



UNIVERZITA KOMENSKÉHO
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
ÚSTAV INFORMATIKY

Martin Troják

Skloňovanie podstatných mien v
slovenčine pomocou neurónovej siete

Diplomová práca

Diplomový školiteľ: Ing. Igor Farkaš PhD.

BRATISLAVA

2005

Týmto prehlasujem, že som diplomovú prácu vypracoval samostatne s odbornou pomocou školiteľa.

Bratislava, apríl 2005

Martin Troják

Pod'akovanie:

Veľké poďakovanie patrí môjmu školiteľovi Ing. Igorovi Farkašovi, PhD. za cenné rady a konzultácie. Tiež chcem poďakovať Tonovi Vaškovi za rady pri optimalizácii algoritmu učenia neurónovej siete a v neposlednej rade doc. PaedDr. Jane Kesselovej (Kat. slovakistiky a odbornej didaktiky, Fak. humanitných a príř. vied, PU v Prešove) za zdroje a informácie týkajúce sa akvizície skloňovania u detí a Marinovi Trnkovi Dipl. Ing. (Spracovanie reči, ústav informatiky, SAV) za pomoc pri vytváraní vstupných dát.

Abstrakt

TROJÁK, Martin: *Skloňovanie podstatných mien v slovenčine pomocou neurónovej siete*, Diplomová práca, Univerzita Komenského, Fakulta matematiky, fyziky a informatiky, Katedra informatiky, diplomový vedúci: Ing. Igor Farkaš PhD., Bratislava, 2005, 44 strán

Skloňovanie podstatných mien (substantív) v slovenčine je zatiaľ dosť nepreskúmaná časť jazyka. Aspoň čo sa týka akvizície skloňovania u detí. Mňa budú zaujímať rýdzo slovenské slová, ktoré sa učia používať deti do štyroch rokov.

K porozumeniu mojej diplomovej práce treba mať základné vedomosti z oblastí neurónových sietí aj tvaroslovia podstatných mien. Ešte si treba uvedomiť, že deti sa neučia pomocou písaného slova, ale zvukov. Preto sa stručne pozrieme aj na základné zvukové elementy jazyka, fonémy.

Skúsím nájsť paralelu v učení neurónovej siete a malého dieťaťa, prípadne predpovedať možné správania a problémy. Taktiež prekúmam, ktoré syntaktické a sémantické informácie sú potrebné k úspešnému naučeniu a zovšeobecňovaniu skloňovania substantív.

Kľúčové slová: dopredná neurónová sieť, podstatné mená, skloňovanie, morfológia, fonémy

Obsah

1	Úvod	5
1.1	Motivácia	5
1.2	Obsahy jednotlivých kapitol	6
2	Základné pojmy	7
2.1	Fonémy	7
2.1.1	Delenie foném a ich kódovanie	8
2.1.2	Fonetický zápis slova	10
2.2	Podstatné mená	11
2.3	Neurónové siete	14
2.3.1	Perceptrón	15
2.3.2	Dopredné neurónové siete a ich učenie	16
3	Spôsobý kódovania charakteristík	19
3.1	Reáločíselné vs. binárne kódovanie hodnôt charakteristík . . .	19
4	Reáločíselné kódovanie charakteristík	21
4.1	Fonetický zápis slova	21
4.2	Kódovanie foném	22
4.3	Pripravenie vstupu pre neurónovú sieť	23
4.4	Spracovanie výstupu siete	24

4.5	Parametre siete	24
5	Binárne kódovanie charakteristík	25
5.1	Fonetický zápis slova	25
5.2	Kódovanie foném	27
5.3	Pripravenie vstupu a spracovanie výstupu siete	29
6	Tvorba trénovacej a testovacej množiny	30
7	Výsledky pokusov s reálnočíselným kódovaním	32
8	Výsledky pokusov s binárnym kódovaním kódovaním	34
8.1	Sieť učená aj s pomôckami	34
8.2	Sieť učená aj s pomôckami, pri testovaní zadávané chybné pomôcky	34
8.2.1	Chybná životnosť	36
8.2.2	Chybná životnosť aj rod	37
8.3	Sieť učená bez pomôcok	40
9	Záver	42
10	Prílohy	43
10.1	Vzory	43
10.2	Množina slov	48
10.3	Tabuľky	50

Kapitola 1

Úvod

1.1 Motivácia

Vďaka svojej morfolologickej bohatosti je slovenčina veľmi dobrým kandidátom na testovanie schopnosti neurónovej siete zvládnuť úlohu skloňovania substantív. Drvivá väčšina doterajších prác z oblasti modelovania prirodzeného jazyka bola zameraná na angličtinu, konkrétne v morfológii na slovesá[1] (učenie minulého času pravidelných a nepravidelných slovík), pretože morfológia substantív je v podstate veľmi jednoduchá. V slovenčine (ako aj ostatných slovanských jazykoch) je morfológická stavba substantíva dôležitým zdrojom informácie pre syntaktickú a sémantickú analýzu vety (zatiaľ čo spomínaná angličtina sa pritom skôr opiera o poradie slov v vete). Modelovanie morfológie možno z matematického hľadiska chápať ako hľadanie zobrazenia medzi vstupom (podstatné meno v prvom páde) a výstupom (skloňovaný tvar), čo je typ úlohy typicky vhodný pre neurónovú sieť. Súčasne, modelovanie kognitívnej úlohy pomocou neurónovej siete predstavuje pokus vysvetliť takýto proces u ľudského jedinca. Preto je pri modelovaní zaujímavé analyzovať, aké typy chýb neurónová sieť robí.

