v modulu *syntézy řeči* je vysyntetizovaná řeč Y odpovídající aktu A.

Uživatel pak na základě svého stavu x_U a signálu Y provede aktualizaci svého stavu a vygeneruje svojí promluvu o. Tato promluva je zpracována subsystémem *automatického rozpoznávání řeči* na rozpoznané jednotky u. Tyto jednotky jsou zpravidla tvořeny slovy, nicméně je možné uvažovat i o subslovních jednotkách jako jsou fonémy nebo slabiky popřípadě i o jiném způsobu reprezentace promluvy. Těmto jednotkám je následně v *modulu porozumění mluvené řeči* přiřazen významový popis t – dialogový akt uživatele.

Poznamenejme, že o, u a t jsou zatíženy neurčitostí, proto jsou zpravidla reprezentovány pomocí pravděpodobnostního rozdělení $\mathrm{P}(\mathsf{O} = o)$, $\mathrm{P}(\mathsf{U} = u|\mathsf{O} = o)$ a $\mathrm{P}(\mathsf{T} = t|\mathsf{U} = u, \mathsf{O} = o)$. Výstupem modulu porozumění řeči je pak pravděpodobnostní rozdělení $\mathrm{P}(\mathsf{T}|\mathsf{U}, \mathsf{O})$, se kterým pracuje modul řízení dialogu, který na základě $\mathrm{P}(\mathsf{T}|\mathsf{U}, \mathsf{O})$, svého stavu x_S a *strategie řízení* π vygeneruje dialogový akt agenta $A = \pi(x_S, \mathrm{P}(\mathsf{T}|\mathsf{U}, \mathsf{O}))$. Dialogový akt agenta je následně opět převeden na

akustický signál a cyklus se opakuje. V průběhu interakce s uživatelem si agent na základě pozorování O aktualizuje svůj stav x_S. Modul řízení dialogu nemusí nutně provádět interakci pouze s uživatelem, ale může také prostřednictvím akcí A_D řídit model úlohy a pozorovat výstupy tohoto modelu O_D. Stav modelu úlohy x_D je pozorován prostřednictvím náhodné proměnné O_D, přičemž model úlohy může, ale nutně nemusí být plně pozorovatelný. V praxi je často modelem úlohy velmi rozsáhlá databáze např. vlakových spojení a z důvodu enormního nárůstu počtu stavů je nepraktické tuto databázi zahrnovat přímo do hlasového agenta, ale je vhodnější vyčlenit ji jako model úlohy.

Klíčovým modulem hlasového dialogového systému je subsystém rozpoznávání řeči a subsystém porozumění řeči. Na přesnosti rozpoznávání a porozumění řeči je závislá efektivita celého hlasového dialogového systému. Přestože je možné jako výsledek procesu rozpoznávání a porozumění získat více hypotéz o téže promluvě, chyby vzniklé při tomto procesu je nutné napravit na úrovni modulu řízení dialogu pomocí interakce s uživatelem (zotavení z chyby). To však prodlužuje a zpomaluje samotný průběh dialogu. Poznamenejme, že chybný návrh rozpoznávání a nebo porozumění řeči může znemožnit zadání některých vstupů bez ohledu na modul řízení dialogu a tím velmi negativně ovlivnit celkovou použitelnost systému.

2.1 Rozpoznávání řeči

Rozpoznávání řeči využívající statistických metod lze formulovat jako úlohu *dekódování podle maximální aposteriorní pravděpodobnosti* [3, 14]. Nechť náhodná proměnná $\mathsf{U} = \mathsf{W}_1\mathsf{W}_2 \ldots \mathsf{W}_N$ je tvořena posloupností N náhodných proměnných W_i reprezentujících jednotlivá slova nebo jiné jednotky promluvy.[1] Posloupnost $o = \{\mathbf{o}_1, \mathbf{o}_2, \ldots \mathbf{o}_T\}$ je posloupnost vektorů příznaků akustického řečového signálu. K tomu jsou použity metody akustické analýzy založené v současnosti nejčastěji na využití Melovských kepstrálních koeficientů (MFCC, [15]) či perceptivní lineární prediktivní analýzy (PLP, [16]).

Cílem rozpoznávání řeči je nalézt takovou posloupnost $\hat{u}$, která maximalizuje aposteriorní pravděpodobnost $\mathrm{P}(\mathsf{U} = u|\mathsf{O} = o)$:

$$\hat{u} = \arg\max_u \mathrm{P}(\mathsf{U}|\mathsf{O}) = \arg\max_u \frac{\mathrm{P}(\mathsf{U})\mathrm{P}(\mathsf{O}|\mathsf{U})}{\mathrm{P}(\mathsf{O})} = \arg\max_u \mathrm{P}(\mathsf{U})\mathrm{P}(\mathsf{O}|\mathsf{U})$$

Tímto jsme aposteriorní pravděpodobnost nahradili součinem dvou modelů – *jazykového modelu* $\mathrm{P}(\mathsf{U})$ a *akustického modelu* $\mathrm{P}(\mathsf{O}|\mathsf{U})$. Tyto modely je možné trénovat

[1] Na rozdíl od literatury [3] nebo [14] je v disertační práci posloupnost rozpoznávaných jednotek značena jako U, nikoli W, neboť výstupem systému automatického rozpoznávání řeči nemusí nutně být slova – lze uvažovat i systémy rozpoznávající posloupnosti fonémů nebo slabik.

nezávisle na sobě a každý z nich nese jinou část znalostí o řešené úloze. Akustický model $\mathrm{P}(\mathsf{O} = o|\mathsf{U} = u)$ vyjadřuje pravděpodobnost pozorování posloupnosti o při uvažování, že byla pronesena slova u. Jazykový model $\mathrm{P}(\mathsf{U} = u)$ pak vyčísluje apriorní pravděpodobnost výskytu posloupnosti u tvořené slovy nebo jinými jednotkami.

Výstupem systému automatického rozpoznávání nemusí být pouze první nejlepší hypotéza, ale rozložení pravděpodobnosti $\mathrm{P}(\mathsf{U}|\mathsf{O})$. Efektivní reprezentací tohoto rozložení je tzv. mřížka (angl. lattice) [17, 18]. Pro odlišení typu jednotek obsažených v mřížce budeme používat i spojení slovní mřížka nebo fonémová mřížka. Mřížka je acyklický graf, který reprezentuje různé hypotézy u odpovídající vstupní posloupnosti o a přiřazuje jim pravděpodobnost $\mathrm{P}(\mathsf{U} = u|\mathsf{O} = o)$. Velmi častou reprezentací mřížek jsou vážené konečné akceptory [6]. Uvažujme, že pravděpodobnostnímu rozložení $\mathrm{P}(\mathsf{U}|\mathsf{O})$ odpovídá vážený konečný akceptor U nad pravděpodobnostním polookruhem. Potom $\oplus$-suma vah všech cest π z počátečního stavu U do některého z koncových stavů U a se vstupními symboly u (tj. $i[\pi] = u$) odpovídá pravděpodobnosti $\mathrm{P}(\mathsf{U} = u|\mathsf{O} = o)$:

$$\mathrm{P}(\mathsf{U} = u|\mathsf{O} = o) = \bigoplus_{\pi \in U : i[\pi] = u} w[\pi] \tag{1}$$

Pro vyhodnocení přesnosti systémů automatického rozpoznávání řeči se používá postupu, kdy rozpoznaná promluva (hypotéza) je nejprve *zarovnána* s referenční transkripcí vytvořenou anotátorem. Pro zarovnání se používá zpravidla algoritmu pro výpočet Levenshteinovy vzdálenosti [19], přičemž průchodem posloupností editačních operací, která vede na minimální editační vzdálenost, jsou získána následující čísla:

- H – počet správně rozpoznaných slov
- S – počet slov, která jsou chybně rozpoznaná jako jiná slova
- D – počet slov chybějících v rozpoznané hypotéze
- I – počet slov přebývajících v rozpoznané hypotéze
- N – počet slov v referenční transkripci

Potom lze definovat míru nazvanou *přesnost* (accuracy, *Acc*):

$$Acc = \frac{N - D - S - I}{N} \tag{2}$$

Tato míra bude použita v experimentální části disertační práce pro vyhodnocení přesnosti subsystému automatického rozpoznávání řeči. Na této míře je rovněž založeno odvození míry pro vyhodnocení přesnosti systému porozumění mluvené řeči.

2.2 Porozumění mluvené řeči

Cílem porozumění mluvené řeči je na základě pravděpodobnostního rozdělení $P(\mathsf{U}|\mathsf{O})$ sestavit pravděpodobnostní rozdělení $P(\mathsf{T}|\mathsf{U},\mathsf{O})$. V praxi je často uvažována aproximace, která předpokládá, že rozpoznané jednotky U obsahují veškerou informaci o významu T a tudíž $P(\mathsf{T}|\mathsf{U},\mathsf{O}) \approx P(\mathsf{T}|\mathsf{U})$.

Konkrétní hodnoty t náhodné proměnné T mohou mít různou strukturu. Nejčastěji se jedná o seznam párů *atribut:hodnota* [20] nebo o *sémantické stromy*. Uzly sémantických stromů, případně atributy v seznamu párů jsou tvořeny *sémantickými koncepty*. Samotný pojem *koncept* přibližuje následující definice:

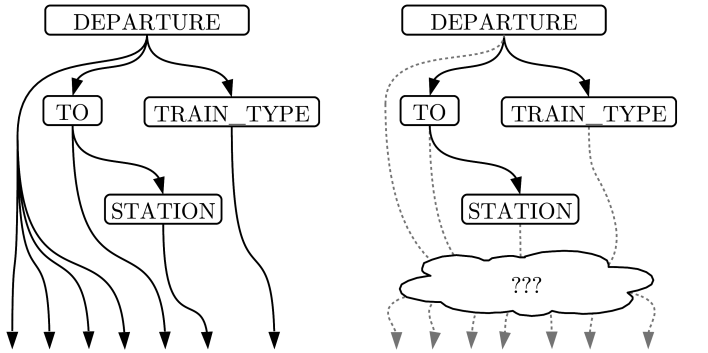
> *[Concept is] an idea or mental image which corresponds to some distinct entity or class of entities, or to its essential features, or determines the application of a term (especially a predicate), and thus plays a part in the use of reason or language.*

> *[Koncept je] představa nebo mentální obraz, který odpovídá nějaké jedinečné entitě nebo třídě entit nebo jejím základním vlastnostem, popřípadě určuje způsob použití jednotlivých termínů (především predikátů) a tudíž hraje roli v uvažování nebo v jazyce jako takovém.*
>
> [The New Oxford Dictionary of English]

Sémantickými koncepty budeme nazývat značky (tagy), které odlišují různé významy promluv a různé třídy entit v rámci jednotlivých promluv. Sémantické koncepty budeme vždy považovat za doménově závislé, množina sémantických konceptů bude jiná pro úlohu inteligentní asistentky a jiná pro úlohu navigačního software. Některé sémantické koncepty mohou mít přiřazenu konkrétní hodnotu – *sémantickou entitu*. U dalších sémantických konceptů pak konkrétní lexikální realizace je nepodstatná, pro samotný význam promluvy je důležitá pouze jejich přítomnost nebo nepřítomnost.

Sémantická entita odpovídá konkrétní lexikální (slovní) realizaci v dané promluvě. Sémantická entita je reprezentována svým typem a interpretací (kapitola 7, str. 29). Typem sémantické entity může být například datum, čas, jméno apod. Interpretace dále doplňuje sémantickou entitu o konkrétní hodnoty a slouží jako obraz reálného objektu v rámci modelu hlasového agenta.

Pro reprezentaci významu t budeme používat strukturu *sémantického stromu*. Sémantický strom vyjadřuje hierarchickou závislost mezi jednotlivými sémantickými koncepty a slovy vstupní promluvy. Budeme proto používat i ekvivalentní termín *zarovnaný sémantický strom*. Sémantické koncepty blíže kořenu sémantického stromu



Obrázek 2: Zarovnaný sémantický strom (vlevo) a abstraktní sémantický strom (vpravo). Často je používán i linearizovaný zápis abstraktního sémantického stromu ve tvaru: DEPARTURE(TO(STATION), TRAIN_TYPE). Obdobně linearizovaný zápis zarovnaného sémantického stromu je ve tvaru: DEPARTURE(*v kolik to jede* TO(*na* STATION(*cheb*)) TRAIN_TYPE(*rychlíky*)).

jsou zpravidla obecnější, sémantické koncepty dále od kořenu pak specifičtější. V listech zarovnaného sémantického stromu jsou uložena slova – lexikální realizace jednotlivých konceptů.

Dále budeme používat i termín *abstraktní sémantický strom* (popř. nezarovnaný sémantický strom). Abstraktní sémantické stromy popisují pouze množinu a strukturu sémantických konceptů přiřazených dané promluvě bez vazby na sémantické entity nebo na slova promluvy. Budeme též říkat, že abstraktní sémantické stromy jsou nezarovnané s původní promluvou, konceptům na základě abstraktního sémantického stromu nelze přiřadit konkrétní hodnoty [21, 22, 23]. Příklad sémantického stromu a abstraktního sémantického stromu je uveden na obrázku 2.

Na tomto obrázku jsou vyobrazeny sémantické stromy sestavené ze sémantických konceptů DEPARTURE, TO, TRAIN_TYPE a STATION. Z pohledu hlasových dialogových systémů můžeme koncepty DEPARTURE a TO uvažovat jako koncepty, u nichž nezáleží na jejich lexikální realizaci, pro řízení dialogu je významná pouze jejich přítomnost v sémantickém stromu. Oproti tomu koncepty TRAIN_TYPE a STATION se pojí se sémantickými entitami *train_type:R* a *station:cheb* reprezentující objekt „rychlík” a objekt „stanice Cheb”.

Odtud vyplývá potřeba definovat poslední z používaných termínů – *částečně zarovnaný sémantický strom*. V těchto stromech je pouze některým sémantickým koncep-

tům přiřazena konkrétní lexikální realizace nebo konkrétní část vstupní promluvy. Použijeme-li příklad stromu na obrázku 2, můžeme částečně zarovnaný sémantický strom zapsat jako DEPARTURE(TO(STATION(*cheb*)), TRAIN_TYPE(*rychlíky*)). Zde je sémantický strom zarovnán pouze se slovy *cheb* a *rychlíky*. Poznamenejme, že částečně zarovnaný sémantický strom nemusí mít ve svých listech uloženy lexikální realizace, ale například sémantické entity získané ze vstupní promluvy.

V případě, kdy nebude nutné detailně rozlišovat mezi zarovnanými, nezarovnanými (abstraktními) a částečně zarovnanými sémantickými stromy, budeme používat jediný společný termín „sémantický strom" a až v případě nutnosti budeme tyto případy rozlišovat vhodným přídavným jménem.

3 Cíle disertační práce

Popis architektury hlasových dialogových systémů a jednotlivých metod pro porozumění řeči a zpracování mluvené řeči v předchozí kapitole pak slouží jako motivace ke stanovení jednotlivých cílů disertační práce. Tyto cíle vyplynuly především z praktických zkušeností autora při výzkumu, vývoji a nasazení hlasových dialogových systémů a dalších technologií pro zpracování řeči – především systémů automatického rozpoznávání řeči a systémů pro indexaci a vyhledávání klíčových slov v audiovizuálních archívech.

Samotná disertační práce obsahuje kapitolu věnovanou přehledu současného stavu řešené problematiky, kde jsou blíže popsány související metody, modely a přístupy nejen k porozumění řeči.

Cíl 1: *Vyvinutí modelu porozumění schopného pracovat s neurčitostí vstupu i výstupu*

S ohledem na cílové nasazení v oblasti hlasových dialogových systému bylo prvním z cílů vyvinout model, který umožňuje efektivně pracovat s neurčitostí vzniklou při rozpoznávání řeči. A to nejen ve smyslu schopnosti generovat více hypotéz o významu vstupní promluvy, ale i v možnosti zpracovávat mřížku obsahující více hypotéz o slovním přepisu vstupní promluvy.

Cíl 2: *Využití fonémového rozpoznávače v oblasti porozumění řeči*

Při vývoji hlasového dialogového systému, nebo obecněji libovolného systému automatického rozpoznávání řeči, je největší překážkou potřeba získat dostatečné množství dat pro robustní jazykový model. Tento jazykový model musí dostatečným způsobem pokrývat slovník dané úlohy. Navíc možnosti přenesení znalostí mezi jednotlivými doménami jsou omezené. V projektech řešených autorem zaměřených na hledání klíčových slov a frází však byly s úspěchem použity metody pro rozpoznávání

řeči na fonémové úrovni. Přestože tyto metody nedosahovaly přesnosti slovních modelů, tvoří jejich použití zajímavou alternativu k rozpoznávání na úrovni slov právě kvůli náročnosti přípravy slovního jazykového modelu. Proto dalším z cílů je výzkum v oblasti využití rozpoznávání řeči na fonémové úrovni za účelem porozumění řeči a získání významového popisu bez znalosti konkrétních slov vyskytujících se v dané úloze. Jelikož i fonémový rozpoznávač řeči vyžaduje jazykový model na fonémové úrovni, bude se disertační práce věnovat i možnostem adaptace fonémového jazykového modelu.

Cíl 3: *Formulace plně pravděpodobnostního diskriminativního modelu*

Předchozí výzkum v oblasti hlasových dialogových systémů na pracovišti autora používal generativní modely pro porozumění řeči. Nicméně experimenty ukázaly, že diskriminativní modely umožňují dosáhnout vyšší přesnosti porozumění. Mezi další cíle zahrneme požadavek vyvinout statistický diskriminativní model, který však bude možné použít v plně pravděpodobnostním modelu hlasového dialogového systému. Tento cíl je formulován především s ohledem na budoucí výzkum v oblasti pravděpodobnostních modelů a rozhodovacích procesů pro řízení dialogu.

Cíl 4: *Návrh vhodné metody pro kombinaci statistického a znalostního přístupu*

Přestože statistický přístup k porozumění řeči je schopen naučit se cílové chování z trénovací množiny, tato množina musí mít dostatečný počet reprezentativních příkladů. Tento předpoklad není v praxi vždy splněn. Nabízí se proto využití znalostního přístupu k vyjádření základních, obecně platných znalostí o dané úloze. Využitím znalostí lze redukovat potřebný počet trénovacích dat. Proto bude část disertační práce věnována i tomu, jak vhodně tuto expertní znalost integrovat do plně pravděpodobnostního diskriminativního modelu.

Cíl 5: *Ověření modelu nad více cílovými doménami*

Posledním cílem disertační práce bude ověření vyvinutého modelu nad více než jedním sémanticky anotovaným korpusem dat z důvodu zabránění „přetrénování" modelu na určitou cílovou doménu.

4 Teoretický základ použitých metod

Předkládaná disertační práce staví na množství metod a postupů. Shrňme tedy tyto metody spolu s citacemi původních pramenů.

Nejprve uveďme teorii klasifikátorů založených na *support vector machines* (SVM) [24, 25]. V disertační práci jsou popsána i rozšíření SVM klasifikátorů pro klasifikaci

do více cílových tříd [26] a metoda pro odhad aposteriorní pravděpodobnosti příslušnosti daného vektoru příznaků do cílové třídy [27]. Klasifikátory založené na SVM jsou použity ve skryté a výstupní vrstvě hierarchickém diskriminativním modelu.

Další popsanou teoretickou oblastí jsou *vážené konečné automaty*. Jsou popsány algoritmy a operátory pro práci s váženými konečnými automaty [8, 28]. Je zde rovněž zmíněn faktorový automat [29] jako nástroj pro efektivní indexaci všech podřetězců zdrojového automatu. Tyto struktury jsou použity pro reprezentaci slovních a fonémových mřížek na výstupu ze systému automatického rozpoznávání řeči. Rovněž slouží k efektivnímu výpočtu racionálních jádrových funkcí (kapitola 6.1) a také k detekci sémantických entit.

Následuje výklad teorie *racionální jádrových funkcí*, které umožňují vyčíslení jádrové funkce mezi dvěma mřížkami [10] a tím pádem lze s jejich využitím natrénovat SVM klasifikátor nad trénovací množinou reprezentovanou mřížkami, nikoli příznakovými vektory. Racionální jádrové funkce jsou použity ve vstupní vrstvě hierarchického diskriminativního modelu.