1.2 Obsahy jednotlivých kapitol

V **druhej kapitole** si vysvetlíme základné pojmy a pozrieme sa bližšie na morfológiu podstatných mien aj na štruktúru a algoritmus učenia dopredných neurónových sietí.

Tretia kapitola obsahuje základný prehľad o možnostiach kódovania foném, porovnanie výhod a nevýhod reálnočíselného a binárneho kódovania charakteristík foném slova.

V **štvrtej kapitole** sa už priamo zaoberám kódovaním vstupov a interpretovaním výstupov pri použití reálnočíselného kódovania.

V **piatej kapitole** popisujem ako som modifikoval spôsob kódovania vstupov a interpretovania výstupov pri prechode na binárne kódovanie. Taktiež popisujem vzrast výpočtovej zložitosti pri tomto kódovaní.

V **šiestej kapitole** popisujem tvorbu trénovacej a testovacej množiny zo zoznamu slov až k vyskloňovaným tvarom zapísaným pomocou foném

Siedma kapitola obsahuje krátky rozbor výsledkov dosiahnutých pri modeli používajúcom reálnočíselné kódovanie charakteristík foném.

V **ôsmej kapitole** sa pozrieme na výsledky pokusov robených s binárnym kódovaním foném. Pozrieme sa na efekt použitia pomôcok pri učení, aj na chyby, ktoré neurónová sieť robí.

Kapitola 2

Základné pojmy

V tejto kapitole vysvetlím základné jazykovedecké pojmy, s ktorými sa v mojej práci môžeme stretnúť. Je nutné pochopiť tvaroslovie(morfológiu) podstatných mien aj zvukovú stránku jazyka. Pozrieme sa na základné gramatické kategórie substantív, ktoré by sa mohli objaviť ako vstupy neurónovej siete, prípadne mohli byť potrebné pri vysvetľovaní výsledkov. Taktiež v krátkosti prezentujem neurónové siete ako celok a podrobnejšie model neurónovej siete, ktorý v implementácii diplomovej práce budem používať.

2.1 Fonémy

Každý určite poznáme slovenskú abecedu, malé deti ju nepoznajú, ale aj tak pomerne veľmi dobre skloňujú. Vedia rozoznávať zvuky, nevedia rozoznávať grafémy(písmená). Preto ani ja nebudem na reprezentáciu substantív používať *grafémy* ale *fonémy*. Pre úspech implementácie je dôležité používať vhodnú množinu foném, a k tomu fonémy ešte aj správne reprezentovať ako vstup neurónovej siete. Prezentujem delenie foném aj možnosti kódovania foném pre počítač. Fonémy sú abeceda zvukov, ktoré vníma ľudské ucho

pri počúvaní hovorenej reči. Sú to najmenšie zvukové jednotky, na ktoré má ešte zmysel počuté slovo deliť. Reprezentujú základné elementy, z ktorých vieme po zvukovej stránke vyskladať každé slovo. Triedenie hlások podľa ich zvukových a artikulačných vlastností, ako aj informácia o vzťahoch medzi nimi (napríklad párové spoluhlásky, hlásky párové podľa dĺžky, mäkkosti) sú potrebné a využívajú sa pri analýze a syntéze slov. Ich vzťahy a vlastnosti pomáhajú pri určovaní vlastností podstatných mien, aj pri následnej tvorbe jednotlivých pádových tvarov. Príkladom je skracovanie a predlžovanie vokálov (samohlások), aby sme dostali rytmickému zákonu. Vzťahy medzi fonémami nám umožňujú vybrať správnu hlásku. V prípade rytmického krátenia takto vyberieme podľa dĺžky komplementárnu samohlásku. Reprezentácia foném by tieto vzťahy mala čo najlepšie zachovávať a pritom nebyť príliš komplexná.

Keď sa pokúsim kódovať celé slová pomocou foném, tak zase potrebujem dosiahnuť, aby podobne znejúce slová mali aj podobný zápis. Ešte je dôležité zohľadniť, ktoré časti slova sú pre mňa viac dôležité, či začiatok, stred alebo koniec slova a podobne. Pritom by ale mala byť reprezentácia čo najkratšia.