Další text popisuje *stochastické bezkontextové gramatiky* [3]. Jsou zmíněny i lexikalizované gramatiky [30], které slouží jako základ pro sémantické gramatiky použité v hierarchickém diskriminativním modelu.

Výklad *n-gramových jazykových modelů pro rozpoznávání řeči* je využit v části věnované adaptaci fonémových jazykových modelů na novou doménu.

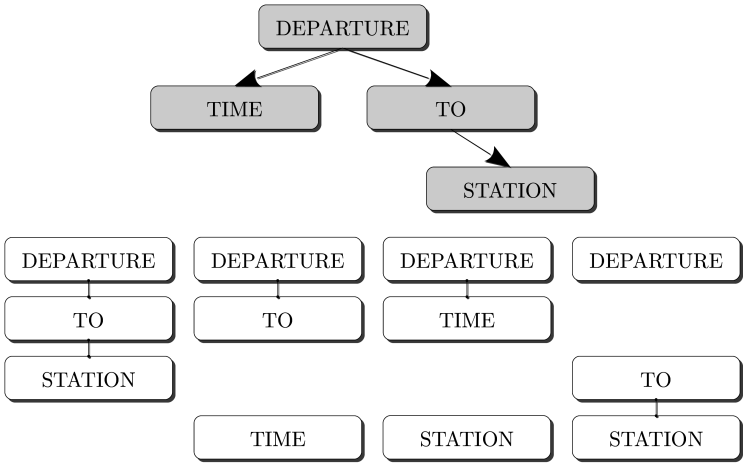
Poslední dvě kapitoly *parser se skrytým vektorovým stavem* [22, 31, 32] a *klasifikátory sémantických n-tic* [33] popisují referenční modely, k nimž jsou vztaženy experimentální výsledky. Tyto modely byly vybrány záměrně – parser se skrytým vektorovým stavem jako reprezentant třídy generativních modelů, klasifikátory sémantických n-tic pak jako představitel diskriminativních modelů.

4.1 Klasifikátory sémantických n-tic

Klasifikátory sémantických n-tic (Semantic tuple classifiers, STC) je přístup k porozumění mluvené řeči vyvinutý na Univerzitě v Cambridge v roce 2009 [33]. Autoři prezentují jednoduchou techniku, která využívá množiny natrénovaných klasifikátorů, které diskriminují jednotlivé sémantické koncepty. Struktura cílových tříd a předzpracování vstupní promluvy umožňuje rekonstrukci sémantického stromu, při trénování není nutná informace o zarovnání konceptů sémantického stromu se vstupní promluvou, STC model se trénuje z abstraktních sémantických anotací.

Vstupem trénovacího algoritmu STC je množina promluv a odpovídajících abstraktních sémantických stromů. Každý sémantický strom je rozdělen na tzv. *sémantické*

n-tice, které mohou být chápány jako podposloupnosti o maximální délce k konceptů. Podposloupnosti jsou tvořeny z posloupností konceptů na cestě z kořene sémantického stromu do libovolného uzlu. Příkladem budiž obrázek 3 zobrazující dekompozici abstraktního sémantického stromu na sémantické n-tice.



Obrázek 3: Sémantický strom (šedá barva) s abstraktní sémantickou anotací DEPARTURE(TIME, TO(STATION)) a jeho dekompozice na sémantické n-tice délky 1 až 3 (bílá barva).

Předpokládejme, že trénovací množina se skládá z dvojic vstupní promluva u_i a odpovídající abstraktní sémantický strom s_i, tj. $\mathcal{T} = \{(u_i, s_i)\}_{i=1}^{l}$. Říkejme, že sémantická n-tice $\mathbf{t}$ náleží do abstraktního sémantického stromu s (tj. $\mathbf{t} \in s$), je-li $\mathbf{t}$ podposloupnost libovolné cesty z kořenového uzlu stromu s do některého z uzlů stromu s. V opačném případě budeme psát $\mathbf{t} \notin s$. Množinu všech sémantických n-tic budeme označovat jako $\mathcal{S} = \{\mathbf{t} \in s_i; s_i \in \mathcal{T}\}$. Obdobně pro sémantické n-tice délky k označme $\mathcal{S}_k = \{\mathbf{t} \in \mathcal{S}; |\mathbf{t}| = k\}$.

Algoritmus pro trénování modelu založeného na klasifikátorech sémantických n-tic pak sestává z následujících kroků [33]:

1. Náhrada všech výskytů sémantických entit odpovídajícími identifikátory.
2. Výpočet lexiko-syntaktických příznaků $\mathbf{x}_i$ pro každou promluvu z množiny $\mathcal{T}$.
3. Pro všechny sémantické n-tice $\mathbf{t}_j \in \mathcal{S}_k$:
   (a) Vytvořit trénovací množinu pro trénování klasifikátoru $\mathcal{T}_j = \{(\mathbf{x}_i, y_i^j\}$, kde $y_i^j = 1$ pokud $\mathbf{t}_j \in s_i$, jinak $y_i^j = -1$.

(b) Využití trénovací množiny $\mathcal{T}_j$ pro trénování binárního klasifikátoru realizujícího funkci $\hat{y}^j = C_j(\mathbf{x})$. V původní práci [33] byly použity binární klasifikátory založené na SVM.

4. Sestavení doménové gramatiky, která generuje všechny sémantické stromy z trénovací množiny $\mathcal{T}$.

Pro dekódování abstraktního sémantického stromu pomocí modelu založeného na klasifikátorech sémantických n-tic je pak použit postup:

1. Náhrada všech výskytů sémantických entit v promluvě u odpovídajícími identifikátory tříd.
2. Výpočet lexiko-syntaktických příznaků $\mathbf{x}$ z promluvy u, přičemž se odstraní ty příznaky, které nebyly pozorovány ve fázi trénování.
3. Pro každé $C_j \in \mathcal{C}$ predikce $\hat{y}^j = C_j(\mathbf{x})$. Sestavení množiny sémantických n-tic odpovídajících promluvě u:

$$\hat{S} = \{\mathbf{t}_j \in S_k : \hat{y}^j = 1\} \tag{3}$$

4. Nalezení odpovídajícího abstraktního sémantického stromu $\hat{s}$ k množině predikovaných sémantických n-tic $\hat{S}$. Algoritmus nejprve vytvoří výstupní strom $\hat{s}$ obsahující pouze kořenový uzel odpovídající startovacímu symbolu doménové gramatiky G. Dále nastaví proměnnou r ukazující na tento uzel. Autoři v [33] popisují dva možné módy algoritmu:
   (a) *Mód s vysokou přesností* – pro každou n-tici $\mathbf{t} = (t_1, t_2, \ldots t_n) \in \hat{S}$, pro kterou $t_1 = r$, přidej do stromu s uzly $t_2, \ldots t_n$ uspořádané tak, že $r = t_1$ a t_{i-1} je předchůdce t_i. Odstraň $\mathbf{t}$ z $\hat{S}$. Rekurzivně opakuj nastavováním r na všechny nezpracované uzly z $\hat{s}$.
   (b) *Mód s vysokou úplností* – vytvoř strom $\hat{s}$ podle (a). Zbývající n-tice z $\hat{S}$, které již není možné do $\hat{s}$ přidat, zpracuj následujícím způsobem:
   Rekurzivně pro všechny uzly r ze stromu $\hat{s}$ a pro všechny zbývající n-tice $\mathbf{t} = (t_1, t_2, \ldots t_n) \in \hat{S}$ hledej v doménové gramatice n-tici $(r, t_1, \ldots t_n)$. Pokud taková n-tice existuje, přidej do stromu s uzly $t_1, \ldots t_n$ uspořádané tak, že r je předchůdce t_1 a t_{i-1} je předchůdce t_i.
5. Zarovnání abstraktního sémantického stromu s odpovídajícími identifikátory lexikálních tříd.
6. Zpětné nahrazení identifikátorů tříd za odpovídající slova vstupní promluvy.

Doménově závislá databáze sémantických entit přináší do úlohy expertní znalost. Je prezentována jako seznam dvojic *sémantický identifikátor/posloupnost slov*, např. STATION = *Plzeň* nebo STATION = *Ústí nad Labem*. Využití expertní znalosti umožňuje rozšíření modelu na úlohy, kde se vyskytuje velký počet různých

hodnot jednotlivých konceptů. Před vyčíslením příznakového vektoru pro libovolnou vstupní promluvu, jak ve fázi trénování, tak ve fázi dekódování, je provedena náhrada podposloupností vstupní promluvy za odpovídající sémantické identifikátory. V případě víceznačného mapování nebo překryvů je vybrán ten sémantický identifikátor, který se shoduje v největším počtu slov.

Parametr l určující maximální délku sémantické n-tice řídí zároveň i vyvážení mezi přesností jednotlivých sémantických klasifikátorů a nejednoznačností generovaného sémantického stromu. Příliš dlouhé sémantické n-tice vedou na triviální rekonstrukci sémantického stromu, ale za cenu nízké přesnosti predikce výskytu dané sémantické n-tice. Autoři původního STC modelu používali fixní délku l sémantických n-tic. Na referenční databázi ATIS [34] bylo nastaveno $l = 3$. Příznakový vektor $\mathbf{x}(u)$ byl získán jako četnost různých n-gramů ($1 \leq n \leq 3$) v promluvě u, což vede na příznakový vektor obsahující desítky tisíc položek. Autoři proto použili SVM klasifikátory s lineární jádrovou funkcí. Celkový počet klasifikátorů byl přibližně 250 a doba zpracování neznámé promluvy byla průměrně méně než 200 ms.

Zkušenosti získané při reimplementaci STC modelu byly hlavní motivací pro výzkum popsaný v disertační práci. Výsledkem je pak diskriminativní model porozumění řeči, který STC model rozšiřuje takřka ve všech směrech:

- *Náhrada posloupností slov identifikátory lexikálních tříd v rozpoznané promluvě.* V rámci disertační práce došlo k úpravě tohoto nahrazování pomocí detekce sémantických entit, která může být navíc realizována nad neurčitým výstupem z rozpoznávání řeči ve tvaru mřížky.
- *Možnost využít celé slovní (a dokonce fonémové) mřížky pro trénování a přiřazování sémantických stromů.* Původní STC model oproti tomu umožňoval zpracování pouze první nejlepší slovní hypotézy.
- *Modelování vzájemně korelovaných výstupů jednotlivých klasifikátorů sémantických entit.* V disertační práci je navrženo rozšíření, které přidává další vrstvu diskriminativních klasifikátorů, která tuto korelaci bere v úvahu.
- *Schopnost generovat více výstupních sémantických stromů (významových hypotéz) s přiřazenými aposteriorními pravděpodobnostmi.* Navíc takto generované sémantické stromy jsou v rámci dále popsané struktury diskriminativního modelu kombinovány s expertní znalostí zahrnutou v detekci sémantických entit.

5 Diskriminativní model pro porozumění mluvené řeči

Přestože diskriminativních modelů pro porozumění řeči existuje celá řada, pokusíme se popsat jeden z možných nových pohledů na tuto problematiku. Jádrem tohoto

pohledu je konceptový model, který ke vstupní promluvě přiřazuje abstraktní sémantický strom. Následně je konceptový model doplněn o detekci sémantických entit a o model zarovnání, které provádějí částečné zarovnání lexikální realizace vstupní promluvy s abstraktním sémantickým stromem. Tuto myšlenku v nejjednodušším provedení lze nalézt již u modelu popsaného dříve – u klasifikátorů sémantických entit. Zde samotné klasifikátory a výstupní heuristika přiřazují výstupní sémantický strom, který je následně s pomocí lexikálních tříd a jednoduchých pravidel zarovnán se vstupní promluvou. Tuto myšlenku dále rozvineme do podoby plně pravděpodobnostního diskriminativního modelu, porozumění řeči je pak prováděno na dvou úrovních:

- *Porozumění nižší úrovně* (lokální) – hledá v promluvě významové entity, které reprezentují elementární prvky významu. Tyto entity jsou hledány striktně lokálně, bez ohledu na výskyt dalších entit nebo konceptů.
- *Porozumění vyšší úrovně* (globální) – přiřazuje promluvě význam jako celku. Tento význam je reprezentován abstraktním sémantickým stromem. Při globálním pohledu jsou již známy významové (sémantické) entity nalezené v promluvě a lze je tudíž použít při predikci sémantického stromu.

U prezentovaného diskriminativního modelu označme jako T náhodnou proměnnou nad množinou (částečně) zarovnaných sémantických stromů. Uvažujme, že proměnná T se skládá z dílčích náhodných proměnných E a C. Náhodná proměnná E je definována nad všemi různými posloupnostmi sémantických entit. Přesněji E je definováno nad množinou posloupností $e = \{e_1, e_2, \ldots\}$, přičemž každá posloupnost e se skládá z dílčích sémantických entit e_i. Každá sémantická entita reprezentuje nějaký konkrétní objekt zmíněný v dané promluvě a významný z pohledu sémantické analýzy. Pro příklad jmenujme sémantické entity typu čas, které se mohou skládat z údaje o hodinách a minutách (více v kapitole 7).

Náhodná proměnná C je pak definována nad množinou různých významů, například nad množinou všech možných sémantických stromů nebo nad množinou možných sémantických konceptů přiřazených dané promluvě. Náhodnou proměnnou C budeme uvažovat jako náhodnou proměnnou definovanou nad všemi možnými abstraktními sémantickými stromy.

Aposteriorní pravděpodobnost $\mathrm{P}(\mathsf{T} = t|\mathsf{U})$ pak odpovídá sdružené pravděpodobnosti $\mathrm{P}(\mathsf{E} = e, \mathsf{C} = c|\mathsf{U})$. Dále zaveďme binární náhodnou proměnnou A, která nabývá hodnoty 1 pokud je možné posloupnost sémantických entit e a abstraktní sémantický strom c zarovnat tak, aby sémantické entity odpovídaly sémantickým konceptům. Je-li $\mathsf{A} = 1$, pak zarovnaný sémantický strom t skládající se ze sémantických entit e a abstraktního sémantického stromu c je *validní*.

V úloze porozumění mluvené řeči nás budou zajímat právě validní zarovnané sémantické stromy, tj. případy, kdy $\mathsf{A} = 1$. Potom:

$$\mathrm{P}(\mathsf{T} = t|\mathsf{U} = u) = \mathrm{P}(\mathsf{E} = e, \mathsf{C} = c|\mathsf{U} = u, \mathsf{A} = 1) \tag{4}$$

Pro přehlednost v dalším odvození vynechejme konkrétní hodnoty náhodných proměnných:

$$\begin{aligned}
\mathrm{P}(\mathsf{E}, \mathsf{C}|\mathsf{U}, \mathsf{A}) &= \mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U}, \mathsf{A}) \cdot \mathrm{P}(\mathsf{E}|\mathsf{U}, \mathsf{A}) \\
&= \frac{\mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U}, \mathsf{A}) \cdot \mathrm{P}(\mathsf{A}|\mathsf{E}, \mathsf{U})}{\mathrm{P}(\mathsf{A}|\mathsf{E}, \mathsf{U})} \cdot \mathrm{P}(\mathsf{E}|\mathsf{U}, \mathsf{A}) \\
&= \frac{\mathrm{P}(\mathsf{A}, \mathsf{C}|\mathsf{E}, \mathsf{U})}{\sum_{\mathsf{C}} \mathrm{P}(\mathsf{A}, \mathsf{C}|\mathsf{E}, \mathsf{U})} \cdot \mathrm{P}(\mathsf{E}|\mathsf{U}, \mathsf{A}) \\
&= \frac{\mathrm{P}(\mathsf{A}|\mathsf{C}, \mathsf{E}, \mathsf{U}) \cdot \mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U})}{\sum_{\mathsf{C}} \mathrm{P}(\mathsf{A}|\mathsf{C}, \mathsf{E}, \mathsf{U}) \cdot \mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U})} \cdot \mathrm{P}(\mathsf{E}|\mathsf{U}, \mathsf{A}) \\
&\approx \frac{\mathrm{P}(\mathsf{A}|\mathsf{C}, \mathsf{E}) \cdot \mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U})}{\sum_{\mathsf{C}} \mathrm{P}(\mathsf{A}|\mathsf{C}, \mathsf{E}) \cdot \mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U})} \cdot \mathrm{P}(\mathsf{E}|\mathsf{U})
\end{aligned} \tag{5}$$

Je zřejmé, že úlohu nalezení modelu modelujícího pravděpodobnostní rozdělení $\mathrm{P}(\mathsf{T}|\mathsf{U})$ lze dekomponovat na úlohu nalezení následujících dílčích modelů:

- $\mathrm{P}(\mathsf{C}|\mathsf{E}, \mathsf{U}) = \mathrm{P}(\mathsf{C} = c|\mathsf{E} = e, \mathsf{U} = u)$ – *konceptový model* predikující pravděpodobnost abstraktního sémantického stromu c při pozorování vstupní promluvy u a odpovídající posloupnosti sémantických entit e. Konceptový model je tvořen *hierarchickým diskriminativním modelem* popsaným v kapitole 6. Tento model pro vstupní promluvu u reprezentovanou pomocí váženého konečného akceptoru (mřížky) predikuje pravděpodobnostní distribuci nad různými abstraktními sémantickými stromy c, tj. rozdělení $P(\mathsf{C} = c|\mathsf{U} = u)$.
- $\mathrm{P}(\mathsf{E}|\mathsf{U}) = \mathrm{P}(\mathsf{E} = e|\mathsf{U} = u)$ – *model detekce sémantických entit* přiřazující posloupnost sémantických entit e vstupní promluvě u. Je použit přístup, kdy sémantické entity jsou popsány pomocí expertně navržených bezkontextových gramatik. Algoritmus detekce sémantických entit (kapitola 7) pak na jejich základě nalezne v mřížce u všechny podřetězce, které patří do jazyka generovaného těmito gramatikami. Jelikož se tyto generované podřetězce mohou překrývat, je nutné nejprve nalézt množinu jednoznačně přiřazených sémantických entit (kapitola 7.1). Pro úlohu porozumění řeči je pak vhodné z této množiny opět sestavit mřížku sémantických entit E, která modeluje možné posloupnosti sémantických entit e a odpovídající pravděpodobnosti $\mathrm{P}(\mathsf{E} = e|\mathsf{U} = u)$ (kapitola 7.2).

- $\mathrm{P}(\mathsf{A}|\mathsf{E},\mathsf{C}) = \mathrm{P}(\mathsf{A}=1|\mathsf{E}=e,\mathsf{C}=c)$ – *model zarovnání* určující, zda posloupnost sémantických entit e lze zarovnat s abstraktním sémantickým stromem c. Model zarovnání byl založený na jednoduchém pravidle: sémantický strom definovaný pomocí e a c je validní ($\mathsf{A}=1$) právě tehdy, když všechny sémantické entity z e lze přiřadit odpovídajícím sémantickým konceptům v abstraktním sémantickém stromu c. Tato definice modelu zarovnání *nevyžaduje* zarovnání všech konceptů z c na sémantické entity e.

Výše zmíněná struktura diskriminativního modelu efektivně kombinuje statistický a znalostní (expertní) přístup k návrhu porozumění řeči. Statistický přístup je zastoupen hierarchickým diskriminativním modelem (kapitola 6). Znalostní přístup pak algoritmem pro detekci sémantických entit (kapitola 7).

6 Hierarchický diskriminativní model

Hierarchický diskriminativní model vznikl při experimentech s modelem STC. Již prvotní implementace modelu STC předčila referenční HVS parser v přesnosti predikce sémantických stromů. I přes velice slibné výsledky měl původní STC model některé nevýhody – především nemožnost získání více výstupních hypotéz s přiřazenými aposteriorními pravděpodobnostmi, dále pak generování výstupního sémantického stromu založené na heuristice a zarovnání vygenerovaného sémantického stromu s lexikální realizací promluvy založené na lexikálních třídách. Mezi další nevýhody STC modelu patří nemožnost trénování a predikce z neurčitého vstupu reprezentovaného mřížkou. Navíc výstupy jednotlivých dílčích klasifikátorů v modelu STC jsou korelovány a tato korelace není žádným způsobem uvažována při rekonstrukci sémantického stromu.

Pro účely popisu HDM si vypůjčíme terminologii z teorie umělých neuronových sítí, konkrétně z popisu dopředných perceptronových sítí. HDM se skládá ze tří vrstev (obrázek 5):

1. *Vstupní vrstva*, která efektivně vyčísluje hodnoty racionální jádrové funkce vzájemně mezi promluvami z trénovací množiny, případně mezi novou dosud neviděnou promluvou a všemi promluvami z trénovací množiny. Výstupem vstupní vrstvy je vektor hodnot racionálních jádrových funkcí použitý pro trénování a predikci pomocí skryté vrstvy.
2. *Skrytá vrstva* odpovídá modelu STC. Ten na základě vektoru hodnot jádrových funkcí generuje výstup skryté vrstvy. Ten je tvořený vektorem vzdáleností vstupní promluvy k oddělovacím nadrovinám binárních SVM klasifikátorů klasifikujících přítomnost jednotlivých sémantických n-tic.

3. *Výstupní vrstva* predikující na základě příznakového vektoru získaného z výstupu skryté vrstvy pravděpodobnosti jednotlivých sémantických pravidel. Z těchto sémantických pravidel je pak sestaven výstupní abstraktní sémantický strom.

6.1 Vstupní vrstva

Vstupní vrstva slouží k výpočtu hodnot jádrové funkce nad dvěma promluvami $u_j, u_k \in \mathcal{T}$ z trénovací množiny, případně k výpočtu hodnot jádrové funkce mezi neviděnou promluvou u a promluvami $u_k \in \mathcal{T}$. V případě modelu STC jsou vstupní promluvy reprezentovány pomocí příznakového vektoru obsahujícího lexikálně-syntaktické příznaky – nazvěme tento vektor $\mathbf{x}(u)$. Jádrová funkce v případě STC je lineární funkcí danou jako:

$$K(u_j, u_k) = \mathbf{x}(u_j) \cdot \mathbf{x}(u_k) \tag{6}$$

Autoři [33] popisují použití pouze lineární jádrové funkce, ale obecně může být použita libovolná jádrová funkce splňující Mercerovu podmínku [24].

Jeden z cílů disertační práce je navrhnout model porozumění, který umožňuje využít nejednoznačný výstup systému automatického rozpoznávání řeči ve formě slovní, nebo fonémové mřížky. Vyčíslení příznakových vektorů pro obecnou slovní nebo fonémovou mřížku může být výpočetně náročný proces. Lze však s výhodou použít teorii racionálních jádrových funkcí [10].

Racionální jádrové funkce jsou definovány nad dvojicí vážených konečných automatů. V této kapitole budeme předpokládat, že tyto automaty jsou slovní resp. fonémové řetězce, popř. mřížky. Jinými slovy budeme předpokládat, že se jedná o acyklické vážené konečné akceptory.

V tomto případě lze racionální jádrovou funkci definovat pomocí transduceru $T \circ T^{-1}$ a funkce ψ, kde T je vážený konečný automat nad polookruhem $\mathbb{K}$ definující tuto jádrovou funkci a funkce ψ provádí zobrazení hodnot z polookruhu $\mathbb{K}$ do prostoru reálných čísel. V disertační práci se omezíme pouze na n-gramové racionální jádrové funkce definované transducerem $T_{n,m}$:

$$T_i = (\mathcal{A} \times \{\epsilon\})^* \otimes (\bigoplus_{x \in \mathcal{A}} \{x\} \times \{x\})^i \otimes (\mathcal{A} \times \{\epsilon\})^* \tag{7}$$

$$T_{n,m} = \bigoplus_{i=n}^{m} T_i \tag{8}$$

Použití tohoto transduceru je ekvivalentní použití příznakového vektoru, který obsahuje střední četnosti n-gramů o délce n až m ve mřížce U spolu s lineární jádrovou funkcí.

Pokud jsou vstupní promluvy reprezentované pomocí řetězců nebo mřížek, je postupné vyčíslování jádrové funkce mezi neznámým vstupem a každým prvkem trénovací množiny neefektivní, neboť mnoho podřetězců vstupních mřížek se vyskytuje velmi často v různých mřížkách (častá slova, výplňová slova apod.). Výpočet racionální jádrové funkce lze často dekomponovat tak, že výpočet nad těmito shodnými částmi mřížek lze provádět pouze jednou a výsledek přiřadit všem mřížkám, které tuto část dat sdílejí. S výhodou lze použít optimalizační algoritmy definované nad váženými konečnými automaty, především determinizaci a minimalizaci. Postup prezentovaný v tomto odstavci je inspirován postupem při optimalizaci indexu v úloze spoken term detection (STD) [35] – optimální index je v tomto případě reprezentován pomocí minimálního deterministického váženého konečného automatu. Obdobně je postupováno při výpočtu racionální jádrové funkce, kdy celá trénovací množina $\mathcal{T}$ je reprezentována jako sjednocení mřížek z trénovací množiny, přičemž každé mřížce je přiřazen symbol umožňující identifikaci jednotlivých prvků. Tento automat pak již může být optimalizován jako běžný vážený konečný automat.

6.1.1 Efektivní výpočet racionální jádrové funkce

Pro výpočet jádrové funkce $K(U_k, U_j)$, kde U_k a U_j jsou acyklické vážené konečné akceptory odpovídající promluvám u_k, u_j, je použita kompozice $U_k \circ T \circ T^{-1} \circ U_j$. Algoritmus pro efektivní výpočet této kompozice je založen na vhodném pořadí kompozice a na optimalizaci mezivýsledků. Optimalizaci lze pro danou trénovací množinu $\mathcal{T}$ předpočítat a následně použít pro vyčíslení racionální jádrové funkce mezi neznámou promluvou U a všemi prvky trénovací množiny $U_j \in \mathcal{T}$.

Následuje popis algoritmu pro předzpracování trénovací množiny $\mathcal{T} = \{U_k\}_{k=1}^{l}$ do podoby minimálního deterministického váženého konečného transduceru R, který je následně použit pro rychlý výpočet racionální jádrové funkce:

1. Kompozice mřížky (akceptoru) U_k a transduceru T:

$$R_k = T^{-1} \circ U_k \tag{9}$$

   Tato kompozice je vyčíslena pro každý prvek trénovací množiny, neboť každý prvek této množiny může být potenciálním podpůrným vektorem při následném trénování skryté vrstvy, která je založena na SVM.
2. Projekce na R_k na vstupní symboly, tj. výpočet $\Pi_1(R_k)$.

3. Rozšíření abecedy symbolů $\Pi_1(R_k)$ o indexy promluv z trénovací množiny. Následně je provedena konkatenace akceptoru $\Pi_1(R_k) \otimes I(k)$, kde $I(k) = \{k\} \times \{k\}$. Jinými slovy dojde k přidání identifikátoru k-té promluvy na konec každé cesty akceptorem $\Pi_1(R_k)$. Tento krok je významný pro pozdější přiřazení hodnoty jádrové funkce konkrétnímu páru (U, U_k), bez přidání identifikátoru $I(k)$ by v následném kroku při determinizace automatu došlo ke sloučení cest se stejnými vstupními symboly, ale příslušející různým mřížkám.
4. Sjednocení všech těchto rozšířených akceptorů napříč všemi prvky U_k trénovací množiny $\mathcal{T}$, odstranění ϵ-přechodů, determinizace a minimalizace:

$$\bar{R} = \min \left[\det \left[\text{rmeps} \bigoplus_{k=1}^{l} \Pi_1(T^{-1} \circ U_k) \otimes I(k) \right] \right] \tag{10}$$

   Nyní je $\bar{R}$ minimální deterministický vážený konečný akceptor nad množinou $\mathcal{A} \cup \{k\}_{k=1}^{l}$.
5. Převedení akceptoru $\bar{R}$ na transducer R aplikací následujícího algoritmu:
   - Inicializace $R \leftarrow \bar{R}$
   - Pro všechny přechody e z akceptoru $\bar{R}$:
     - Je-li $i[e] \in \{k = 1, 2, \ldots, l\}$, pak $i[e] = \epsilon$.
     - Jinak $o[e] = \epsilon$.

Výsledný transducer R definuje relaci mezi řetězci akceptovanými transducerem T^{-1} a číselnými indexy k prvků trénovací množiny $\mathcal{T}$. Je nutné poznamenat, že tento transducer je minimální a deterministický vzhledem ke vstupním symbolům s výjimkou přechodů vedoucích do koncových stavů transduceru R, které po aplikaci výše uvedeného algoritmu mají vstupní symboly ϵ a výstupním symbolem je identifikátor prvku trénovací množiny k.

Transducery R a T definují parametry vstupní vrstvy. Zatímco T reprezentuje racionální jádrovou funkci, transducer R efektivním, minimálním a deterministickým způsobem uchovává prvky trénovací množiny. Výpočet jádrové funkce $K(U, U_k)\ \forall\ U_k \in \mathcal{T}$ mezi vstupní mřížkou U a všemi mřížkami trénovací množiny U_k je pak možné realizovat následujícím algoritmem:

1. Projekce kompozice $U \circ T$ na výstupní symboly, dále odstranění ϵ-přechodů a determinizace:

$$L = \det \left[\text{rmeps}\, \Pi_2(U \circ T) \right] \tag{11}$$

2. Výpočet kompozice $LR = L \circ R$.
3. Výpočet racionální jádrové funkce $K(U, U_k)$ procházením všech cest transducerem LR. Hodnota této funkce je rovna $\oplus$-sumě vah všech cest z počátečního

do koncového stavu transduceru LR, které mají výstupní symbol k. Pro zobrazení z prostoru polookruhu $\mathbb{K}$ do reálných čísel je použita funkce ψ:

$$K(U, U_k) = \psi \left(\bigoplus_{\substack{\pi \in \mathcal{P}(I_{LR}, \mathcal{F}_{LR}) \\ o[\pi]=k}} \lambda[p[\pi]] \otimes w[\pi] \otimes \rho[n[\pi]] \right) \tag{12}$$

Je důležité zdůraznit, že algoritmus procházející transducer vyčísluje celý vektor $K = [K(U, U_k)]_{k=1}^{l}$ naráz v jednom průchodu. Algoritmus počítá pouze ty prvky vektoru, pro které existuje řetězec symbolů přijímaný jak akceptorem $\Pi_2(U \circ T)$, tak akceptorem $\Pi_1(T^{-1} \circ U_k)$. Jinými slovy – jsou vyčíslovány pouze prvky $K(U, U_k)$, pro které je kompozice $U \circ T \circ T^{-1} \circ U_k$ neprázdný transducer.

6.2 Skrytá vrstva

Jak již bylo řečeno, HDM je rozšířením modelu STC, přičemž jako skrytá vrstva modelu HDM je použit model STC s několika modifikacemi. Tento model oproti původní implementaci zmiňované v [33] používá klasifikátory sémantických n-tic, přičemž délky těchto n-tic jsou voleny od jedné až po $n_{\max}$. Takto definované sémantické n-tice se překrývají a výstupy binárních klasifikátorů jsou silně korelované, např. pokud je v trénovacím stromu s_i přítomna n-tice (DEPARTURE, TO, STATION), pak jsou v tomto trénovacím stromu přítomny i n-tice (DEPARTURE, TO), (TO, STATION), (DEPARTURE), (TO) a (STATION) a každé z nich odpovídá právě jeden binární klasifikátor.

Protože již první úroveň sémantického stromu může obsahovat více sémantických konceptů – například anotace TIME, TO(STATION) – je do každého sémantického stromu navíc vložen nový kořenový uzel S. Z výše zmíněné anotace se pak stane S(TIME, TO(STATION)) a jsou zde obsaženy následující sémantické n-tice: (S), (TIME), (TO), (STATION), (S, TIME), (S, TO), (TO, STATION), (S, TO, STATION). Kořenový uzel je vkládán při generování množiny $\mathcal{S}$ (viz níže) a jeho vložení umožňuje rozlišit, zda daný sémantický koncept se v sémantickém stromu vyskytuje jako přímý následovník S nebo je zanořen hlouběji ve stromu.

Označme $\mathcal{S} = \{\mathbf{t} \in s_i : s_i \in \mathcal{T}\}$ množinu všech sémantických n-tic $\mathbf{t}$ získaných ze stromů s_i trénovací množiny $\mathcal{T}$. Experimenty s přesností klasifikace výskytu sémantických n-tic ukázaly, že přesnost klasifikace málo četných n-tic je nízká, proto pro trénování skryté vrstvy HDM modely jsou vybrány pouze n-tice četnější než definovaný práh N. Označme $\mathrm{cnt}(\mathcal{T}, \mathbf{t})$ počet výskytů n-tice $\mathbf{t}$ ve stromech z trénovací množiny $\mathcal{T}$. Pro další trénování je pak zvolena pouze podmnožina:

$$\mathcal{S}_N = \{\mathbf{t} \in \mathcal{S} : \mathrm{cnt}(\mathcal{T}, \mathbf{t}) \geq N\}' \tag{13}$$

Pro každý prvek $\mathbf{t} \in \mathcal{S}_N$ definujme množinu pozitivních a negativních trénovacích příkladů:

$$\begin{aligned}
\mathcal{T}_{\mathbf{t}}^{+} &= \{(U_k, +1) : \mathbf{t} \in s_k \wedge (U_k, s_k) \in \mathcal{T}\} \quad \mathbf{t} \in \mathcal{S}_N, k = 1, 2, \ldots, l \\
\mathcal{T}_{\mathbf{t}}^{-} &= \{(U_k, -1) : \mathbf{t} \notin s_k \wedge (U_k, s_k) \in \mathcal{T}\} \quad \mathbf{t} \in \mathcal{S}_N, k = 1, 2, \ldots, l \\
\mathcal{T}_{\mathbf{t}} &= \mathcal{T}_{\mathbf{t}}^{+} \cup \mathcal{T}_{\mathbf{t}}^{-}
\end{aligned} \tag{14}$$

Samotná skrytá vrstva HDM se skládá z binárních SVM klasifikátorů:

$$f_{\mathbf{t}}(U, \alpha(\mathbf{t})) : \mathcal{U} \to \{-1, +1\}, \quad \mathbf{t} \in \mathcal{S}_N, \quad U \in \mathcal{U} \tag{15}$$

získaných natrénováním nad trénovací množinou $\mathcal{T}_{\mathbf{t}} = \mathcal{T}_{\mathbf{t}}^{+} \cup \mathcal{T}_{\mathbf{t}}^{-}$. Množina $\mathcal{U}$ značí množinu všech možných acyklických automatů – reprezentuje prostor všech mřížek na výstupu systému automatického rozpoznávání řeči. Jako jádrové funkce při trénování a predikci je použita racionální jádrová funkce $K(U_i, U_j)$.

Poznamenejme, že pro predikci dosud neviděné promluvy U je nutné nejprve vyčíslit racionální jádrovou funkci $K(U, U_k)$, $\forall U_k \in \mathcal{T}$ a následně tyto hodnoty použít pro klasifikaci přítomnosti n-tice $\mathbf{t}$ ve výstupním stromu $\hat{s}$. Označme nyní $d_{\mathbf{t}}$ vzdálenost promluvy k rozhodovací nadrovině $H_{\mathbf{t}}$ klasifikátoru klasifikujícího přítomnost sémantické n-tice $\mathbf{t}$:

$$d_{\mathbf{t}} = \sum_{k=1}^{l} \alpha_k^{\mathbf{t}} y_k^{\mathbf{t}} K(U, U_k) + b^{\mathbf{t}} \tag{16}$$

Na základě hodnot $d_{\mathbf{t}}$ lze přímo rekonstruovat abstraktní sémantický strom pomocí dekódovacího algoritmu modelu STC. V disertační práci je však z hodnot $d_{\mathbf{t}}$ sestaven příznakový vektor, který je použit ve třetí (výstupní) vrstvě diskriminativních klasifikátorů.

6.3 Výstupní vrstva

Problém s nejednoznačností sémantického stromu při použití sémantických n-tic pro malá n je možné vyřešit použitím další vrstvy diskriminativních klasifikátorů. Na abstraktní sémantický strom je možné se dívat jako na derivační strom gramatiky. Abstraktní sémantické stromy jsou však, obdobně jako u STC modelu, uvažovány jako neuspořádané – neexistuje v nich relace uspořádání mezi následovníky jednoho uzlu. Derivační pravidla popisující generování takového stromu tedy musí tuto vlastnost brát v úvahu.

Proto každé z derivačních pravidel předpokládejme ve tvaru $A \to \beta\ (p)$, kde A je sémantický koncept, β je množina sémantických konceptů – následovníků sémantického konceptu A a $p = \mathrm{P}(A \to \beta|u)$ je pravděpodobnost realizace derivace při dané promluvě u.

Jako *sémantickou gramatiku promluvy* u nazývejme trojici $G_u = (\Theta, \mathcal{R}_u, S)$, kde:

- Θ je množina sémantických konceptů,
- $\mathcal{R}_u$ je množina derivačních pravidel ve tvaru $A \to \beta\ (p)$, kde $A \in \Theta$. Toto pravidlo říká, že ve stromu s pro vstupní promluvu u se vyskytuje uzel označený konceptem A s potomky β s pravděpodobností p:

$$p = \mathrm{P}(A \to \beta|u) = \mathrm{P}(\beta|A, u) \tag{17}$$

  Platí:

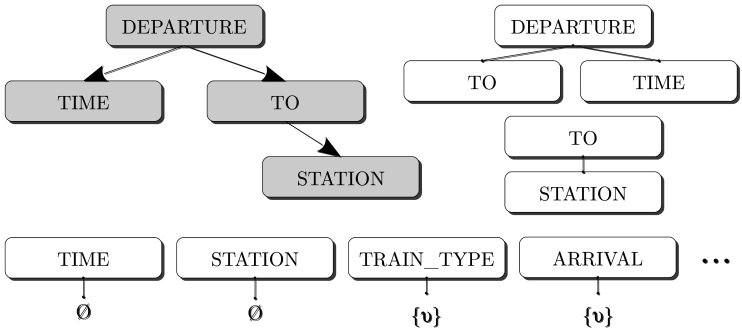
$$\sum_{\beta} \mathrm{P}(A \to \beta|u) = \sum_{\beta} \mathrm{P}(\beta|A, u) = 1 \tag{18}$$

- Množina β je podmnožinou množiny všech sémantických konceptů Θ doplněná o speciální symbol ν (viz níže).

$$\beta \subset \{\nu\} \cup \Theta \tag{19}$$

- $S \in \Theta$ je kořenový sémantický koncept, startovací symbol gramatiky G_u.

Tato definice sémantické gramatiky promluvy u je obdobná definici bezkontextových gramatik s následujícími rozdíly:



Obrázek 4: Sémantický strom z obrázku 3 dekomponovaný na pravidla sémantické gramatiky R. Pro všechny ostatní sémantické koncepty nevyskytující se v tomto stromu platí pravidlo $A \to \{\nu\}$, např. TRAIN_TYPE $\to \{\nu\}$.

- Symboly gramatiky nejsou děleny na terminální a neterminální, existuje pouze jediná množina sémantických konceptů Θ.
- Pravděpodobnosti přiřazené pravidlům sémantické gramatiky jsou parametrizovány promluvou u. Tímto se do sémantické gramatiky zavádí lexikalizace obdobně jako u lexikalizovaných bezkontextových gramatik [30].
- Pravidlo ve tvaru $A \to \emptyset$ (p) znamená, že s pravděpodobností p koncept A je v abstraktním sémantickém stromu pro promluvu u listovým uzlem (nemá potomky).
- Pravidlo ve tvaru $A \to \{\nu\}$ (p) znamená, že s pravděpodobností p se koncept A nenachází v abstraktním sémantickém stromu pro promluvu u.

Předpokládejme nyní, že pro danou promluvu u jsme získali sémantickou gramatiku $G_u = (\Theta, \mathcal{R}_u, S)$. Cílem dekódovacího algoritmu je z této gramatiky vygenerovat nejpravděpodobnější abstraktní sémantický strom.