2.1.1 Delenie foném a ich kódovanie

Základné delenie písmen v Slovenčine je na *samohlásky*, *spoluhlásky* a *dvojhlásky*. Zaujímavé sú ešte slabikotvorné spoluhlásky. Medzi slabikotvorné spoluhlásky patria *l, r, ľ* a *ř*, ktoré sa dajú v slove použiť na pozícii samohlásky (napr. *vlk*, *stĺp*, vieme vysloviť aj *vak* alebo *stap*, ale nevieme vysloviť *vmp* alebo *stfp*, lebo *m* a *f* nie sú slabikotvorné). Klasifikáciu foném, z ktorej som vychádzal si môžete pozrieť v knihe [3], na moje použitie ale nie je celkom vyhovujúca. Neskôr v texte sa bližšie pozrieme na tabuľku foném, ktorú naozaj používam (tabuľka 10.3), a ktorá spĺňa požadované vlastnosti.

V tabuľke foném môžeme nájsť charakteristiky foném, pomocou ktorých by som potreboval jednoznačne a neredundantne určiť každú fonému. Uvediem niektoré kódovania foném, ktoré by sa dali použiť:

- Fonému kódujem poradovým číslom v zozname foném
- Lokálnym (*Onehot*) spôsobom, môžem požiť binárny vektor, všetky hodnoty budú 0, okrem hodnoty ktorá zodpovedá poradovému číslu fonémy v zozname.
- Kódujem pomocou charakteristiky, na každú charakteristiku použijem k bitov, kde k je minimálne a pritom platí $2^k \geq$ počet hodnôt charakteristiky.
- Kódujem pomocou charakteristiky, na každú charakteristiku použijem jednu reálnu hodnotu z intervalu $\langle 0, 1 \rangle$, prípadne $\langle -1, 1 \rangle$. Každéj hodnote tejto charakteristiky potom pripadá subinterval tohto intervalu.

Prvý a druhý spôsob nie príliš vyhovujú, lebo vôbec neuvažujú zvukovú stránku foném ani ich prípadný vzťah.

Tretí spôsob pri implementácii aj tak funguje tak, že na každý bit budem používať jedno float číslo (prečo, vychádza z neurónových sietí). Pri tomto spôsobe ešte treba používať Grayovo kódovanie 2.1.1 namiesto štandardného binárneho kódovania. Problém binárneho kódovania je, že napr. 3 a 4 sú susedné čísla a majú podstatne iný kód (011 vs 100).

K tretiemu a štvrtému spôsobu treba správne zvoliť charakteristiky aj správne numerické ohodnotenie jednotlivých hodnôt charakteristík, lebo to podstatným spôsobom môže ovplyvniť úspešnosť programu.

číslo	binárne kódovanie	Grayovo kódovanie
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

Tabuľka 2.1: Grayovo kódovanie

2.1.2 Fonetický zápis slova

Spôsob ako urobíme fonetický zápis slova je porovnateľne dôležitý ako rozumný spôsob zápisu jednotlivých foném. Od zápisu chceme, aby umožnil neurónovej sieti odhaliť, čo najviac vzťahov medzi zvukovou stránkou jednotlivých slov. Pozriem sa krátko, aké štruktúry boli v minulosti navrhnuté pre reprezentáciu zvukovej stránky slova.

Prvá metóda (Plunkett a Marchman [4]) kódovania foném slova berie slovo ako *jednoduchý reťazec* foném. Každú fonému kódovali pomocou šiestich binárnych jednotiek. Čiže na trojfonémové slovo bolo treba 18 jednotiek. Problém reprezentácie bol, že predpokladali slabiky typu CVC, VCC, CCV. Čiže sa nedali reprezentovať dvojhlásky.

Druhá metóda (Mäkiläinen [2]) používa 5 reálnočíselných hodnôt na reprezentáciu fonologických vlastností fonémy. Problémom je reprezentovanie slov s rozdielnou dĺžkou, hoci sú zvukovo veľmi podobné, napr. *spot* a *pot*. Budú mať úplne rozdielnú fonetickú reprezentáciu spôsobenú hláskou *s* na

začiatku slova *spot*. *P* v slove *pot* je kódované na pozície 1-5 a v *spot* na pozície 6-10.

MacWhinney a Leinbach ([1]) navrhli *rubrikovú reprezentáciu* (slot-based representation). Fonológiu slova kódujeme do šablóny s pevnou množinou slotov. Napríklad na reprezentáciu trojslabikového anglického slovesa použili šablónu CCCVVCCC VVCCC VVCCC, kde **C** je spoluhláska (*consonant*) a **V** je samohláska (*vocal*). Týmto spôsobom čiastočne zmizne problém s rezdielnou fonetickou dĺžkou slov. *Spot* zakódujeme **spCoVtCC** a *pot* zakódujeme **pCCoVtCC**. Tieto vzorky sa už podobajú viac, ako v predošlom prípade. Miernou modifikáciou môžeme dosiahnuť ešte lepší efekt: **CspVoCCt** a *pot* zakódujeme **CCpVoCCt**. Pôvodná šablóna je zarovnávaná vľavo, modifikovaná je zarovnávaná vpravo. Takto sa šablóny zhodujú na ešte viac pozíciách. Keď používame pevnú dĺžku šablóny je otázne, či budeme slovo zarovnávať vpravo alebo vľavo. Keď v slovenčine dáme pred slovo predponu a budeme zarovnávať vľavo, tak sa kódy slova s predponou a bez nej budú veľmi líšiť. Podobný problém nastane pri prípone a kódovaní vpravo. V princípe ale platí, že ak je pre problém dôležitejší koniec slova, tak by sme mali použiť zarovnávanie vpravo. Ak je dôležitý koreň slova a pri tom sa mení dĺžka prípony (predpony), tak máme problém pri reprezentácii vpravo (vľavo).