Jako *částečný abstraktní sémantický strom* označme posloupnost pravidel $r = (r_1, r_2, \dots r_n)$, $r_i \subseteq \mathcal{R}_u$, kde r_1 je ve tvaru $S \to \beta$ a pro libovolný koncept A_i takový, že $r_i = A_i \to \beta_i$, existuje pravidlo $r_j = A_j \to \beta_j, 1 \leq j < i$ takové, že $A_i \in \beta_j$. Jinými slovy, koncept A_i je do derivačního stromu přidán pravidlem r_j a jeho potomci jsou určeni pravidlem r_i, přičemž musí platit $j < i$. Částečný sémantický strom odpovídá mezivýsledku při postupném generování derivačního stromu (popsáno k kapitole 6.3.1).

Množinu sémantický konceptů v částečném stromu r označme jako Θ_r:

$$\Theta_r = \bigcup_{A \to \beta \in r} \beta \tag{20}$$

Dále označme $L(A)$ jako počet prvků r_i takových, že $r_i = A \to \beta$ a $R(A)$ počet prvků r_i takových, že $r_i = C \to \beta,\ A \in \beta$. Jinými slovy $L(A)$ říká, kolikrát se ve stromu r objevil sémantický koncept A na levé straně nějakého pravidla. $R(A)$ pak vyčísluje, kolikrát se sémantický koncept A objevil na pravé straně nějakého pravidla.

Nyní můžeme označit množinu $\mathcal{E}_r$ expandovaných sémantických konceptů v částečném stromu r jako podmnožinu všech sémantických konceptů, pro které platí $L(A) = R(A)$:

$$\mathcal{E}_r = \{A : A \to \beta \in r \wedge L(A) = R(A)\} \tag{21}$$

Množinu neexpandovaných sémantických konceptů v částečném stromu r jako $\mathcal{M}_r$:

$$\mathcal{M}_r = \Theta_r \setminus \mathcal{E}_r \tag{22}$$

O částečném sémantickém stromu r budeme říkat, že je plně expandovaný, pokud $\mathcal{M}_r = \emptyset$.

Definujme množinu všech možných následovníků konceptu A jako $\mathcal{B}_A$:

$$\mathcal{B}_A = \{\beta : A \to \beta \in \mathcal{R}_u\} \tag{23}$$

Pak $\mathrm{P}(A \to \beta|u)$ je pravděpodobnostní rozdělení nad množinou $\mathcal{B}_A$ podmíněné sémantickým konceptem A a vstupní promluvou u.

Nakonec definujme pravděpodobnost částečného sémantického stromu $\mathrm{P}(r|u)$ jako součin pravděpodobností

$$\mathrm{P}(r|u) = \prod_{A \to \beta \in r} \mathrm{P}(A \to \beta|u) \tag{24}$$

6.3.1 Algoritmus určení abstraktního sémantického stromu

Pro určení nejpravděpodobnějšího abstraktního sémantického stromu ze sémantické gramatiky promluvy u je použit algoritmus prohledávání s nejmenší cenou. Cena $\mathrm{cost}(r)$ částečného sémantického stromu r je definována jako záporný logaritmus pravděpodobnosti $\mathrm{P}(r|u)$:

$$\begin{aligned} \mathrm{cost}(r) &= -\ln \mathrm{P}(r|u) = -\ln \prod_{A \to \beta \in r} \mathrm{P}(A \to \beta|u) \\ &= \sum_{A \to \beta \in r} -\ln \mathrm{P}(A \to \beta|u) \end{aligned} \tag{25}$$

Pro hledání je použita standardní podoba algoritmu hledání s nejmenší cenou.

Poznamenejme, že pravidla $A \to \emptyset$ a $A \to \{\nu\}$ slouží k definování vazby mezi sémantickým stromem a lexikální realizací promluvy u. Pomocí těchto pravidel je definována pravděpodobnost, že daný sémantický koncept se v promluvě vyskytuje, resp. nevyskytuje. Navíc pokud zvolíme množinu pravidel $\mathcal{R}_u$ tak, že neobsahuje pravidla ve tvaru $\nu \to \beta$, dosáhneme tím nemožnosti plně expandovat stromy obsahující pravidlo $A \to \{\nu\}$ a dekódovaný abstraktní sémantický strom tak nemůže obsahovat uzly ν.

6.3.2 Omezení na sémantické stromy generované HDM

Ze způsobu, jakým je definována sémantická gramatika pro promluvu u vyplývá množina omezení na třídu derivačních stromů generovaných HDM:

1. Abstraktní sémantické stromy jsou neuspořádané, tj. HDM, podobně jako STC, neumožňuje diskriminovat mezi dvěma abstraktními sémantickými anotacemi lišícími se pouze pořadím sémantických konceptů.
2. Více výskytů jednoho sémantického konceptu A v různých uzlech sémantického stromu sdílí stejné derivační pravidlo $A \to \beta$ a tudíž i stejnou pravděpodobnost derivace $\mathrm{P}(A \to \beta|u)$.
3. Sémantická gramatika neumožňuje diskriminovat mezi dvěma abstraktními sémantickými anotacemi lišícími se pouze počtem uzlů se stejným sémantickým konceptem a stejným rodičovským uzlem.
4. Sémantická gramatika neumožňuje generovat abstraktní sémantické stromy s uzly obsahujícími stejný sémantický koncept, ale lišícími se následovnickými uzly.

Výše uvedená omezení nejsou na překážku dobře pracujícímu modelu porozumění, neboť jejich důsledkům lze předcházet vhodným návrhem anotačního schématu. Mnohá tato omezení jsou společná s modelem STC a vyplývají především ze způsobu reprezentace vstupní promluvy ve formě příznakového vektoru obsahujícího četnost jednotlivých n-gramů (více v [33]).

6.3.3 Určení množiny pravidel $\mathcal{R}_u$

Sémantická gramatika využívá množinu pravidel $\mathcal{R}_u$ zahrnující pro každé pravidlo pravděpodobnostní rozdělení $\mathrm{P}(A \to \beta|u)$ podmíněné promluvou u. V modelu HDM je nejprve určena množina $\mathcal{R}$ obsahující všechny možné expanze sémantických konceptů vyskytující se v trénovací množině $\mathcal{T}$ a následně je použita sada diskriminativních klasifikátorů pro přiřazení aposteriorní pravděpodobnosti $\mathrm{P}(A \to \beta|u)$ těmto pravidlům a tedy k získání parametrizované množiny pravidel $\mathcal{R}_u$.

Podmíněnou pravděpodobnost $\mathrm{P}(A \to \beta|u) = \mathrm{P}(\beta|A, u), \beta \in \mathcal{B}_A$ je možné odhadnout pomocí metod strojového učení. Tato pravděpodobnost je podmíněna sémantickým konceptem A a vstupní promluvou u. Vyplývá z toho tedy potřeba natrénovat pro každý koncept $A \in \Theta$ jeden klasifikátor diskriminující mezi prvky $\mathcal{B}_A$ a mající na vstupu příznakový vektor odvozený od promluvy u, potažmo od mřížky U.

Klasifikátor pro koncept A na základě příznakového vektoru získaného z mřížky U provádí klasifikaci do jedné ze tříd $\beta \in \mathcal{B}_A$. Musí ale zároveň poskytovat odhad aposteriorní pravděpodobnosti pro všechny cílové třídy, což je právě hledaná podmíněná pravděpodobnost $\mathrm{P}(A \to \beta|u)$.

Možných klasifikačních metod, které podporují klasifikaci do více tříd s odhadem aposteriorní pravděpodobnosti jednotlivých tříd je celá řada. V disertační práci byly

použity SVM implementující klasifikaci do více cílových tříd a odhad aposteriorních pravděpodobností.

Klasifikátor poskytující odhad aposteriorní pravděpodobnosti jednotlivých derivací uzlu A má na vstupu příznakový vektor. Experimenty prokázaly vhodnost použít hierarchickou architekturu, kde skrytá vrstva je realizována jako STC, nicméně neprovádí přímo klasifikaci jednotlivých sémantických n-tic, ale pouze určuje vzdálenost k jednotlivým rozhodovacím nadrovinám $H^{\mathbf{t}}$. Pro promluvu u reprezentovanou mřížkou U je pak příznakový vektor složen z prvků $d_{\mathbf{t}}(U)\ \forall \mathbf{t} \in \mathcal{S}_N$. Poznamenejme, že příznakový vektor $\mathbf{d}(U)$ může být rozšířen o další příznaky získané ze vstupní promluvy u, popřípadě z mřížky U.

Označme klasifikátor klasifikující příznakový vektor $\mathbf{d}(U)$ do jedné ze tříd z množiny $\mathcal{B}_A$ jako

$$g_A(\mathbf{d}(U), \alpha(A)) : \mathbb{R}^{|\mathcal{S}_N|} \rightarrow \mathcal{B}_A, \quad A \in \Theta \tag{26}$$

kde $\alpha(A)$ je vektor parametrů klasifikátoru trénovaného pro koncept A.

Jako $cls(s_k, A)$ označme funkci, která pro sémantický strom s_k a sémantický koncept A vrátí množinu sémantických konceptů $\beta \in \mathcal{B}_A$ – následovníků sémantického konceptu A ve stromu s_k. Je-li sémantický koncept A obsažen v listovém uzlu, je vrácena prázdná množina, není-li A obsažen v s_k, pak vrátí $\{\nu\}$:

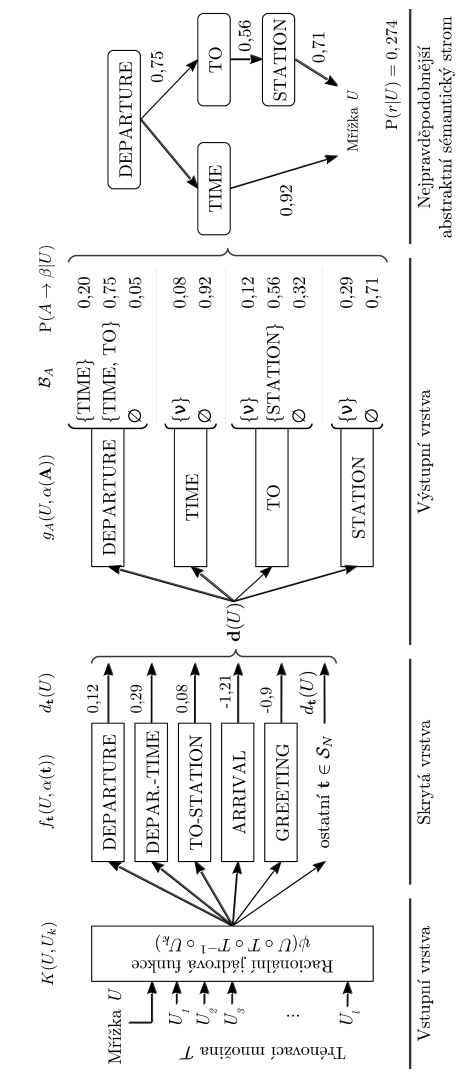
$$\mathrm{cls}(s_k, A) = \begin{cases} \beta & \text{pokud } A \rightarrow \beta \in s_k \\ \emptyset & \text{pokud } A \text{ je list } s_k \\ \{\nu\} & \text{jinak} \end{cases} \tag{27}$$

Obdobně jako u binárních klasifikátorů sémantických n-tic $f_{\mathbf{t}}$ je vhodné omezit množinu možných cílových tříd pro každý sémantický koncept A. Označíme-li $\mathrm{cnt}(\mathcal{T}, A \rightarrow \beta)$ počet výskytů uzlu se sémantickým konceptem A a následníky β napříč trénovací množinou $\mathcal{T}$, pak můžeme definovat modifikovanou funkci $\mathrm{cls}_M(s_k, A)$ následujícím způsobem:

$$\mathrm{cls}_M(s_k, A) = \begin{cases} \beta & \text{pokud } A \rightarrow \beta \in s_k \ \wedge\ \mathrm{cnt}(\mathcal{T}, A \rightarrow \beta) \geq M \\ \emptyset & \text{pokud } A \text{ je list } s_k \\ \{\nu\} & \text{jinak} \end{cases} \tag{28}$$

kde M je volitelný práh. Každý klasifikátor $g_A(\mathbf{d}(U), \alpha(A))$ je pak trénován z trénovací množiny $\mathcal{T}_A$, která je vygenerována z množiny sémanticky anotovaných promluv $\mathcal{T} = (U_k, s_k)_{k=1}^{l}$ jako:

$$\mathcal{T}_A = \{(\mathbf{d}(U_k), \mathrm{cls}_M(s_k, A)) : (U_k, s_k) \in \mathcal{T},\ k = 1, 2, \ldots, l\} \tag{29}$$



Obrázek 5: Schéma hierarchického diskriminativního modelu.

7 Detekce sémantických entit

Nyní se zaměřme na popis algoritmu vyvinutého pro hierarchický diskriminativní model, který umožňuje získání pravděpodobnostního rozdělení $\mathrm{P}(\mathsf{E} = e|U = u)$ z promluvy u, popř. z mřížky U. Poznamenejme, že popisovaný algoritmus předpokládá použití slovní mřížky, popř. první nejlepší slovní hypotézy.

Sémantickou entitou myslíme konkrétní objekt zmíněný v dané promluvě a významný z pohledu sémantické analýzy. Sémantické entity mohou být různých *typů*, například časové údaje, datum, položky z rozsáhlých databází (seznamy stanic, osob). Sémantické entity mohou mít svoji vnitřní strukturu – *sémantickou interpretaci*. Pro příklad jmenujme sémantické entity typu čas, které se mohou skládat z údaje o hodinách a minutách.

V disertační práci je pro modelování sémantických entit použit *znalostní přístup*. Důvodem je především snaha pomocí expertních znalostí posílit statisticky založený model porozumění a zvýšit tak jeho robustnost a přesnost. Je důležité zdůraznit, že expertní znalosti mohou být často automaticky generovány z vhodné databáze (seznam stanic, seznam osob) nebo dokonce mohou být relativně jednoduše přenositelné mezi různými doménami (gramatiku sémantických entit typu čas lze použít bez úprav ve více různých úlohách).

Předpokládejme, že sémantické entity daného typu mají svoji vnitřní strukturu popsatelnou bezkontextovou gramatikou. Přestože lze uvažovat i stochastické bezkontextové gramatiky, je v praxi velmi obtížné expertním způsobem určit pravděpodobnosti expanzí jednotlivých pravidel. Bezkontextové gramatiky jsou intuitivní, standardizovaný [36] způsob zápisu znalostí návrháře hlasového dialogového systému, navíc jejich užití umožňuje znovupoužít existující bázi znalostí z existujících hlasových dialogových systémů, kde rozpoznávání nebo porozumění řeči je založeno na těchto gramatikách.

Ve vstupní promluvě je nutné označit (a tudíž gramatikami modelovat) pouze takové podposloupnosti terminálních symbolů, které odpovídají nějaké gramatice. Tím se řádově zjednodušuje úloha návrhu gramatiky reprezentující sémantické entity daného typu, neboť není zpravidla nutné uvažovat různá výplňová slova, která nenáleží žádné sémantické entitě. Při použití jediné globální stochastické bezkontextové gramatiky by bylo nutné výplňová slova modelovat a zahrnout je mezi terminální symboly gramatiky.

Předpokládejme, že bezkontextová gramatika není rekurzivní, tj. pomocí pravidel gramatiky nelze z neterminálního symbolu A odvodit derivační podstrom obsahující A v jiném uzlu než v kořeni. Potom lze tuto bezkontextovou gramatiku převést na regulární [37]. V případě, že bezkontextová gramatika je rekurzivní, je možné

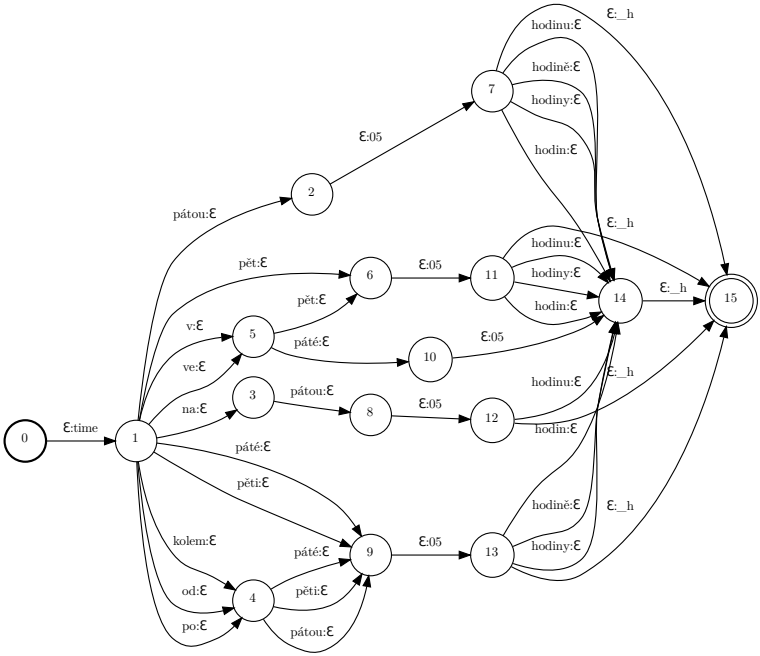
v průběhu kompilace hlídat hloubku rekurze a od určité hloubky znoření aplikovat omezení, např. dále neexpandovat neterminál způsobující rekurzi a nahradit jej prázdným symbolem. Výsledný vážený konečný automat však poté bude přijímat pouze aproximaci původní bezkontextové gramatiky [37, 38]. V oblasti zpracování mluvené řeči však tato omezení nejsou limitující, neboť možné promluvy uživatele hlasového dialogového systému jsou svojí délkou v čase a tudíž i v počtu slov omezené a aproximace bezkontextové gramatiky konečným automatem je pro tyto omezené posloupnosti slov dostačující. Problematice kompilace bezkontextových gramatik na vážené konečné automaty se věnují například publikace [39, 40].

Výstupem kompilace bezkontextové gramatiky odpovídající sémantickým entitám určitého typu je konečný transducer převádějící posloupnost slov na posloupnost značek reprezentujících sémantickou entitu, tj. typ a interpretaci. Budeme předpokládat, že první symbol posloupnosti značek identifikuje typ sémantické entity, zbylé symboly jsou závislé na typu sémantické entity a reprezentují interpretaci. Pro přehlednost zápisu budou v příkladech jednotlivé symboly sémantických entit odděleny znakem dvojtečky. Příklad takové gramatiky je zobrazen na obrázku 6. Sémantické gramatiky mohou záměrně obsahovat i množství negramatických cest, např. čas *ve pět hodin* ve výše uvedeném příkladě. Tímto přístupem lze účinně podchytit např. chybně rozpoznaná slova na výstupu systému automatického rozpoznávání řeči a zvýšit tak robustnost detekce sémantických entit.

Nyní předpokládejme, že každé sémantické entitě z odpovídá gramatika G_z a z ní kompilací získaný konečný transducer T_z. Gramatika G_z je konstruována tak, aby transducer T_z při akceptaci posloupnosti vstupních symbolů jako první symbol vrátil identifikátor z. Pak lze všechny sémantické entity reprezentovat pomocí transduceru Z získaného jako sjednocení dílčích T_z:

$$Z = \bigoplus_z T_z \tag{30}$$

Transducer Z nemusí nutně být funkcionální, tj. pro jeden akceptovaný vstupní řetězec může vrátit obecně více výstupních řetězců. Proto tento transducer nelze přímo optimalizovat pomocí algoritmů determinizace a minimalizace. Nicméně lze použít postup popsaný např. v [35] – zde jsou přeznačeny vstupní a výstupní symboly přechodů tak, že nově nesou jediný symbol vzniklý zakódováním původní dvojice vstupní-výstupní symbol. Takto vznikne (vážený) konečný akceptor, na který již lze aplikovat algoritmus determinizace a minimalizace. Po optimalizaci zakódovaného automatu je nutné převést zakódované symboly na přechodech zpět do původní abecedy vstupních a výstupních symbolů. Tímto krokem se může porušit vlastnost determinismu a minimality výsledného automatu, nicméně míra nedeterminismu vyjádřená počtem přechodů z daného stavu označených stejným symbolem je menší nebo rovná původnímu automatu před optimalizací.