2.2 Podstatné mená

Podstatné mená (*substantíva*) patria medzi ohybné slovné druhy v slovenskom jazyku. Patria tu ešte prídavné mená, zámená, číslovky, slovesá a príslovky. Najzložitejšiu morfológiu z týchto slovných druhov majú práve podstatné mená. V iných jazykoch to vôbec nemusí byť tak. V anglickom jazyku je napr. morfológia podstatných mien oveľa jednoduchšia (pre každé

slovo iba tvar pre množné a jednotné číslo). Morfológia slovies je už porovnateľná. Zaujímavé pre nás je skúmať väčší problém, v angličtine to sú slovesá, u nás podstatné mená.

V tejto časti textu sa bližšie pozrieme na gramatické kategórie podstatných mien, ich morfológiu a ešte aj vzťah k ich sémantike. Porozumenie základným vzťahom a problémom pri skloňovaní podstatných mien nám umožní navrhnúť vhodný model neurónovej siete. Na druhej strane z tohto modelu môžeme získať nový zaujímavý náhľad na skúmanú tématiku. V slovenskom jazyku dokonca aj prvý náhľad tohto typu.

Takže ku kategóriám:

- **rod: ženský, stredný, mužský.** U väčšiny slov ho vieme bez rozmýšľania korektne určiť, ale ako to vlastne robíme? Aké informácie potrebujeme o slove, aby sme správne určili rod? Všimnime si slová *kosť* a *hosť*. Prvé má ženský rod, druhé mužský. Aký rod by malo slovo *most*? Keď sa pokúsime určiť rod len z toho ako slovo znie, prideme na to, že sa to nedá. Nestačí nám lexikálny prístup, podľa zvukovej podobnosti slov sa nedá určiť ich rod. Gramatickú kategóriu rod sa jedinec musí pravdepodobne naučiť priamo priradovať k jednotlivým slovám ¹.

- **číslo: jednotné, množné.**

- **násobné, pomnožné, normálne.** Túto kategóriu tiež nevieme určiť zo základného tvaru, podobne ako rod a životnosť.

Pri pomnožných podstatných menách používame aj pre jednotné číslo tvary množného čísla. Príkladom pre pomnožné substantíva sú napr.

¹Moja domnienka. Možno je, že sa najprv naučíme skloňovať a potom už máme ľahké určiť rod podľa pádových koncoviek.

nohavice, nožnice. Ešte si všimneme ďalšiu zaujímavosť. V spojení *päť krabíc* tvar *krabíc* nie je nominatív množného čísla, ale genitív množného. Nominatív množného je *krabice*(povieme *dve krabice*).

Násobné zase používajú jednotné číslo na pomenovanie viac predmetov. Ako príklad pre násobné uvediem *lístie*. Množné číslo neexistuje.

V nasledujúcej tabuľke si môžeme všimnúť dve normálne, jedno pomnožné a jedno násobné substantívum. V zátvorkách sú napísané pády, ktoré boli využité². Aby bola tabuľka korektná, tak budem predpokladať, že slovo *lístie* použijem, až keď vidím kopy aspoň *päť listov* (Treba si uvedomiť, že *lístie* a *listy* sú naozaj rozdielne substantíva).

jeden objekt	dva objekty	päť objektov	dvojo objektov
mesto(NS)	mestá(NP)	miest(GP)	miest(GP)
chlap(NS)	chlapi(NP)	chlapov(GP)	chlapov(GP)
nohavice(NP)	–	nohavíc(GP)	nohavíc(GP)
–	–	lístie(NS)	–

Tabuľka 2.2: Špeciálne číslo: násobné, pomnožné, normálne

- **pád:** Poznáme 6 pádov jednotného a 6 pádov množného čísla. Niekedy sa určoval aj vokatív, ale ten ani ja už používať nebudem.
- **vzor:** Množina vybraných slov sú slová z jazyka, ktoré spôsobom svojho skloňovania pokrývajú množinu všetkých skloňovaní substantív. Žiadne dva vzory by sa nemali skloňovať rovnako. Dve slová sa skloňujú rovnako, ak vo všetkých pádoch majú rovnaké pádové prípony a taktiež majú pri tých istých pádoch rovnaké alternácie koreňa.