Obrázek 6: Konečný automat získaný kompilací gramatiky sémantické entity typu *čas*. Pro názornost byla vybrána pouze ta část automatu, která odpovídá výstupním symbolům *time:05:_h* (pět hodin).

Nyní uvažujme vstupní promluvu u a odpovídající slovní mřížku U. Cílem je nalézt pravděpodobnostní rozložení $\mathrm{P}(\mathsf{E} = e|\mathsf{U} = U)$ posloupností sémantických entit $e = (e_1, e_2, \ldots, e_n)$, přičemž každé posloupnosti e odpovídá posloupnost typů sémantických entit $(z_1, z_2, \ldots, z_n)$ a pro každou sémantickou entitu e_i platí, že kompozice $T_{z_i} \circ e_i = T$ je neprázdný automat. Jinými slovy, pro každou sémantickou entitu e_i existuje cesta automatem T_{z_i} taková, že posloupnost výstupních symbolů odpovídá e_i. Automat U se předpokládá v takové podobě, kdy jeho váhové ohodnocení tvoří pravděpodobnostní distribuci nad množinou cest přijímaných tímto automatem.

Pro řešení úlohy detekce sémantických entit použijeme přístup faktorového transduceru, který umožňuje efektivně reprezentovat všechny možné faktory vstupní mřížky U a jejich aposteriorní pravděpodobnosti. Navíc díky tomuto přístupu není nutné modelovat výplňová slova mezi jednotlivými sémantickými entitami, neboť ty faktory, která tato výplňová slova obsahují neodpovídají žádné cestě transducerem Z

a tudíž neovlivní ostatní detekované sémantické entity. Nicméně, jednotlivé faktory různých délek v rámci faktorového automatu $F(U)$ se překrývají a tudíž jsou generovány nadbytečné výskyty sémantických entit. Proto je pro nalezení množiny jednoznačně přiřazených sémantických entit použito celočíselného programování (kapitola 7.1) a následně je popsán algoritmus, který umožňuje z množiny jednoznačně přiřazených sémantických entit rekonstruovat mřížku sémantických entit (kapitola 7.2).

7.1 Nalezení jednoznačně přiřazených sémantických entit

Před samotným popisem procesu hledání sémantických entit pomocí vážených konečných transducerů uveďme výčet případů, které mohou pro danou cestu mřížkou U nastat:

1. Jedné cestě v mřížce U odpovídá prázdná posloupnost sémantických entit. Váhu této cesty je však nutné promítnout do pravděpodobnosti $\mathrm{P}(\mathsf{E} = \{\}|\mathsf{U} = U)$.
2. Jedné cestě v mřížce U odpovídá právě jedna posloupnost sémantických entit.
3. Jedné cestě v mřížce U může odpovídat více různých posloupností sémantických entit.
4. Více různých cest v mřížce U může mít přiřazeno stejnou posloupnost sémantických entit.

Proto byla navržena následující heuristika *jednoznačného maximálního přiřazení*:

- Každé slovo dané cesty π_U mřížkou U může náležet pouze jedné sémantické entitě.
- A zároveň, z možných posloupností sémantických entit pro danou cestu π_U je vybrána taková, která maximalizuje počet slov dané cesty, která jsou součástí některé sémantické entity.

Uveďme nyní algoritmus hledání pravděpodobnostního rozdělení $\mathrm{P}(\mathsf{E} = e|\mathsf{U} = u)$ posloupností sémantických entit pro danou mřížku:

1. Mřížka (vážený akceptor) U je převedena na vážený transducer U_T tak, že vstupní symboly přechodů jsou nahrazeny v rámci mřížky jednoznačnými identifikátory, výstupní symboly jsou zachovány.
2. Vážený transducer U_T je převeden na vážený faktorový automat $F(U_T)$.
3. Kompozicí $F(U_T) \circ Z$ jsou vybrány ty faktory, které odpovídají nějaké sémantické entitě ze Z.
4. Je aplikována heuristika *jednoznačného přiřazení*. Pro její aplikaci jsou použity jednoznačné identifikátory přechodů mezi stavy zavedené v bodu 1. Výsledkem je množina nepřekrývajících se sémantických entit P_Z^*.

5. Z množiny sémantických entit je rekonstruován vážený konečný akceptor přijímající všechny posloupnosti sémantických entit e odpovídajících mřížce U.
6. Nad výsledným akceptorem je provedeno odstranění ϵ-přechodů, determinizace a minimalizace. Akceptor E získaný optimalizací odpovídá mřížce sémantických entit a jeho ohodnocení vahami odpovídá pravděpodobnostní distribuci $\mathrm{P}(\mathsf{E} = e|\mathsf{U} = U)$.

Po provedení kompozice $F_Z = F(U_T) \circ Z$ výsledný transducer reprezentuje množinu faktorů cest transducerem U_T označených sémantickými entitami. Vstupní symboly libovolné cesty $\pi(F_Z)$ reprezentují část cesty (faktor) v původní mřížce U_T, výstupní symboly pak již samotné sémantické entity definované transducerem Z.

Pro každý faktor $\pi^i(F_Z) = (u^i, y^i)$ sestavme pětici:

$$(u^i, y^i, p[u^i], n[u^i], \mathrm{P}(u^i \in U))$$

kde:

- u^i je faktor $\Pi_1(U_T)$ – posloupnost unikátních identifikátorů přechodů o délce k^i symbolů,
- y^i je posloupnost značek, jako celek jednoznačně definuje typ sémantické entity a její hodnotu,
- $p[u^i]$ je počáteční stav přechodu s identifikátorem u^i_1 v automatu U_T,
- $n[u^i]$ je koncový stav přechodu s identifikátorem $u^i_{k^i}$ v automatu U_T,
- $\mathrm{P}(u^i \in U_T) = \exp(-w_{F_Z}[u^i, y^i])$ je aposteriorní pravděpodobnost výskytu faktoru (u^i, y^i) v mřížce U.

Předpokládejme, že těchto cest (faktorů) je celkem n a definujme jejich libovolné uspořádání s indexem $i = 1, 2, \ldots, n$ do posloupnosti P_Z:

$$P_Z = \{(u^i, y^i, p[u^i], n[u^i], \mathrm{P}(u^i \in U_T))\}_{i=1}^{n} \tag{31}$$

Z posloupnosti P_Z je nyní nutné vybrat takovou podposloupnost P_Z^*, která splňuje požadavky heuristiky jednoznačného maximálního přiřazení. Pro rozhodnutí o tom, zda konkrétní dvojice (u^i, y^i) odpovídá heuristice, formulujme optimalizační úlohu binárního celočíselného programování. Její omezení zajišťují splnění prvního bodu heuristiky (pro každou cestu mřížkou je libovolné slovo přiřazeno nejvýše jedné sémantické entitě), optimalizační kriterium pak splnění druhého bodu (z možných řešení je vybráno takové, které pokrývá maximální počet slov dané cesty). Úlohu formulujme následovně:

$$\begin{aligned} \mathbf{G} \cdot \mathbf{x} &\leq \mathbf{h} \\ \mathbf{c}^\intercal \cdot \mathbf{x} &\to \max \end{aligned} \tag{32}$$

kde optimalizace probíhá vzhledem k prvkům n-rozměrného vektoru $\mathbf{x} = [x_i]_{i=1}^n$, kde prvky $x_i \in \{0, 1\}$. Pro optimální řešení platí, že je-li $x_i = 1$, pak (u^i, y^i) splňuje heuristiku jednoznačného maximálního pokrytí a y^i patří do některé posloupnosti sémantických entit e přiřazených mřížce U.

Pro definici matice $\mathbf{G}$ nejprve zkonkretizujme požadavek prvního bodu heuristiky jednoznačného maximálního pokrytí. Poněvadž nad mřížkou U_T může obecně existovat velké množství cest, je nutné pro každý pár (u^i, u^j), $i \neq j$ stanovit, zda v automatu U_T neexistuje cesta π_{U_T} taková, že faktory u^i a u^j se překrývají. Budeme říkat, že faktory u^i a u^j $(i \neq j)$ *se překrývají*, pokud je splněna alespoň jedna z následujících podmínek:

- Existuje neprázdná posloupnost u' a posloupnosti a, b takové, že $u^i = au'$ a zároveň $u^j = u'b$.
- Existuje neprázdná posloupnost u' a posloupnosti a, b takové, že $u^i = u'a$ a zároveň $u^j = bu'$.
- Existují posloupnosti a, b takové, že $u^i = au^jb$.
- Existují posloupnosti a, b takové, že $u^j = au^ib$.

Podotkněme, že tato definice je symetrická, pokud se u^i překrývá s u^j, pak se i u^j překrývá s u^i. Pokud se faktory u^i a u^j překrývají, pak pro splnění heuristiky jednoznačného maximálního pokrytí je nutné, aby ve výsledné posloupnosti P_Z^* byl nejvýše jeden z nich a tedy:

$$x_i + x_j \leq 1 \tag{33}$$

Předpokládejme, že v posloupnosti P_Z existuje m párů (u^i, u^j), které se překrývají. Platí $0 \leq m \leq \frac{n^2-n}{2}$. Pro případ, kdy neexistují žádné překrývající se faktory $(m = 0)$, definujme $x_i = 1,\ i = 1, 2, \ldots, n$.

Pro případy $m > 0$ sestavíme matici $\mathbf{G} = [g_{kl}]$ o $m \times n$ prvcích, pro kterou platí, že pokud se u^i a u^j překrývají, pak existuje řádek matice k takový, že $g_{ki} = g_{kj} = 1$ a $g_{kl} = 0$ pro $l \neq i, j$. Vektor $\mathbf{h}$ je m-rozměrný sloupcový vektor samých jedniček.

Kriteriální funkce má za cíl vybrat ze všech možných řešení takové, které maximalizuje počet slov v mřížce s přiřazenou sémantickou entitou. Kritérium je dáno n-rozměrným sloupcovým vektorem $\mathbf{c}$ s prvky c_i, pro jejichž výpočet byl použit následující vzorec:

$$c_i = (k^i)^2 \cdot \mathrm{P}(u^i \in U) \tag{34}$$

Tento tvar kriteria má za cíl preferovat ta řešení, která zahrnují delší faktory u^i (k^i je délka faktoru u^i). Vážení aposteriorní pravděpodobností pak zajistí, že jsou do posloupnosti P_Z^* prioritně vybírány ty faktory, které mají vyšší aposteriorní pravděpodobnost. Druhá mocnina délky faktoru pak má za cíl při optimalizaci prioritizovat

ta řešení, která mají menší počet faktorů. Zabrání se tím rozdělení sémantických entit na jednotlivé části – například oddělení hodin a minut u časového údaje do samostatných sémantických entit.

Po aplikaci algoritmu binárního celočíselného programování a získání optimálního vektoru $\mathbf{x}$ již lze provést omezení posloupnosti P_Z na podposloupnost P_Z^* splňující heuristiku jednoznačného maximálního pokrytí:

$$P_Z^* = \{(u^i, y^i, p[u^i], n[u^i], \mathrm{P}(u^i \in U) \in P_Z) : x_i = 1\} \tag{35}$$

7.2 Sestavení mřížky sémantických entit

Cílem detekce sémantických entit však není nalezení množiny faktorů splňujících výše uvedenou heuristiku, nýbrž získání pravděpodobnostního rozdělení $\mathrm{P}(\mathsf{E} = e|\mathsf{U} = U)$, přičemž e je posloupnost sémantických entit tvořená prvky y^i z P_Z^*.

V rámci disertační práce byl vyvinut exaktní algoritmus, který pro dané faktory z P_Z^* sestaví minimální deterministický acyklický vážený konečný akceptor E, jehož cesty odpovídají různým posloupnostem e a váhy pak pravděpodobnosti $\mathrm{P}(\mathsf{E} = e)$. Pokud byly faktory P_Z^* generovány z mřížky U s použitím transduceru Z, pak E reprezentuje mřížku sémantických entit a přímo modeluje podmíněnou pravděpodobnost $\mathrm{P}(\mathsf{E} = e|\mathsf{U} = U)$.

8 Definice úlohy

První z úloh, na niž byly prezentované metody ověřovány, byla úloha *Nádraží* řešená na Katedře kybernetiky Západočeské univerzity v Plzni jako modelová úloha pro vývoj hlasových dialogových systémů nové generace v rámci projektu Centra aplikované kybernetiky (CAK). Předmětem této úlohy je vývoj hlasového dialogového systému pro podávání informací o odjezdech a příjezdech vlaků. Pro vývoj tohoto hlasového dialogového systému byl sestaven korpus *HHTT* (Human-Human Train Timetable) [41].

Korpus HHTT obsahuje záznamy dialogů probíhajících v rámci provozu informačního centra o odjezdech a příjezdech vlaků. Tyto dialogy probíhaly vždy mezi dvěma lidmi. Tato skutečnost je významná především z pohledu variability a obsahu různých promluv vyskytujících se v dialozích – v rámci dialogů člověk-člověk dochází mnohem méně často k nedorozuměním v porovnání s dialogy člověk-stroj. Navíc lidé při komunikaci používají i různé neřečové komunikační prostředky, do proudu zvuku vkládají neřečové události, např. ehm-hmm, ehm-mm apod.

Data byla sbírána v době od dubna 2000 do srpna 2000. Volající byli především Češi mluvící spontánní češtinou. Audio signál byl získán z analogové telefonní linky vzorkovaný frekvencí 8kHz a komprimovaný A-Law kompresí, přičemž oba kanály (operátor a uživatel) byly smíchány do jediného monofonního kanálu. Každá promluva byla rozdělena na segmenty, přičemž každému segmentu byl následně přiřazen právě jeden dialogový akt [41].

Pro trénování subsystému porozumění je využita dimenze SEMANTICS tohoto korpusu, která charakterizuje význam dané promluvy. Navíc byly pro trénování použity jak promluvy operátora, tak uživatele telefonní linky. Toto sloučení obou stran dialogu je odůvodnitelné, neboť dialogy probíhají v rámci totožné domény (vlaková spojení) a operátor i uživatel sdílí jak slovník, tak i množinu sémantických konceptů. Sloučením dojde ke zvýšení počtu trénovacích vět, což vede obecně k robustnějšímu modelu porozumění.

V tabulce 3 jsou uvedeny vlastnosti korpusu HHTT po odstranění promluv, které neodpovídají slovní transkripci, a také promluv, které obsahují více jak jeden dialogový akt.

Druhou úlohou, nad kterou byly navržené modely evaluovány, je úloha telefonní inteligentní assistentky. Korpus telefonní inteligentní asistentky vznikl v rámci výzkumného projektu MPO TIP FR-TI1/518 Inteligentní telefonní asistentka. Tento projekt řešený firmou SpeechTech s.r.o. a Katedrou kybernetiky Západočeské univerzity v Plzni si klade za cíl výzkum a vývoj hlasového dialogového systému Telefonní inteligentní asistentka (TIA). Tento systém byl měl poskytovat skupinám v rámci malých a středních podniků unifikované hlasové rozhraní k nástrojům pro organizaci času – především k osobním a sdíleným kalendářům, k plánování sdílených prostředků jako jsou automobily, projektory, zasedací místnosti apod. Další z funkcionalit by měla umožňovat napojení na telefonní seznam organizace a spojování přímých i konferenčních hovorů.

Pro účely vývoje tohoto hlasového dialogového systému a výzkumu metod pro porozumění řeči byl nahrán a anotován řečový korpus. Tento korpus obsahuje dvě části. První z nich byla nahrávána pomocí hlasového dialogového systému, který simuloval chování budoucího systému pomocí posloupnosti jednotlivých poddialogů. Tato část je složena ze 187 dialogů a 2469 vět. Druhá část korpusu byla zaměřena na cílený sběr promluv obsahujících vybrané sémantické entity, dotazovaní odpovídali na předpřipravené otázky typu *Kdy jste se narodil?* (datum), *Kdy je neděle vzhledem k dnešku?* (relativní datum), *Pršelo včera?* (souhlas/nesouhlas). Ukázkový dialog z korpusu TIA je zachycen v tabulce 4.

č.	*ml.*	*promluva*	*abstraktní sémantický strom*
1	O	informace prosím	GREETING
2	U	dobrý den	GREETING
		já bych potřebovala zítra ráno kolem osmé nebo sedmé nějaký vlak do prahy	DEPARTURE(TIME, TO(STATION))
3	O	takže buď vám jede šest třicet šest rychlík	DEPARTURE(TIME, TRAIN_TYPE)
		ten je v praze osm deset	ARRIVAL(TO(STATION), TIME)
		a nebo rychlík osm nula dva	DEPARTURE(TRAIN_TYPE, TIME)
		a praha devět čtyřicet šest	ARRIVAL(TO(STATION), TIME)
4	U	devět čtyřicet šest	TIME
5	O	ano	ACCEPT
6	U	a všechno jsou to rychlíky	TRAIN_TYPE
7	O	oba dva ano	ACCEPT
8	U	děkuji	—
		nashledanou	—
9	O	není zač	—
		nashledanou	—

Tabulka 2: Ukázka anotovaného dialogu (dimenze SEMANTICS) z korpusu HHTT. Sloupec *ml.* určuje mluvčího (O – operátor, U – uživatel).

	train	*devel*	*test*
Počet vět	5240	570	1439
Celková délka h:m:s	2:40:25	0:17:22	0:44:59
ø doba 1 věty ($\pm\sigma$)	1,84±1,44	1,83±1,25	1,88±1,31
Počet tokenů	21517	2301	5838
ø počet tokenů 1 věty ($\pm\sigma$)	4,11±3,47	4,04±3,21	4,06±3,22
Velikost slovníku	1656	476	731
Četnost OOV	–	4,00%	7,45%
Počet konceptů v sémantických stromech	8967	997	2584
Počet unikátních konceptů	32	28	28
ø počet konceptů 1 věty ($\pm\sigma$)	1,71±1,24	1,75±1,24	1,80±1,30
Počet vět s 1 konceptem	3439	360	896
ø počet konceptů 1 věty ($\pm\sigma$), má-li věta více než 1 koncept	3,07±1,29	3,03±1,26	3,11±1,32

Tabulka 3: Vlastnosti korpusu HHTT.

promluva	*abstraktní sémantický strom*
potřebovala bych rezervovat zasedací místnost na čtvrtek ve čtyři hodiny	VYTVOR(REZERVACE(VEC, DATUM, T))
je tento termín volný	ZJISTI(KALENDAR)
potom potřebuju zasedací místnost na úterý to je zítra o+	VYTVOR(REZERVACE(VEC, DATUM))
na zítra v deset hodin zasedací místnost číslo tři	VYTVOR(REZERVACE(DATUM, T, VEC))
ne to stačí	NE
všechny problémy mám vyřešené	OOT
již nemám další dotaz ukončím tuto anketu	NE
jaké schůzky mám sjednaný na p+ sjednané na příští týden	ZJISTI(KALENDAR(RELATIVNI))
lze zjistit jaké schůzky mám zjiš+ zji+ sjednané na příští týden	ZJISTI(KALENDAR(RELATIVNI))
ne již nemám žádné přání děkuji	NE, DIKY
ne všechny problémy mám vyřešené	NE

Tabulka 4: Ukázkové promluvy z korpusu TIA, slova ukončená symbolem + jsou nedořeky.

	train	*devel*	*test*
Počet vět	4166	452	1054
Celková délka h:m:s	6:45:57	0:40:30	1:39:38
ø doba 1 věty ($\pm\sigma$)	3,80±1,34	3,81±1,30	3,83±1,29
Počet tokenů	33562	3501	8387
ø počet tokenů 1 věty ($\pm\sigma$)	7,74±7,63	7,46±7,28	7,82±7,34
Velikost slovníku	2655	703	1181
Četnost OOV	–	4,14%	8,62%
Počet konceptů v sémantických stromech	9027	1017	2305
Počet unikátních konceptů	25	24	24
ø počet konceptů 1 věty ($\pm\sigma$)	2,08±1,60	2,17±1,62	2,15±1,61
Počet vět s 1 konceptem	2662	274	637
Počet vět označených OOT	915	74	210
ø počet konceptů 1 věty ($\pm\sigma$), má-li věta více než 1 koncept	3,80±1,34	3,81±1,30	3,83±1,29

Tabulka 5: Vlastnosti korpusu TIA.

8.1 Metriky použité pro vyhodnocení

Byla navržena metodika vyhodnocení využívající algoritmus pro výpočet editační vzdálenosti mezi dvěma abstraktními sémantickými stromy. Mějme seznam L, tvořený editačními operacemi l_i ve tvaru:

- $a \to a$ (shoda)
- $a \to b$ (substituce konceptu a konceptem b)
- $a \to \lambda$ (odstranění konceptu a)
- $\lambda \to b$ (vložení konceptu b)

Editační vzdálenost $D(T_1, T_2)$ mezi stromy T_1 a T_2 je pak definována jako:

$$D(T_1, T_2) = \min_L \{\gamma(L) | L \text{ je posloupnost editačních operací převádějících } T_1 \text{ na } T_2\}$$

kde $\gamma(L) = \sum_i \gamma(l_i)$ je suma cena přiřazených jednotlivým editačním operacím.

Předpokládejme testovací množinu $\mathcal{T}_e = \{(u_i, s_i)\}_{i=1}^n$, kde u_i je vstupní promluva a s_i je odpovídající referenční sémantický strom. Při vyhodnocení přesnosti daného modelu je pro vstupní promluvu u_i vygenerován predikovaný, hypotetický sémantický strom $\hat{s_i}$. Označme seznam editační operací převádějících s_i na $\hat{s_i}$ jako L_i^*:

$$L_i^* = \text{argmin}_{L_i} \{\gamma(L_i) | L_i \text{ je posloupnost editačních operací převádějících } s_i \text{ na } \hat{s_i}\}$$

Definujme funkci $\delta_l(\cdot, \cdot)$ jako:

$$\delta_l(a, b) = \begin{cases} 1 & \text{pokud } l \text{ je ve tvaru } a \to b \\ 0 & \text{jinak} \end{cases} \tag{36}$$

Potom můžeme pro danou testovací množinu $\mathcal{T}_e$ zadefinovat čísla:

- Počet správných konceptů: $H = \sum_{i=1}^n \sum_{l \in L_i^*} \delta_l(a, a)$
- Počet referenčních konceptů: $N = \sum_{i=1}^n |s_i|$
- Počet chyb vynechání: $D = \sum_{i=1}^n \sum_{l \in L_i^*} \delta_l(a, \lambda)$
- Počet chyb vložení: $I = \sum_{i=1}^n \sum_{l \in L_i^*} \delta_l(\lambda, a)$
- Počet chyb substituce: $S = \sum_{i=1}^n \sum_{l \in L_i^*} \delta_l(a, b), \quad a \neq b$

Definujme *konceptovou přesnost cAcc* a *konceptovou správnost cCorr*. Tyto míry slouží k vyhodnocení přesnosti predikce sémantických stromů tvořených sémantickými koncepty, přičemž jsou schopny detailně podchytit i míru shody mezi dvěma různými stromy:

$$cAcc = \frac{N - D - S - I}{N} = \frac{H - I}{N} \tag{37}$$

Pro vyhodnocení přesnosti modulu detekce sémantických entit byla použita modifikovaná ROC křivka (Receiver Operating Characteristic). Ta vychází z klasické ROC křivky pro binární klasifikátor. Číslo *TP* (True Positives) znamená, kolikrát klasifikátor správně predikoval třídu 1; *FP* (False Positives) kolikrát klasifikátor chybně predikoval třídu 1, pokud v referenčních datech byla třída 0; *FN* (False Negatives) kolikrát klasifikátor chybně predikoval třídu 0, pokud v referenčních datech byla třídu 1; *TN* (True Negatives) kolikrát klasifikátor správně predikoval třídu 0. Následně můžeme definovat veličiny *TPR* (True Positives Rate) a *FPR* (False Positives Rate):

$$TPR = \frac{TP}{TP + FN}, \quad FPR = \frac{FP}{FP + TN} \tag{38}$$

Binární klasifikátor často pro neznámý příznakový vektor $\mathbf{x}$ nevrací pouze predikci cílové třídy, ale i určité skóre $d(\mathbf{x})$. Toto skóre může odpovídat například aposteriorní pravděpodobnosti nebo vzdálenosti k oddělovací nadrovině. Jako rozhodovací pravidlo je pak použito porovnání s daným prahem θ:

$$\hat{y} = \begin{cases} 1 & \text{pokud } d(\mathbf{x}) \geq \theta \\ 0 & \text{pokud } d(\mathbf{x}) < \theta \end{cases} \tag{39}$$

ROC křivka je pak definována jako křivka vyjadřující závislost *TPR* na *FPR* pro spojitě se měnící hodnotu prahu θ. ROC křivka začíná v bodě $[0; 0]$ a končí v bodě $[1; 1]$. Její průběh pomáhá určit vhodný pracovní bod klasifikátoru pomocí prahu θ, kdy poměr správně detekovaných příkladů (*TPR*) a četností falešných poplachů (*FPR*) odpovídá cílové úloze.

Pro jiné úlohy než pro binární klasifikaci není vhodné veličinu *FPR* počítat jako poměr počtu falešných poplachů (*FP*) a celkového počtu negativních příkladů (třída 0), neboť počet různých sémantických entit může být nekonečný a tudíž i počet negativních příkladů může být nekonečný. Proto zavádíme modifikovanou ROC křivku, která vyjadřuje závislost veličiny *TPR* na veličině FPR_{norm} při proměnném prahu θ. Normalizovaná četnost falešných poplachů FPR_{norm} je pak definována jako $FPR_{norm} = \frac{FP}{n}$, kde n je počet jednotek, ke kterému je vztažen počet falešných poplachů. V disertační práci bude jako n použit počet promluv v testovací množině. Veličina FPR_{norm} pak vyjadřuje relativní četnost falešných poplachů na jednu promluvu.

Pro porovnání různých modifikovaných ROC křivek je možné použít hodnotu získanou jako velikost plochy pod modifikovanou ROC křivkou – *AUC* (Area Under the Curve). V disertační práci budeme tuto plochu počítat z hodnot *TPR* pro FPR_{norm} z intervalu $< 0; 1 >$:

$$AUC = \int_0^1 TPR(FPR_{norm}) dFPR_{norm} \tag{40}$$

8.2 Systém automatického rozpoznávání řeči

Pro parametrizaci nahrávek byla využita perceptivní lineární prediktivní analýza (PLP, [16]) se 12 koeficienty a delta- a delta-delta- koeficienty. Jako akustické modely byly použity standardní třístavové levopravé HMM modely trifónů s 2000 stavy. Výstupní hustoty pravděpodobností byly modelovány jako směs Gaussovských rozdělení (Gaussian Mixture Model, GMM) se 16 složkami na stav. Systém automatického rozpoznávání řeči je implementován jako rozpoznávač spojité řeči s rozsáhlým slovníkem (Large Vocabulary Continuous Speech Recognizer, LVCSR) [42].

Pro rozpoznání korpusu HHTT dostupná databáze všech vlakových zastávek a stanic umožnila vytvoření trigramového jazykového modelu s třídami, kde jednotlivé třídy reprezentovaly stanice v různých gramatických pádech. Trigramový jazykový model pro úlohu TIA byl získán z trénovacích dat totožných s daty popsanými v tabulce 5.

Pro experimenty s porozuměním řeči založeným na fonémových jazykových modelech byly pro úlohu HHTT a TIA vytvořeny fonémové jazykové modely. Na základě experimentů s rozpoznáváním založeným na fonémech byly použity 5-gramové fonémové jazykové modely, neboť tyto modely poskytovaly nejvyšší fonémovou přesnost rozpoznávání (*Acc*), přičemž byla zachována schopnost dekódovat promluvy v reálném čase.

Byly použity tři různé typy fonémových jazykových modelů – jazykový model trénovaný z fonémově zarovnané přepisu (*ph-fa*), obecný jazykový model z korpusu *Bezplatné hovory* (BH, *ph-bh*) a jazykový model z korpusu BH adaptovaný na cílovou doménu (*ph-ad*). Adaptace fonémových jazykových modelů byla uvažována jako učení bez učitele. Algoritmus adaptace byl následující:

1. Natrénování obecného fonémového jazykového modelu (např. z dat BH)
2. Rozpoznání adaptačních dat pomocí obecného fonémového jazykového modelu
3. Z výsledného hypotetického fonémového přepisu natrénování adaptovaného jazykového modelu
4. Použití adaptovaného jazykového modelu pro rozpoznávání

Bylo rovněž zkoumáno použití tzv. *pseudofonémových mřížek*, které byly automaticky vygenerovány ze slovních mřížek. Pro jejich vytvoření byl použit výslovnostní slovník specifický pro danou úlohu (HHTT nebo TIA). Výslovnostní slovník každému slovu z rozpoznávacího slovníku $\mathcal{V}$ přiřazuje možné posloupnosti fonémů – výslovnostní varianty. Pro snížení neurčitosti byla pro každé slovo ve výslovnostním slovníku ponechána pouze jediná výslovnostní varianta (ta s nejvyšším počtem fonémů). Takto vygenerované pseudofonémové mřížky budeme v následujícím textu označovat pomocí identifikátoru *ph-map*.

Vyhodnocení přesnosti rozpoznávání je možné najít v tabulce 8. Přesnost rozpoznávání je vyhodnocena pomoc slovní/fonémové přesnosti *Acc*. V tabulce je rovněž uvedena míra oracle accuracy (hodnoty v závorkách). Tato míra vyjadřuje slovní/fonémovou přesnost *Acc* vyhodnocenou nad hypotézou, která minimalizuje Levenshteinovu vzdálenost k referenčnímu přepisu (bez ohledu na váhu přiřazenou hypotéze). Oracle accuracy jistým způsobem vyjadřuje o kolik více informace je uloženo v mřížce v porovnání s první nejlepší hypotézou.

9 Experimentální ověření

Nejprve uveďme popis postupu realizovaného při experimentálním ověření popsaných metod nad sémantickými korpusy HHTT a TIA. V prvním kroce byla použita trénovací sada korpusu HHTT, která byla rozdělena v poměru 72:8:20 na sady train_t, train_d a train_e. Ty byly následně použity jako trénovací, development a testovací sada pro nastavení parametrů modelu HDM. Tento krok byl proveden, aby původní development a testovací sada korpusu HHTT nebyla při nastavování parametrů použita, čímž se zabrání vychýlení (přetrénování) nastavovaných parametrů na konkrétní data nebo dokonce rozdělení dat. V rámci nastavování parametrů modelu HDM byly postupně určovány parametry vstupní vrstvy, skryté vrstvy a výstupní vrstvy.

Po nastavení parametrů s využitím výše popsaných sad train_t, train_d a train_e byl natrénován HDM model nad celým korpusem HHTT a nad korpusem TIA. Dosažené výsledky jsou shrnuty v kapitole 9.1.

V kapitole 9.2 následuje vyhodnocení modelu pro detekci sémantických entit. Detailněji jsou popsány vlastnosti použitých gramatik a vyhodnocení pomocí modifikovaných ROC křivek a hodnot metriky AUC.

Model detekce sémantických entit a model zarovnání následně umožňují kombinaci s hierarchickým diskriminativním modelem pomocí pravděpodobnostních vztahů popsaných v kapitole 5. Tento výsledný diskriminativní model pro porozumění řeči je vyhodnocen z pohledu konceptové přesnosti opět na korpusy HHTT a TIA, výsledky tohoto vyhodnocení jsou popsány v kapitole 9.3.

Vzhledem k tomu, že model HDM byl navržen tak, aby jej bylo možné trénovat a následně provozovat nad velmi různorodými strukturami vstupních dat, setkáme se s následujícím označením:

- *Slovní přepis* – jedná se o přepis přiřazený dané promluvě člověkem anotátorem. Přepis není ovlivněn chybami automatického rozpoznávání řeči.
- *Slovní 1. hypotéza* – nejlepší slovní hypotéza ze systému automatického rozpoznávání řeči.

	HHTT		TIA
	operátor	*zákazník*	
Počet vět	30802	27862	4166
Počet tokenů	191360	210945	33562
Velikost slovníku	11282	12638	2655
PPL (devel)	66,9	67,0	36,1
PPL (test)	50,5	62,4	38,0

Tabulka 6: Vlastnosti jazykových modelů použitých pro rozpoznávání korpusů.

Model	*HHTT*		*TIA*	
	devel	*test*	*devel*	*test*
ph-fa	7,02	6,59	8,08	8,26
ph-bh	15,63	15,76	20,82	20,36
ph-ad	8,25	7,79	11,54	11,65

Tabulka 7: Perplexity fonémových jazykových modelů, *ph-fa* – fonémový jazykový model získaný ze zarovnaných slovních přepisů, *ph-bh* – obecný jazykový model získaný z korpusu *ph-bh*, *ph-ad* – fonémový jazykový model *ph-bh* adaptovaný na cílovou úlohu (HHTT nebo TIA).

Model		*HHTT*		*TIA*	
		devel	*test*	*devel*	*test*
Slovní		70,5 (82,7)	72,9 (84,8)	72,4 (78,6)	62,5 (71,1)
Fonémový	*ph-fa*	74,7 (79,5)	76,2 (81,1)	77,2 (81,7)	68,5 (74,2)
	ph-bh	65,5 (75,4)	67,6 (76,7)	58,6 (70,8)	51,4 (64,0)
	ph-ad	72,5 (78,5)	74,4 (80,0)	69,4 (76,4)	61,8 (69,6)
Pseudofon.	*ph-map*	74,8 (79,3)	76,1 (80,3)	79,3 (82,0)	72,5 (76,7)

Tabulka 8: Přesnost rozpoznávání (*Acc* v procentech) pro slovní a fonémové jazykové modely použité v experimentální části. Hodnoty v závorkách odpovídají míře oracle accuracy, která vyjadřuje přesnost (*Acc*) té hypotézy z mřížky, která je nejbližší referenci.

- *Slovní mřížka* – výstupní mřížka ze systému automatického rozpoznávání řeči.
- *Fonémová 1. hypotéza* – nejlepší fonémová hypotéza z fonémového rozpoznávače.
- *Fonémová mřížka* – výstupní mřížka z fonémového rozpoznávače.
- *Pseudofonémová mřížka* – mřížka vzniklá převedením slovních mřížek na fonémové.

U dat, která reprezentují fonémy, je vždy ještě uveden fonémový jazykový model použitý k jejich vygenerování. Jsou použity zkratky *ph-fa*, *ph-bh*, *ph-ad* a *ph-map* popsané na straně 41.

HDM má řadu parametrů a metaparametrů, které budou v následujícím textu souhrně označeny pojmem parametry modelu:

- *Vstupní vrstva* je parametrizována použitou racionální jádrovou funkcí. Omezíme se pouze na n-gramové racionální jádrové funkce s minimálním řádem n a maximálním řádem m definované pomocí transduceru $T_{n,m}$ z rovnice (8). Volitelně je možné použít normalizaci racionální jádrové funkce.
- *Skrytá vrstva* – při nastavování parametrů modelu HDM je nejprve zodpovězena otázka samotného přínosu skryté vrstvy. Je porovnávána konceptová přesnost HDM modelu sestaveného pouze ze vstupní a výstupní vrstvy s HDM modelem používajícím všechny tři vrstvy. Dále je určen práh N determinující množinu sémantických n-tic $|\mathcal{S}_N|$. Nakonec je vybrán způsob nastavování regularizačního parametru $C^{\mathbf{t}}$ pro každý SVM klasifikátor predikující sémantickou n-tici $\mathbf{t}$.
- *Výstupní vrstva* je parametrizována jediným parametrem M určujícím trénovací množinu $\mathcal{T}_A$ pro klasifikátor g_A predikující potomky konceptu A ve výstupním sémantickém stromu.

Pro nastavení parametrů byla použita trénovací množina train_t (3815 promluv), development sada train_d (371 promluv) a testovací sada train_e (1054 promluv). Tabulka 9 shrnuje všechny parametry modelu HDM, jejichž volba je odůvodněna v disertační práci.

9.1 Vyhodnocení HDM nad neviděnými daty

Prezentujme nyní výsledky HDM nad korpusem HHTT při použití celé trénovací množiny a oddělené development a testovací sady nepoužité v předchozích experimentech. Výsledky budou rovněž porovnány s referenčními modely a vyhodnocení modelu HDM bude také provedeno nad korpusem TIA.

Typ dat	*Normalizace*	*n*	*m*	*Typ modelu*	*N*	*M*
Slovní přepis	ano	1	1	HDM-3	30	4
Slovní mřížka	ano	1	2	HDM-3	30	4
Fonémová mřížka	ano	1	5	HDM-3	30	4
Slovní 1. hypotéza	ano	1	2	HDM-3	30	4
Fonémová 1. hypotéza	ano	1	5	HDM-3	30	4
Pseudofon. mřížka	ano	1	5	HDM-3	30	4

Tabulka 9: Parametry hierarchického diskriminativního modelu použité v dalších experimentech. Sloupce *normalizace*, n a m ovlivňují vstupní vrstvu, *typ modelu* a N skrytou vrstvu a M vrstvu výstupní.

Porovnání výsledků hierarchického diskriminativního modelu a referenčních modelů přináší tabulky 10 a 11. Referenční HVS model byl trénován jako HVS parser s levopravým větvením bez parametrizace vstupním příznakovým vektorem, tj. pouze s využitím slov dané hypotézy. Tento model odpovídá modelu popsanému v rámci práce [22]. Model STC byl trénován jako HDM model avšak bez výstupní vrstvy, která byla nahrazena dekódovacím algoritmem popsaným v kapitole 4.1.

Nejprve je vhodné okomentovat výsledky dosažené referenčními modely a shrnuté v tabulce 10. Upozorněme na propad v konceptové přesnosti u modelu HVS při porovnání testovacích sad korpusu HHTT a TIA. Zde rozdíl činí téměř šest procentních bodů, pokud jsou modely trénovány a vyhodnoceny nad slovním přepisem od anotátora. Pokud však porovnáme konceptové přesnosti nad rozpoznanými daty (slovní 1. hypotéza), je tento rozdíl již téměř deset procentních bodů. Pokud se podíváme na výsledky nad development sadou, nejsou tyto rozdíly již tak markantní, lze se tedy domnívat, že při trénování HVS modelu nad korpusem TIA dochází k jistému přetrénování a následně k nízké přesnosti predikce sémantických stromů nad neviděnými daty.

Pro model STC došlo ke zlepšení konceptové přesnosti v porovnání s modelem HVS jak nad slovním přepisem od anotátora, tak nad rozpoznanými slovními posloupnostmi a to konzistentně jak pro development, tak pro testovací sady obou sémantických korpusů. Toto zlepšení je pro korpus HHTT přibližně čtyři procentní body, pro korpus TIA pak devět až deset procentních bodů.

Model HDM byl natrénován za použití parametrů z tabulky 9. Přínos tohoto modelu je zřejmý z tabulky 11. Podívejme se nejprve na výsledky získané při trénování ze slovního přepisu od anotátora a z první nejlepší rozpoznané hypotézy. Zde model HDM vylepšuje výsledky modelu STC nad korpusem HHTT o přibližně tři procentní body. V porovnání s modelem HVS jsou tyto výsledky lepší o více než osm procentních bodů nad slovním přepisem a o téměř sedm procentních bodů nad roz-

		HHTT		TIA	
Model	*Typ dat*	*devel*	*test*	*devel*	*test*
HVS	Slovní přepis	74,2	73,6	69,9	67,9
	Slovní 1. hypotéza	63,4	65,8	61,6	56,3
STC	Slovní přepis	78,7	78,0	73,7	76,8
	Slovní 1. hypotéza	67,4	69,2	69,5	66,4

Tabulka 10: Hodnoty konceptové přesnosti (*cAcc*) v procentech pro referenční modely.

poznanými daty. Mnohem markantnější rozdíl je nad sémantickým korpusem TIA, kde HDM přidává oproti modelu STC šest procentních bodů.