²NS znamená nominatív singuláru, GP je genitív plurálu

Čiže chceme, aby každé podstatné meno malo párové podstatné meno v množine vzorov, ktoré sa skloňuje rovnako. Keď vezmem takúto striktnú definíciu vybraných slov, tak ten štandard(4 vzory pre každý rod), čo každý poznáme, ju nespĺňa. Je tam priveľa výnimiek. Mám množinu slov vybranú z knihy *Sapfo* [3], ktorá by mala byť kompletne pokrývajúca celú množinu podstatných mien. Keď mám podstatné meno a jeden z týchto vzorov, tak viem s istotou toto podstatné meno správne vyskloňovať(podľa tohto vzoru).

- **životnosť: životné, neživotné.** Životné delíme na zvieracie a osobné. Táto kategória vyplýva priamo z významu slova a nie z jeho zápisu. Používame ju na správne určenie vzoru mužských podstatných mien³. Z toho vyplýva, že aj túto kategóriu asi treba vedieť určiť predtým ako skúsime skloňovať podstatné meno, podobne ako rod.
- **konkrétne, abstraktné.** Sémantická kategória. Potrebujeme vedieť význam slova, aby sme správne určili. Neviem, či má relevantný vzťah k skloňovaniu. Abstraktné slová sú z princípu neživotné.

2.3 Neurónové siete

V tejto kapitole si predstavíme základný model neurónových sietí, dopredné neurónové siete, ako z teoretického, tak aj aplikačného hľadiska. Oboznámime sa so základnými pojmami týkajúcimi sa prvkov, architektúry a učenia neurónových sietí, ktoré boli pôvodne inšpirované neurobiológiou mozgu.

³čiže v princípe pri skloňovaní

2.3.1 Perceptrón

Perceptrón je základný element umelej neurónovej siete tak isto, ako je neurón základným elementom organických neurónových sietí⁴. Existuje viac modelov perceptrónov, odlišujú sa spôsobom, akým sa funkciou približujú biologickým neurónom. V princípe sa ale za väčšie priblíženie platí väčšou výpočtovou náročnosťou a pri tom neplatí, že viac je nutne aj lepšie. Popíšem hlavne funkcionalitu štandardného perceptrónu používaného v dopredných neurónových sieťach.

Perceptrón prijíma vstupné signály zo vstupu $\bar{x} = (x_1, x_2, \dots, x_n)$; $x_i \in \mathbb{R}$ a pomocou synaptických váh tvoriacich váhový vektor $\bar{w} = (w_1, w_2, \dots, w_n)$; $w_i \in \mathbb{R}$ im priraduje dôležitosť aj excitačnú alebo inhibičnú funkciu. Zložky váhového vektora sú ľubovoľné reálne čísla. Záporná váha je inhibičná, kladná excitačná, ak je váha blízka 0, daný vstup výstup neovplyvňuje. Vstupný vektor sa nazýva vzor alebo obrazec (angl. pattern). Zložky vstupného vektora môžu nadobúdať reálne alebo binárne hodnoty. Perceptróny pracujúce s reálnymi číslami umožňujú aproximáciu zložitejších spojitých funkcií. Podstatná je ešte hodnota prahu excitácie θ . Zo vstupného, váhového vektora a prahu potom vieme vyrátať výstup perceptrónu.

$$o = f(\text{net}) = f(\bar{x} \cdot \bar{w}) = f\left(\sum_{j=1}^{n+1} x_j w_j\right) = f\left(\sum_{j=1}^{n+1} x_j w_j - \theta\right) \quad (2.1)$$

Funkcia f sa nazýva **aktivačná funkcia**.

Štandardne sa používa $w_{n+1} = \theta$ a $x_{n+1} = -1$.

$$f(\text{net}) = \begin{cases} 1 & \text{net} \geq 0 \iff \sum_{j=1}^n > \theta \\ -1 & \text{net} \geq 0 \iff \sum_{j=1}^n > \theta \end{cases} \quad (2.2)$$

⁴V texte budem používať termín neurón pre simulované neuróny(perceptróny)

Túto bipolárnu binárnu aktivačnú funkciu 2.2 v roku zaviedol Rosenblatt[5]. Hranica medzi vstupmi, ktoré dávajú výstup 1 a -1 je daná rovnicou $\sum_{j=1}^{n+1} x_j w_j = \theta$. Ak by mal vstup dimenziu 2, tak táto rovnica by určovala deliacu priamku, ak by mal dimenziu 3, tak by zase určovala deliacu rovinu. Všeobecne platí, že táto rovnica určuje nejakú nadrovinu, čiže jedným binárnym perceptrónom sa dajú riešiť iba lineárne separovateľné problémy. Unipolárna 2.3 aj bipolárna 2.4 reálno-číselná aktivačná funkcia sa používa vo väčšine štandardných neurónových sietí. Pribudol nám tu ďalší parameter, **strmosť sigmoidy** λ . Zvyčajne sa používa s hodnotou 1 až 2.

$$f(net) = \frac{1}{1 + e^{-\lambda net}} \quad (2.3)$$

$$f(net) = \frac{2}{1 + e^{-\lambda net}} - 1 \quad (2.4)$$

Pre diskkrétne aj spojité neuróny existuje algoritmus učenia, pomocou ktorého vieme perceptrón naučiť riešiť lineárne separovateľné problémy.

2.3.2 Dopredné neurónové siete a ich učenie

Väčšina reálnych problémov má nelineárny charakter. To znamená, že sa nedajú riešiť pomocou jedného perceptrónu. Je asi prirodzené skúsiť získať väčšiu silu tým, že skúsime použiť nejakým spôsobom spoluprácu viacerých neurónov. Takouto spoluprácou sú práve neurónové siete. V prírode máme ukážku, že neurónové siete sú naozaj použiteľné.

Zase máme veľa možností ako neurónovú sieť vytvoriť. Najprv sa rozhodneme, ktorý typ neurónov použijeme na jej zosrojenie⁵. Potom ešte musíme zvoliť architektúru siete a v poslednej rade aj algoritmus učenia, ktorý sa podstatne mení pri rôznych architektúrach aj typoch perceptrónu.

⁵okrem perceptrónov to môžu byť napríklad RBF neuróny

Jeden z možných modelov je aj dopredná neurónova sieť so skrytými neurónmi. Dopredné neurónové siete sa vyznačujú tým, že v nich existujú iba dopredné spojenia medzi neurónmi. Každý neurón jednej vrstvy vysiela signály na každý neurón nasledujúcej vrstvy. Spojenia do predchádzajúcej vrstvy ani v rámci jednej vrstvy neexistujú. Každý neurón najnižšej vstupnej vrstvy vidí všetky vstupné hodnoty. Výstupom neurónovej siete sú výstupy neurónov najvyššej výstupnej vrstvy. Programátor musí kódovať vzory ako reálnocíselné vstupy siete a podobne zase interpretovať výstupy. Práve pomocou tohto modelu Rumelhart, Hinton a Williams [6] ukázali, že schopnosti neurónových sietí nie sú ohraničené len na riešenie lineárne separovateľných problémov.

Zaviedli pravidlo tréningu nazvané metóda spätného šírenia chýb (angl. error backpropagation). Vezmime, že naša sieť má niekoľko vrstiev neurónov. Vektor \bar{o} definujeme ako vektor výstupov najvyššej vrstvy.

$$o_k^p = f(\text{net}_k^p) \quad (2.5)$$

Chceme aby sa naučila mapovať vstupné vektory $(\bar{y}^1, \bar{y}^2 \dots \bar{y}^P)$ na požadované výstupné $(\bar{d}^1, \bar{d}^2 \dots \bar{d}^P)$, kde $\bar{y}^i = (y_1^i, y_2^i \dots y_n^i); y_j^i \in \mathbb{R}$ a podobne $\bar{d}^i = (d_1^i, d_2^i \dots d_K^i); d_j^i \in \mathbb{R}$.⁶ Definujeme účelovú funkciu E^p nazývanú **chybová funkcia** 2.6, ktorú chceme minimalizovať. Podstatné na tejto funkcii je, že naozaj vyjadruje vzdialenosť výstupu \bar{o}_k^p od želaného \bar{d}_k^p a k tomu ju môžeme rozumne derivovať podľa o_k^p , čo neskôr budeme potrebovať.

$$E^p = \frac{1}{2} \cdot \sum_{k=1}^K (d_k^p - o_k^p)^2 \quad (2.6)$$

Na učenie takejto neurónovej sa používa metóda záporného gradientu (metóda najstrmšieho spádu). Pri riešení všetkých problémov zložitejších

⁶Obor hodnôt d_k^i by sme mali ešte obmedziť na hodnoty z intervalu $\langle -1,1 \rangle$ alebo $\langle 0,1 \rangle$ podľa toho, či používame bipolárne perceptróny alebo nie.

ako lineárna funkcia sa táto metóda musí používať iteračne(inkrementálne), čiže aj pri neurónovej sieti. Odvodím najprv zmenu j-tej váhy k-teho neurónu 2.7 výstupnej vrstvy.

$$\Delta w_{kj} = -\alpha \frac{\partial E^p}{\partial w_{kj}} = -\alpha \frac{\partial E^p}{\partial net_k} \frac{\partial net_k}{\partial w_{kj}} = \alpha \delta_{o_k} y_j \quad (2.7)$$

$$\frac{\partial net_k}{\partial w_{kj}} = \frac{\partial \sum_{j=1}^{n+1} y_j w_{kj}}{\partial w_{kj}} = y_j \quad (2.8)$$

$$\delta_{o_k} = -\frac{\partial E}{\partial net_k} = -\frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial net_k} = -\frac{\partial \left[\frac{1}{2} \cdot \sum_{k=1}^K (d_k - o_k)^2 \right]}{\partial o_k} \frac{\partial o_k}{\partial net_k} = (d_k - o_k) \mathbf{f}'_k \quad (2.9)$$