Z tabulky lze rovněž vyčíst přínos použití slovních mřížek v porovnání s první nejlepší slovní hypotézou ze systému automatického rozpoznávání řeči. Tento přínos je v případě korpusu HHTT přibližně jeden procentní bod a v případě korpusu TIA dva procentní body. Další řádky ukazují výsledky při použití fonémových mřížek namísto mřížek slovních. Je zde uveden i řádek odpovídající první nejlepší fonémové posloupnosti. Porovnáním tohoto výsledku s výsledky získanými z fonémových mřížek *ph-fa* lze konstatovat, že na fonémové úrovni je příspěvek ke konceptové přesnosti způsobený použitím mřížek ještě nižší než na úrovni mřížek slovních. Další řádky porovnávají konceptové přesnosti modelů trénovaných z mřížek generovaných pomocí různých fonémových jazykových modelů. Poslední řádek pak odkazuje na použití pseudofonémových mřížek, tj. fonémových mřížek vygenerovaných ze slovních mřížek pomocí výslovnostních slovníků. Diskuze výsledků bude uvedena v závěru na straně 52.

9.2 Detekce sémantických entit

Kromě hierarchického diskriminativního modelu byl v disertační práci popsán i model detekce sémantických entit. Model detekce sémantických entit byl ověřen na datech ze slovního rozpoznávače řeči v obou použitých úlohách, tj. jak nad korpusem HHTT, tak TIA. Pro ověření tohoto modelu bylo nutné získat expertně navržené gramatiky pro jednotlivé typy sémantických entit.

Gramatiky pro úlohu HHTT byly vyvinuty v rámci letního workshopu 2011 na Katedře kybernetiky Západočeské univerzity v Plzni. Hlasový dialogový systém vyvíjený v rámci tohoto workshopu byl zaměřen na vyvinutí znalostní verze hlasového dialogového systému pro úlohu poskytování informací o odjezdech a příjezdech vlaků [43]. Z tohoto hlasového dialogového systému byly použity gramatiky pro následu-

		HHTT		TIA	
Model	*Typ dat*	*devel*	*test*	*devel*	*test*
HDM	Slovní přepis	82,7	81,9	80,9	82,6
	Slovní 1. hypotéza	70,2	72,3	73,9	72,9
	Slovní mřížka	70,7	73,5	76,1	74,8
	Fonémová 1. hypotéza, *ph-fa*	67,0	70,9	73,8	70,9
	Fonémová mřížka, *ph-fa*	67,0	71,0	73,8	71,5
	Fonémová mřížka, *ph-bh*	61,5	66,5	67,5	65,7
	Fonémová mřížka, *ph-ad*	68,8	69,8	70,7	69,6
	Pseudofon. mřížka, *ph-map*	73,4	75,6	76,6	75,5

Tabulka 11: Hodnoty konceptové přesnosti (*cAcc*) v procentech pro hierarchický diskriminativní model trénovaný z různých typů dat.

jící typy sémantických entit: *station* (jméno stanice v různých pádech), *time* (časový údaj nebo údaj o datu) a *train_type* (typ vlaku).

Gramatiky pro úlohy TIA byly zpracovány v rámci výzkumného projektu MPO TIP FR-TI1/518 Inteligentní telefonní asistentka. Pro popsané řečové korpusy byly použity gramatiky reprezentující následující typy sémantických entit: *jmeno* (křestní jméno osoby, příjmení nebo jejich kombinace), *vec* (jméno prostředku k rezervaci), *t* (časový údaj), *datum* (údaj o datu).

Vlastnosti bezkontextových gramatik použitých k popisu sémantických entit a odpovídajících transducerů získaných jejich kompilací jsou zachyceny v tabulce 12. Pro vyhodnocení modelu detekce sémantických entit je použita modifikovaná ROC křivka. Dále je použita metrika odpovídající ploše pod touto křivkou při 0 až 1 falešném poplachu na jednu promluvu (AUC) definovaná v rovnici (40).

Vzhledem k tomu, že ani korpus HHTT ani korpus TIA neobsahují označené referenční sémantické entity, je nutné referenční data vygenerovat pomocí bezkontextových gramatik použitých následně i pro predikci. Tímto přístupem není možné vyhodnotit pokrytí sémantických entit gramatikami, tj. poměr sémantických entit detekovaných pomocí gramatik vůči všem referenčním sémantickým entitám. Je třeba zdůraznit, že maximalizace pokrytí zahrnuje netriviální množství expertní práce a proto není předmětem disertační práce, která se specializuje na kombinaci statistického a expertního přístupu, nikoli na zdokonalování stávajících expertních znalostí.

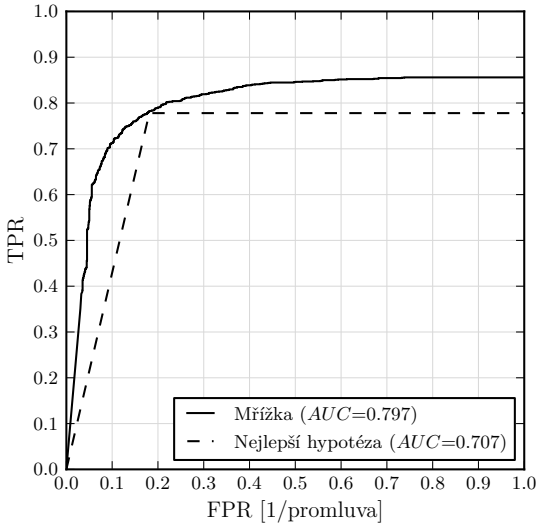
Referenční data pro detekci sémantických entit jsou vygenerována z referenčních textových přepisů jednotlivých promluv. Je tak možné vyčíslit především vliv systému

automatického rozpoznávání řeči na přesnost detekce sémantických entit. V experimentech se zaměříme na zodpovězení otázky, zda lze při detekci sémantických entit zužitkovat informaci obsaženou ve slovních mřížkách a na porovnání výsledků s přístupem, kdy se pro detekci sémantických entit používá první nejlepší hypotéza.

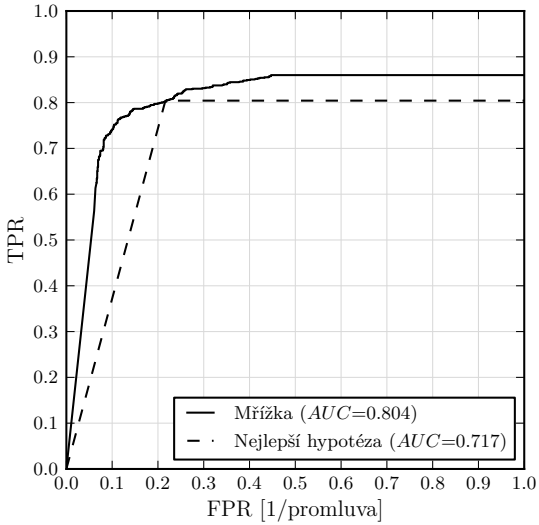
Detekce sémantických entit byla prováděna nad testovacími množinami korpusů HHTT a TIA. Modifikované ROC křivky a odpovídající hodnoty *AUC* jsou zachyceny na obrázcích 7 (HHTT) a 8 (TIA). Dvojice grafů na těchto obrázcích porovnává modifikované ROC křivky při detekci sémantických entit ze slovních mřížek a z první nejlepší hypotézy. Je vidět, že při detekci sémantických entit z *první nejlepší hypotézy* není možné precizně volit vyvážení detekční schopnosti (svislá osa) a počtu falešných poplachů (vodorovná osa). Pro oba korpusy lze zvolit optimální pracovní bod odpovídající přibližně 0,2 falešného poplachu na jednu promluvu při správné detekci přibližně 80% sémantických entit. Poznamenejme, že detekci sémantických entit z první nejlepší hypotézy lze provádět i jednodušším způsobem, než je postup popsaný v kapitole 7, příkladem budiž náhrada lexikálních tříd používaná v STC modelu (kapitola 4.1).

Přínos detekce sémantických entit z mřížek je zřejmý. Z tvaru modifikované ROC křivky lze vyčíst, že jak pro četnost falešných poplachů nižší než 0,2 na jednu promluvu, tak i pro četnost vyšší, lze docílit lepší detekční schopnosti. Například pro korpus TIA lze při snížení četnosti falešných poplachů na 0,1 na jednu promluvu (tj. na 50%) docílit stále velice přijatelné detekční schopnosti kolem 75%. Z hodnot na opačném konci vodorovné osy lze vyčíst, že ze slovní mřížky lze detekovat přibližně o 5% (absolutně) více sémantických entit v porovnání s první nejlepší hypotézou (nárůst z 80% na 85%).

Z výše uvedeného vyplývá, že detekce sémantických entit z mřížek má svůj přínos především při redukci četnosti falešných poplachů. Dále díky tomu, že poskytuje i aposteriorní pravděpodobnost výskytu dané posloupnosti sémantických entit ve vstupní promluvě, je vhodná i pro nasazení v hlasových dialogových systémech se stochastickým řízením dialogu, které jsou schopny tuto neurčitost zužitkovat. V neposlední řadě lze detekci sémantických entit použít i v kombinaci s hierarchickým diskriminativním modelem pro porozumění mluvené řeči.



Obrázek 7: ROC křivka pro detekci sémantických entit nad slovní mřížkou a nad první nejlepší hypotézou v korpusu HHTT.



Obrázek 8: ROC křivka pro detekci sémantických entit nad slovní mřížkou a nad první nejlepší hypotézou v korpusu TIA.

Korpus	*s.e. z*	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{B}\rvert$	$\lvert\mathcal{E}\rvert$	$\lvert\mathcal{Q}\rvert$	N	$\mathcal{C}(z)$
HHTT	station	7516	3005	34405	5564	417	STATION
	time	437	221	13375	2898	791	TIME
	train_type	16	10	24	11	140	TRAIN_TYPE
TIA	jmeno	105	30	243	31	56	JMENO
	vec	46	14	100	20	31	VEC
	t	159	42	2471	688	414	T, INTERVAL
	datum	324	180	10904	2210	359	DATUM, RELATIVNI

Tabulka 12: Tabulka shrnující vlastnosti gramatik a transducerů vzniklých jejich kompilací pro jednotlivé typy sémantických entit. Sloupce $\lvert\mathcal{A}\rvert$ a $\lvert\mathcal{B}\rvert$ vyjadřují velikost vstupní, resp. výstupní abecedy transduceru T_z získaného z gramatiky pro typ z, sloupce $\lvert\mathcal{E}\rvert$ a $\lvert\mathcal{Q}\rvert$ pak odpovídají počtu stavů, resp. přechodů v T_z a N zobrazuje celkový počet výskytů označených touto gramatikou v testovací sadě daného korpusu. V posledním sloupci $\mathcal{C}(z)$ je uvedena množina konceptů, na které lze sémantickou entitu typu z zarovnat v modelu zarovnání.

9.3 Kombinace HDM a detekce sémantických entit

Popišme nyní experimentální ověření kombinace detekce sémantických entit $\mathrm{P}(\mathsf{E}|\mathsf{U})$, konceptového modelu $\mathrm{P}(\mathsf{C}|\mathsf{E},\mathsf{U})$ a modelu zarovnání $\mathrm{P}(\mathsf{A}|\mathsf{E},\mathsf{C})$ podle teorie popsané v kapitole 5. V této kapitole budeme uvažovat konfiguraci modelu detekce sémantických entit shodnou s předchozí kapitolou 9.2. Jako konceptový model použijeme hierarchický diskriminativní model.

Pro kombinaci těchto modelů je nutné použít model zarovnání $\mathrm{P}(\mathsf{A}=1|\mathsf{E}=e,\mathsf{C}=c)$. Předpokládejme, že e je posloupnost sémantických entit e_i, entita e_i má typ z_i a množina sémantických konceptů, které mohou být zarovnány s daným typem z_i nazvěme $\mathcal{C}(z_i)$. Model zarovnání byl použit v následujícím tvaru:

$$\mathrm{P}(\mathsf{A}=1|\mathsf{E}=e,\mathsf{C}=c)=\lambda^n\cdot(1-\lambda)^m \tag{41}$$

kde n je počet sémantických entit z posloupnosti e, pro které lze jejich typ z_i zarovnat s některým sémantickým konceptem c_k ze stromu c (tj. $c_k\in\mathcal{C}(z_i)$). Dále m je počet sémantických entit, které výše uvedeným způsobem zarovnat nelze. Platí $|e|=n+m$. Hodnota parametru $\lambda=0,95$ byla experimentálně určena na development sadách použitých korpusů. Množiny $\mathcal{C}(z)$ byly určeny expertně a pro jednotlivé typy sémantických entit z jsou uvedeny v tabulce 12.

Pro vyhodnocení kombinace konceptového modelu (reprezentovaného hierarchickým diskriminativním modelem), modelu detekce sémantických entit a modelu zarovnání byl zrealizován experiment, kdy byla použita informace o sémantických entitách v HDM modelu a následně bylo provedeno zarovnání abstraktního sémantického stromu a posloupnosti sémantických entit, přičemž výsledná pravděpodobnost byla

		HHTT		TIA	
Model	*Typ dat*	*devel*	*test*	*devel*	*test*
konceptový model	Slovní přepis	82,7	81,9	80,9	82,6
$P(\mathsf{C}\|\mathsf{U})$	Slovní mřížka	70,7	73,5	76,1	74,8
	Pseudofon. mřížka	73,4	75,6	76,6	75,5
kombinace	Slovní přepis	83,8	83,4	82,1	83,4
$P(\mathsf{C}\|\mathsf{E},\mathsf{U})\cdot$	Slovní mřížka	72,2	76,2	77,2	75,1
$\cdot P(\mathsf{E}\|\mathsf{U})P(\mathsf{A}\|\mathsf{E},\mathsf{C})$	Pseudofon. mřížka	74,1	76,9	76,7	75,4

Tabulka 13: Porovnání konceptové přesnosti samotného konceptového modelu a kombinace konceptového modelu, modelu detekce sémantických entit a modelu zarovnání.

vážena pravděpodobností přiřazenou modelem zarovnání z rovnice (41). Tento postup je ekvivalentní reskórování (přeuspořádání) n-nejlepších sémantických stromů vzhledem k možným posloupnostem sémantických entit.

Při trénování modelu HDM byla použita informace z modelu detekce sémantických entit. Výsledný model pak predikuje pravděpodobnost $P(\mathsf{C}|\mathsf{E},\mathsf{U})$. Z mřížky sémantických entit E byly získány střední počty výskytů $cnt(E, z_k)$ jednotlivých typů sémantických entit z_k a z nich byl sestaven vektor příznaků $\mathbf{d}(E) = [cnt(E, z_k)]$, přičemž pro korpus HHTT bylo $z_k \in \{station, time, train\_type\}$ a pro korpus TIA $z_k \in \{jmeno, vec, t, datum\}$. Následně byl vektor příznaků na výstupu skryté vrstvy doplněn o vektor $\mathbf{d}(E)$.

Pro vyhodnocení byla použita míra *cAcc* abstraktního sémantického stromu predikovaného kombinovaným modelem. Tabulka 13 zobrazuje konceptové přesnosti dosahované samotným HDM modelem a modelem, ve kterém je použita kombinace HDM s detekcí sémantických entit a modelem zarovnání. Z tabulky je zřejmé, že využitím informace o sémantických entitách lze dosáhnout zvýšení přesnosti predikce abstraktních sémantických stromů. K tomuto nárůstu dochází konzistentně na různých typech dat a zároveň jak na development, tak testovací sadě.

10 Závěr

Vytvořený diskriminativní model pro porozumění řeči je vyhodnocen nad testovacími sadami korpusů HHTT a TIA. Výsledky porozumění z referenčního slovního přepisu (tabulka 14), porozumění z rozpoznaných slov (tabulka 15) a porozumění z rozpoznaných fonémů (tabulka 16) jsou vyjádřeny v podobě konceptové přesnosti *cAcc* spolu s 95% intervalem spolehlivosti v podobě větné přesnosti *sAcc*.

První modelový problém – *porozumění z referenčního slovního přepisu* (tabulka 14) – srovnává výsledky referenčních modelů (HVS parser a STC model) s HDM modelem a s kombinací HDM modelu a modelu detekce sémantických entit. Tento modelový problém odpovídá porozumění z textového přepisu bez neurčitosti způsobené systémem automatického rozpoznávání řeči. Přestože HDM model nebyl pro tuto úlohu primárně navržen, ukazuje se, že i zde překonává výsledky obou referenčních modelů.

Druhý modelový problém – *porozumění z rozpoznaných slov* (tabulka 15) – srovnává opět výsledky referenčních modelů s HDM modelem a odpovídá skutečnému nasazení v hlasových dialogových systémech. V tabulce je zobrazen nárůst konceptové přesnosti při použití slovních mřížek namísto první nejlepší hypotézy a dále při převedení slovních mřížek na mřížky pseudofonémové. Jako poslední modifikace je k HDM trénovanému z pseudofonémových mřížek přidána detekce sémantických entit.

Třetí modelový problém – *porozumění z rozpoznaných fonémů* (tabulka 16) – odpovídá situaci, kdy není dostupné dostatečné množství dat pro získání přesného a aktuálního jazykového modelu. Pro redukci objemu prací nutných k získání slovního přepisu je možné v tomto případě použít adaptaci fonémového jazykového modelu. Tabulka ukazuje vývoj konceptové přesnosti nejprve při trénování z fonémových mřížek získaných použitím obecného fonémového jazykového modelu trénovaného z korpusu BH (řádek *ph-bh*), dále pak použitím adaptovaného fonémového jazykového modelu (řádek *ph-ad*) a použitím fonémového jazykového modelu získaného ze zarovnaného slovního přepisu (řádek *ph-fa*). Pro srovnání je uveden i řádek odpovídající pseudofonémovým mřížkám generovaným ze slovních mřížek (řádek *ph-map*).

Z tabulky je vidět, že použití adaptovaných fonémových jazykových modelů zvýší přesnost porozumění v porovnání s obecným fonémovým jazykovým modelem, dosahované výsledky se téměř blíží výsledkům modelu trénovaného z mřížek získaných s využitím fonémového jazykového modelu ze zarovnaných slovních přepisů. Tento argument podporuje závěr, že na základě rozpoznané fonémové mřížky lze velice jednoduše provádět klasifikaci promluvy, a to nejen jednoduchými značkami, ale lze jim pomocí modelu HDM přiřazovat i komplexnější struktury v podobě abstraktních sémantických stromů.

Výsledek nad pseudofonémovými mřížkami také napovídá, že použitím výrazně lepšího fonémového rozpoznávače (srovnejte přesnost rozpoznávání při použití jednotlivých fonémových jazykových modelů uvedenou v tabulce 8) lze snadno dosáhnout lepších výsledků než při použití samotných slovních mřížek.

Přestože zde prezentované metody byly ověřeny nad sémantickými korpusy HHTT a TIA, které obsahovaly pouze promluvy v češtině, nejsou tyto metody omezeny pouze na tento jazyk. I když nárůst konceptové přesnosti při použití pseudofonémo-

	HHTT			TIA		
Model/modifikace	*cAcc*	95% i.s.	*sAcc*	*cAcc*	95% i.s.	*sAcc*
HVS	73,6	71,5÷75,8	68,9	67,9	65,2÷70,5	55,4
STC	78,0	76,1÷80,0	73,3	76,8	74,1÷79,5	73,4
slovní přepis	81,9	80,1÷83,7	77,0	82,6	80,3÷84,9	78,6
+ detekce sém. entit	**83,4**	81,7÷85,1	78,6	**83,4**	81,2÷85,7	79,0

Tabulka 14: Porozumění z referenčního slovního přepisu

	HHTT			TIA		
Model/modifikace	*cAcc*	95% i.s.	*sAcc*	*cAcc*	95% i.s.	*sAcc*
HVS	65,8	63,4÷68,2	63,2	56,3	52,7÷59,8	46,9
STC	69,2	66,9÷71,5	66,0	66,4	63,4÷69,3	59,7
1. hypotéza	72,3	70,2÷74,5	68,8	72,9	70,2÷75,6	66,0
slovní mřížka	73,5	71,4÷75,7	70,0	74,8	72,2÷77,4	68,2
pseudofon. mřížka	75,6	73,6÷77,6	71,7	75,5	73,0÷78,0	68,5
+ detekce sém. entit	**76,9**	75,0÷78,9	71,5	**75,4**	72,8÷78,1	69,1

Tabulka 15: Porozumění z rozpoznaných slov

	HHTT			TIA		
Model/modifikace	*cAcc*	95% i.s.	*sAcc*	*cAcc*	95% i.s.	*sAcc*
ph-bh	66,5	63,4÷68,7	63,1	65,7	62,8÷68,6	59,5
ph-ad	69,8	67,7÷72,0	66,1	69,6	66,8÷72,4	62,5
ph-fa	71,0	68,9÷73,1	67,0	71,5	68,7÷74,2	64,3
ph-map (pseudofon.)	75,6	73,6÷77,6	71,7	75,5	73,0÷78,0	68,5

Tabulka 16: Porozumění z rozpoznaných fonémů

Vysvětlivky k tabulkám 14-16: Tabulky shrnující výsledky pro tři modelové úlohy. Sloupec *cAcc* vyjadřuje konceptovou přesnost v %, ve sloupci 95% i.s. jsou zaneseny 95% intervaly spolehlivosti konceptové přesnosti v % a sloupec *sAcc* zobrazuje větnou přesnost v % (procento dekódovaných abstraktních sémantických stromů, které se shodují s referenční anotací). Výsledky byly vyhodnoceny nad testovacími sadami korpusů HHTT a TIA. Řádky uvedené jako HVS a STC obsahují výsledky referenčních modelů.

vých mřížek (oproti mřížkám slovním) lze z jisté části přičíst flektivní povaze češtiny, může uvedené předzpracování vstupních dat přinést jistá pozitiva i pro neflektivní jazyky, obzvláště při použití systému automatického rozpoznávání řeči.