Zderivovaním a dosadením dostaneme pre unipolárnu sigmoidu

$$\mathbf{f}'_k = \frac{\partial o_k}{\partial net_k} = o_k(1 - o_k) \quad (2.10)$$

$$\mathbf{f}'_k = \frac{\partial o_k}{\partial net_k} = (1/2)(1 - o_k^2) \quad (2.11)$$

Teraz odvodím najprv zmenu j-tej váhy k-teho neurónu 2.12 výstupnej vrstvy. Vstupy tejto vrstvy budem označovať \bar{x} a váhy perceptrónov tejto vrstvy v . Hodnoty δ vrstvy nad touto označím Ψ .

$$\Delta v_{ji} = -\alpha \frac{\partial E^p}{\partial v_{ji}} = -\alpha \frac{\partial E^p}{\partial net_j} \frac{\partial net_j}{\partial v_{ji}} = \alpha \delta_{y_j} x_i \quad (2.12)$$

Odvođenje δ_{y_j} je trochu zdĺhavé, výsledný vzťah 2.13, čo dostaneme, závisí od hodnôt rátaného neurónu a niektorých hodnou vo vrstve nad ním.

$$\delta_{y_j} = \left[\sum_{k=1}^K \Psi_k w_{kj} \right] \mathbf{f}'_j \quad (2.13)$$

Kapitola 3

Spôsoby kódovania charakteristík

3.1 Reálnočíselné vs. binárne kódovanie hodnôt charakteristík

Pri použití reálnočíselného kódovania namiesto binárneho sa zväčša dosahujú menšie rozmery neurónovej siete. Keď máme napríklad 6 hodnotovú charakteristiku, tak nám stačí jeden neurón z intervalom výstupných hodnôt rozdeleným na 6 podintervalov, kde každý z intervalov zastupuje jednu hodnotu. Pri binárnom kódovaní týchto hodnôt potrebujeme 3 neuróny. Na $5(2^2 + 1)$ až $8(2^3)$ hodnôt potrebujeme 3 bity. Skúsme si spraviť obraz ako sa toto zväčšenie počtu neurónov môže prejaviť na časovej néročnosti algoritmu učenia. Predpokladajme, že na binárne kódovanie vstupov aj výstupov potrebujeme rádovo dvakrát toľko neurónov. Na počtoch neurónov v skrytých vrstvách sa to nutne musí prejaviť tiež, predpokladajme, že aj tam zvýšime počet neurónov dvakrát. Z rovnice 2.1 pre výpočet výstupu jedného neurónu hocikde v sieti máme vzrast výpočtovej zložitosti dvojnásobný, pritom je neurónov dvakrát toľko. Čiže pri zráťavaní výstupu neurónovej siete máme

štyrikrát väčšiu výpočtovú zložitosť. Rovnaký vzrast zložitosti nastáva pri použití algoritmu spätného šírenia chýb.

Na druhej strane neurónová sieť s binárnym kódovaním má väčšiu diskriminatívnosť na vstupoch a výstupoch. Pri reálnočíselnej neurónovej už veľmi malé odchylky spôsobujú zmenu hodnoty charakteristiky kódovanej daným neurónom. Problémom sú hlavne hodnoty v strede oboru výstupných hodnôt neurónu, lebo tam je sigmoida najstrmšia. A z toho vyplýva potenciálny problém pri zovšeobecňovaní, ak už nie priamo pri učení.

Kapitola 4

Reálnočíselné kódovanie charakteristík

4.1 Fonetický zápis slova

V tejto časti sa kúsok vrátim k úvodu, kde sme preberali spôsoby kódovania slova. Potenciálne najlepší spôsob kódovania navrhli MacWhinney a Leinbach[1] pomocou šablón typu *CCCVVCCCVV*. Prevezmem tento spôsob a upravím ho pre potreby práce so slovenčinou. Slovenské slovo neobsahuje viac ako tri samohlásky za sebou. Nevie, či existuje slovo v detskej slovnej zásobe, čo obsahuje 3 samohlásky za sebou. Pri spoluhláskach je to kúsok zložitejšie, lebo ešte musíme brať do úvahy, že existujú aj spoluhlásky slabikotvorné. Slovenské slovo neobsahuje potom viac ako 4 *neslabikotvorné* spoluhlásky priamo za sebou. Čiže pre slovenčinu je možné použiť šablóny tohoto typu, akurát tam musím pridať jednu spoluhlásku. Slabikotvorné spoluhlásky budem nejak mapovať na pozície samohlások. Keď si všimneme pádové koncovky, tak viditeľne viac ich končí samohláskou, preto som schému zvolil tak, aby končila samohláskami (vokál V). Všimnime si ešte raz, aké

možnosti zarovňavania slova (napr. puškou) do šablóny máme:

	c	c	c	c	v	v	v	c	c	c	c	v	v	v	c	c	c	c	v	v	v
1.	p	-	-	-	u	-	-	š	k	-	-	o	u	-	-	-	-	-	-	-	-
2.	-	-	-	p	-	-	u	-	-	š	k	-	o	u	-	-	-	-	-	-	-
3.	-	-	-	-	-	-	-	-	-	-	p	-	-	u	-	-	š	k	-	o	u
4.	-	-	-	-	-	-	-	p	-	-	-	u	-	-	š	k	-	-	o	u	-