Závěrečné kapitola disertační práce obsahuje shrnutí vytyčených cílů spolu s výsledky a odkazy na příslušná místa disertační práce. Uveďme ještě krátce další možné směry výzkumu a využití navrženého modelu:

Hlasové dialogové systémy

Popsaný model porozumění byl navrhován s ohledem na konkrétní použití v hlasových dialogových systémech. Bude použit jako modul porozumění řeči v hlasovém dialogovém systému pro podávání informací o odjezdech a příjezdech vlaků a v hlasovém dialogovém systému telefonní inteligentní asistentka, které jsou vyvíjené na Katedře kybernetiky Západočeské univerzity v Plzni.

Porozumění řeči a audiovizuální archivy

Kromě využití v oboru porozumění řeči se zdá velice perspektivní možnost rychlého vyhledávání výskytu sémantických entit v audiovizuálních archivech. Vzhledem k tomu, že tyto archivy lze postavit na metodách využívajících výhod faktorového automatu, je doplnění těchto vyhledávacích metod o detekci sémantických entit triviálním a z pohledu praktických aplikací velice žádaným rozšířením.

Poloautomatická klasifikace a shlukování hovorů

Zde lze s výhodou použít adaptované fonémové jazykové modely, které z pohledu porozumění řeči a klasifikace promluvy umožňují dosáhnout obdobné přesnosti jako fonémové jazykové modely trénované ze zarovnaných slovních přepisů. Přestože výsledky dosahované při porozumění z dat na slovní úrovni jsou vyšší, jsou vykoupeny nutností sestavit slovní jazykový model a rozpoznávací slovník. V praktických úlohách však toto může být velice problematické, neboť získání vhodných trénovacích dat nemusí být možné nebo data mohou mít velice dynamickou povahu, například v čase velice proměnný slovník. Potom lze s výhodou použít právě metody založené na použití fonémového rozpoznávače.

Literatura

Uvedeny pouze zdroje použité v autoreferátu, úplný seznam použitých pramenů je součástí disertační práce.

[1] Alan M. Turing. "Computing Machinery and Intelligence". In: *Mind* 59.236 (1950), s. 433–460. ISSN: 00264423.

[2] Gokhan Tur a Renato De Mori. *Spoken language understanding: Systems for extracting semantic information from speech*. Hoboken, New Jersey: Wiley, 2011. ISBN: 978-0-470-68824-3.

[3] D. Jurafsky, J.H. Martin, A. Kehler, et al. *Speech and language processing*. New York: Prentice Hall, 2000. ISBN: 978-0-13-187321-6.

[4] Frederick Jelinek. "Continuous Speech Recognition by Statistical Methods". In: *Proceedings of IEEE* 64.4 (1976), s. 532–556. ISSN: 0018-9219. DOI: 10.1109/PROC.1976.10159.

[5] Steve Young. *Frederick Jelinek 1932 – 2010: The Pioneer of Speech Recognition Technology*. 2010.

[6] Mehryar Mohri, Fernando C. N. Pereira a Michael Riley. "Weighted automata in text and speech processing". In: *Proceedings of the 12th biennial European Conference on Artificial Intelligence*. Budapest: John Wiley a Sons, 1996, s. 46–50.

[7] Mehryar Mohri, Fernando C. N. Pereira a Michael Riley. "Weighted finite-state transducers in speech recognition". In: *Computer Speech & Language* 16.1 (2002), s. 69–88. DOI: 10.1006/csla.2001.0184.

[8] Mehryar Mohri. "Weighted automata algorithms". In: *Handbook of weighted automata* (2009).

[9] Jason Eisner. "Expectation semirings: Flexible EM for learning finite-state transducers". In: *Proceedings of the ESSLLI workshop on finite-state methods in NLP*. August. 2001, s. 1–5.

[10] Corinna Cortes a Patrick Haffner. "Rational kernels: Theory and algorithms". In: *The Journal of Machine Learning* 5 (2004), s. 1035–1062.

[11] Joseph Weizenbaum. "ELIZA—a computer program for the study of natural language communication between man and machine". In: *Communications of the ACM* 9.1 (led. 1966), s. 36–45. ISSN: 00010782. DOI: 10.1145/365153.365168.

[12] Steve Young. "Talking to machines (statistically speaking)". In: *Proceedings of International Conference on Spoken Language Processing*. Denver, 2002, s. 9–16.

[13] Steve Young. “Still talking to machines (cognitively speaking)”. In: *Keynote Interspeech*. 2. Chiba, Japan: International Speech Communication Association, 2010, s. 10.

[14] J. Psutka, L. Müller, J. Matoušek, et al. *Mluvíme s počítačem česky*. Praha: Academia, 2006, s. 752. ISBN: 80-200-1309-1.

[15] Paul Mermelstein. “Distance measures for speech recognition, psychological and instrumental”. In: *Pattern recognition and artificial intelligence* (1976), s. 374–388.

[16] Hynek Heřmanský. “Perceptual linar predictive (PLP) analysis of speech”. In: *Journal of the Acoustical Society of America* 87.4 (1990), s. 1738–1752. DOI: 10.1121/1.399423.

[17] M. Oerder a Hermann Ney. “Word graphs: an efficient interface between continuous-speech recognition and language understanding”. In: *IEEE International Conference on Acoustics Speech and Signal Processing*. Minneapolis: Ieee, 1993, 119–122 vol.2. ISBN: 0-7803-0946-4. DOI: 10.1109/ICASSP.1993.319246.

[18] Daniel Povey, Mirko Hannemann, Gilles Boulianne, et al. “Generating Exact Lattices in the WFST Framework”. In: *IEEE International Conference on Acoustics Speech and Signal Processing*. Sv. 213850. 102. Kyoto, Japan: IEEE, 2012, s. 4213–4216. ISBN: 978-1-4673-0044-5. DOI: 10.1109/ICASSP.2012.6288848.

[19] V. I. Levenshtein. “Binary Codes Capable of Correcting Deletions, Insertions and Reversals”. In: *Doklady Akademii Nauk SSSR* 163.4 (1965), s. 845–848.

[20] Roberto Pieraccini, Evelyne Tzoukermann, Zakhar Gorelov, et al. “Progress report on the Chronus system: ATIS benchmark results”. In: *Proceedings of the workshop on Speech and Natural Language*. Harriman, New York: Association for Computational Linguistics, 1992, s. 67–71. ISBN: 1-55860-272-0. DOI: 10.3115/1075527.1075543.

[21] Christian Raymond a Giuseppe Riccardi. “Generative and Discriminative Algorithms for Spoken Language Understanding”. In: *Proceedings of Interspeech 2007*. International Speech Communication Association, 2007.

[22] Filip Jurčíček. “Statistical approach to the semantic analysis of spoken dialogues”. Dis. University of West Bohemia, 2007.

[23] Yulan He a Steve Young. “Semantic processing using the Hidden Vector State model”. In: *Computer Speech & Language* 19.1 (led. 2005), s. 85–106. ISSN: 08852308. DOI: 10.1016/j.csl.2004.03.001.

[24] Corinna Cortes a Vladimir Vapnik. “Support-Vector Networks”. In: *Machine learning* 20.3 (1995), s. 273–297. DOI: 10.1007/BF00994018.

[25] Christopher CJ Burges. "A tutorial on support vector machines for pattern recognition". In: *Data mining and knowledge discovery* 2.2 (1998), s. 121–167. DOI: 10.1023/A:1009715923555.

[26] Chih-wei Hsu a Chih-jen Lin. "A comparison of methods for multiclass support vector machines". In: *IEEE Transactions on Neural Networks* 13 (2002), s. 415–425.

[27] John C. Platt. *Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods*. Ed. A.J. Smola, P.L. Bartlett, B. Schölkopf, et al. Cambridge: MIT Press, 2000, s. 61–74.

[28] Cyril Allauzen, Michael Riley a Johan Schalkwyk. "OpenFst: A general and efficient weighted finite-state transducer library". In: *Implementation and Application of Automata* 4783 (2007), s. 11–23. DOI: 10.1007/978-3-540-76336-9_3.

[29] Mehryar Mohri, Pedro Moreno a Eugene Weinstein. "Factor automata of automata and applications". In: *Implementation and Application of Automata* 4783 (2007), s. 168–179. DOI: 10.1007/978-3-540-76336-9_17.

[30] Yves Schabes a RC Waters. "Lexicalized context-free grammars". In: *Proceedings of the 31st annual meeting on Association for Computational Linguistics*. Stroudsburg: Association for Computational Linguistics, 1993, s. 121–129. DOI: 10.3115/981574.981591.

[31] Yulan He a Steve Young. "Hidden vector state model for hierarchical semantic parsing". In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003.1* 1 (2003), s. 268–271. DOI: 10.1109/ICASSP.2003.1198769.

[32] Jan Švec. "Sémantická analýza promluv systému NÁDRAŽÍ". Diplomová práce. Katedra kybernetiky, Západočeská univerzita v Plzni, 2007, s. 74.

[33] François Mairesse, Milica Gašić, Filip Jurčíček, et al. "Spoken language understanding from unaligned data using discriminative classification models". In: *IEEE International Conference on Acoustics, Speech and Signal Processing, 2009. ICASSP 2009.* Taipei: IEEE, 2009, s. 4749–4752. ISBN: 978-1-4244-2353-8. DOI: 10.1109/ICASSP.2009.4960692.

[34] Deborah A. Dah, Madeleine Bates, Michael Brown, et al. "Expanding the scope of the ATIS task: The ATIS-3 corpus". In: *Proceedings of the workshop on Human Language Technology*. Stroudsburg, 1994, s. 43–48. ISBN: 1-55860-357-3. DOI: 10.3115/1075812.1075823.

[35] Dogan Can a Murat Saraclar. "Lattice Indexing for Spoken Term Detection". In: *IEEE Transactions on Audio, Speech and Language Processing* 19.8 (2011), s. 2338–2347. DOI: 10.1109/TASL.2011.2134087.

[36] A. Hunt a S. McGlashan. *Speech Recognition Grammar Specification Version 1.0*. W3C Recommendation. Břez. 2004.

[37] Fernando C. N. Pereira a Rebecca N. Wright. "Finite-state approximation of phrase structure grammars". In: *Proceedings of the 29th annual meeting on Association for Computational Linguistics ACL '91*. Berkeley: Association for Computational Linguistics, 1991, s. 246–255. DOI: 10.3115/981344.981376.

[38] Alan W. Black. "Finite state machines from feature grammars". In: *Finite State Machines from Feature Grammars*. Ed. Masaru Tomita. Pittsburgh: Carnegie Mellon University, 1989, s. 277–285.

[39] Mehryar Mohri a Fernando C. N. Pereira. "Dynamic compilation of weighted context-free grammars". In: *Proceedings of the 17th international conference on Computational linguistics*. Montreal, Quebec, Canada: Association for Computational Linguistics, 1998, s. 891–897. DOI: 10.3115/980432.980716.

[40] Michael Brown a Bruce Buntschuh. "A Context-free Grammar Compiler for Speech Understaing System". In: *Proceedings of 3rd International Conference on Spoken Language Processing (ICSLP 94)*. September. Yokohama: ISCA, 1994, s. 21–24.

[41] Filip Jurčíček, Jiří Zahradil a Libor Jelínek. "A human-human train timetable dialogue corpus". In: *Proceedings of EUROSPEECH, Lisboa* (2005), s. 1525–1528.

[42] Josef Psutka, Jan Švec, Josef V. Psutka, et al. "System for Fast Lexical and Phonetic Spoken Term Detection in a Czech Cultural Heritage Archive". In: *EURASIP Journal on Audio, Speech, and Music Processing* 2011.1 (2011), s. 10. ISSN: 1687-4722. DOI: 10.1186/1687-4722-2011-10.

[43] Tomáš Valenta, Jan Švec a Luboš Šmídl. "Spoken Dialogue System Design in 3 Weeks". In: *Text, Speech and Dialogue* 7499.IV (2012), s. 624–631. ISSN: 0302-9743. DOI: 10.1007/978-3-642-32790-2_76.

Seznam publikací

[A1] Jáchym Kolář, Jan Švec a Josef Psutka. "Automatic punctuation annotation in Czech broadcast news speech". In: *SPECOM´ 2004*. Saint-Petersburg: SPIIRAS, 2004, s. 319–325. ISBN: 5-7452-0110-X.

[A2] Jáchym Kolář, Jan Švec, Stephanie Strassel, et al. "Czech spontaneous speech corpus with structural metadata". In: *Proceedings of Interspeech 2005*. Lisbon: International Speech Communication Association, 2005, s. 1165–1168.

[A3] Filip Jurčíček, Jan Švec, Jiří Zahradil, et al. "Use of negative examples in training the HVS semantic model". In: *Text, Speech and Dialogue* 4188 (2006), s. 605–612. ISSN: 0302-9743. DOI: 10.1007/11846406_76.

[A4] Jan Švec, Filip Jurčíček a Luděk Müller. "Parameterization of the Input in Training the HVS Semantic Parser". In: *Text, Speech and Dialogue* 4629 (2007), s. 415–422. ISSN: 0302-9743. DOI: 10.1007/978-3-540-74628-7_54.

[A5] Jan Švec. "Sémantická analýza promluv systému NÁDRAŽÍ". Diplomová práce. Katedra kybernetiky, Západočeská univerzita v Plzni, 2007, s. 74.

[A6] Jáchym Kolář a Jan Švec. "Structural Metadata Annotation of Speech Corpora: Comparing Broadcast News and Broadcast Conversations". In: *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*. Sv. 94. Marrakech, Morocco: European Language Resources Association, 2008. ISBN: 2-9517408-4-0.

[A7] Filip Jurčíček, Jan Švec a Luděk Müller. "Extension of HVS semantic parser by allowing left-right branching". In: *Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing 2008*. 1. IEEE, 2008, s. 4993–4996. ISBN: 1424414849. DOI: 10.1109/ICASSP.2008.4518779.

[A8] Aleš Pražák, Pavel Ircing, Jan Švec, et al. "Efficient combination of n-gram language models and recognition grammars in real-time LVCSR decoder". In: *Proceedings of 9th International Conference on Signal Processing, 2008*. Beijing: IEEE, 2008, s. 587–591. ISBN: 978-1-4244-2178-7. DOI: 10.1109/ICOSP.2008.4697201.

[A9] Jan Švec. "Hlasové dialogové systémy". Odborná práce ke státní doktorské zkoušce. Katedra kybernetiky, Západočeská univerzita v Plzni, 2009, s. 45.

[A10] Jan Švec a Jiří Zahradil. "BigList: Speech-based selection of items from huge lists". In: *Proceedings of the 9th WSEAS international conference on signal, speech and image processing*. Budapest: World Scientific, Engineering Academy, a Society, 2009, s. 57–62. ISBN: 978-960-474-114-4.

[A11] Jáchym Kolář a Jan Švec. “The Czech Broadcast Conversation Corpus”. In: *Text, Speech and Dialogue* 5729 (2009), s. 101–108. ISSN: 0302-9743. DOI: 10.1007/978-3-642-04208-9_17.

[A12] Jan Švec a Filip Jurčíček. “Extended Hidden Vector State Parser”. In: *Text, Speech and Dialogue* 5729 (2009), s. 403–410. ISSN: 0302-9743. DOI: 10.1007/978-3-642-04208-9_55.

[A13] Josef Psutka, Jan Švec, Josef V. Psutka, et al. “Fast Phonetic/Lexical Searching in the Archives of the Czech Holocaust Testimonies: Advancing Towards the MALACH Project Visions”. In: *Text, Speech and Dialogue* 6231.III (2010), s. 385–391. ISSN: 0302-9743. DOI: 10.1007/978-3-642-15760-8_49.

[A14] Josef Psutka, Jan Švec, Josef V. Psutka, et al. “System for Fast Lexical and Phonetic Spoken Term Detection in a Czech Cultural Heritage Archive”. In: *EURASIP Journal on Audio, Speech, and Music Processing* 2011.1 (2011), s. 10. ISSN: 1687-4722. DOI: 10.1186/1687-4722-2011-10.

[A15] Jan Švec a Luboš Šmídl. “Prototype of Czech Spoken Dialog System with Mixed Initiative for Railway Information Service”. In: *Text, Speech and Dialogue* 6231.IV (2010), s. 568–575. ISSN: 0302-9743. DOI: 10.1007/978-3-642-15760-8_72.

[A16] Jan Švec, Jan Hoidekr a Daniel Soutner. “Web Text Data Mining for Building Large Scale Language Modelling Corpus”. In: *Text, Speech and Dialogue* 6836 (2011), s. 356–363. ISSN: 0302-9743. DOI: 10.1007/978-3-642-23538-2_45.

[A17] Jan Švec a Luboš Šmídl. “Real-time large vocabulary spontaneous speech recognition for spoken dialog systems”. In: *Proceedings of 4th International Congress on Image and Signal Processing, 2011*. Sv. 5. Shanghai: IEEE, říj. 2011, s. 2431–2436. ISBN: 978-1-4244-9306-7. DOI: 10.1109/CISP.2011.6100773.

[A18] Tomáš Valenta, Jan Švec a Luboš Šmídl. “Spoken Dialogue System Design in 3 Weeks”. In: *Text, Speech and Dialogue* 7499.IV (2012), s. 624–631. ISSN: 0302-9743. DOI: 10.1007/978-3-642-32790-2_76.

[A19] Petr Stanislav a Jan Švec. “Unsupervised Synchronization of Hidden Subtitles with Audio Track Using Keyword Spotting Algorithm”. In: *Text, Speech and Dialogue* 7499.III (2012), s. 422–430. ISSN: 0302-9743. DOI: 10.1007/978-3-642-32790-2_51.

[A20] Jan Švec a Pavel Ircing. “Efficient Algorithm for Rational Kernel Evaluation in Large Lattice Sets”. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2013*. Vancouver, Canada: IEEE, 2013, s. 3133–3137. ISBN: 978-1-4799-0355-9.

[A21] Jan Švec, Luboš Šmídl a Pavel Ircing. “Hierarchical Discriminative Model for Spoken Language Understanding”. In: *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2013*. Vancouver, Canada: IEEE, 2013, s. 8322–8326. ISBN: 978-1-4799-0355-9.

[A22] Jan Lehečka a Jan Švec. “Improving Speech Recognition by Detecting Foreign Inclusions and Generating Pronunciations”. In: *Text, Speech and Dialogue* (2013), (accepted for publication).

[A23] Jan Švec a Luboš Šmídl. “On the Use of Phoneme Lattices in Spoken Language Understanding”. In: *Text, Speech and Dialogue* (2013), (accepted for publication).

[A24] Jan Vavruška, Jan Švec a Pavel Ircing. “Phonetic Spoken Term Detection in Large Audio Archive Using the WFST Framework”. In: *Text, Speech and Dialogue* (2013), (accepted for publication).