Tabuľka 4.1: Možnosti zarovňavania do šablóny

Keďže skloňovanie sa prejavuje na podstatných menách skôr na konci, prípadne v strede slova, tak som predbežne zvolil 3. alebo 4. prípad zarovňavania. Finálne som sa potom rozhodol pre spôsob 3. Ľahšie sa implementuje a k tomu dodatočne umocňuje sústreďenie na koniec slova.

4.2 Kódovanie foném

Teraz sa už priamo pozrime na môj návrh kódovania. Keď si pozrieme tabuľku 10.3, tak vidíme, že samohlásky a dvojhlásky majú 4 charakteristiky (atribúty), ale posledný atribút v princípe nepoužívajú, lebo je stále nastavený na 0. Ďalej som sa snažil aby dvojhlásky mali čo najviac atribútov z tej samohlásky, ktorú predlžujú (i[^] a predlžuje a). Dĺžku dvojhlásky som nastavil na dlhá.

Podobne môžeme vidieť, že spoluhlásky používajú 4 charakteristiky s tým, že ich používajú naozaj všetky. Slabikotvorným spoluhláskam som dal atribúty, čo najviac podobné zodpovedajúcim spoluhláskam (k r= patrí spoluhláska r). Do atribútu 0 som ale dal dĺžku namiesto miesta artikulácie, lebo dĺžka je na tej pozícii pri samohláskach. Keďže slabikotvorné spoluhlásky

mapujem do šablóny na pozície samohlások, tak aj každá V pozícia v šablóne musí umožniť zakódovať 4 charakteristiky, hoci ináč by stačili len 3. Na každú pozíciu v šablóne ešte ukladám typ. Na pozícii V môže byť typ samohláska, dvojláska, slabikotvorná spoluhláska a žiadna (šablóna $CCCCVVV$ je štandardne "deravá"). Na pozícii C môže byť typ spoluhláska a žiadna. Na každú fonému v šablóne potom používam 5 hodnôt, čiže 5 neurónov.

4.3 Pripravenie vstupu pre neurónovú sieť

V tejto časti sa bližšie pozriem na postup, akým som sa dostal zo zápisu slova pomocou foném k vstupu pre neurónovú sieť. Na vstup neurónovej siete dávam nominatív podstatného mena a požadovaný pád. Pridávam tam prípadne gramatické kategórie, ktoré by mohli pomôcť neurónovej sieti pri vygenerovaní správneho výstupného tvaru. Požadované výstupy sa tvoria rovnako s tým, že tam ide iba holé slovo a žiadny pád ani nejaké pomôcky.

Ako som už povedal, zvolil som zarovnávanie foném slova v šablóne doprava. Fonetický zápis slova obsahuje iba zápisy foném zodpovedajúcim stringom. Najprv vytvorím pole foném, ktoré obsahuje fonémy aj s ich typmi a atribútmi¹. Potom vytvorím šablónu($CCCVVVV$ vektor) a následne aj reálnočíselný vektor², ktorý je súčasťou vstupu neurónovej siete. Prázdnu fonému kódujem tak, že jej nastavím typ na žiadny a ostatné atribúty na 0. Všetky fonémy kódujem tak ako vidieť v tabuľke 10.3.

Pád kódujem one hot spôsobom a dávam na prvých 12 pozícií vstupu. Pomôcky rod a životnosť ukladám tesne za pád reálnočíselným kódovaním (vstupy 13 a 14).

¹opakované volanie FonemaHladac::fonemaFromCharP

²Slovo::updateVectors v phoneme.cpp

4.4 Spracovanie výstupu siete

Mám 5 reálnočíselných hodnôt pre každú fonému. Za výstupnú fonému volím tú, čo má najmenšiu odchylku svojich hodnôt od hodnôt na výstupe. Žiadna fonéma má prioritu³.

4.5 Parametre siete

Pokúšal som sa experimentovať s počtom neurónov, strmostou sigmoidy, aj počtom vrstiev. Dve skryté vrstvy sa prejavili negatívne, sieť sa pomalšie učila. Najväčšiu úspešnosť⁴ na tréningovej množine som získaval pri $\lambda = 2$.

³Nastaviť hranicu, kedy tam fonéma ešte je a kedy už nie, bol dosť problém, hlavne keď na tom neuróne bol kódovaný aj typ fonémy

⁴Rozumné hodnoty $\lambda \in \langle 1.5, 2 \rangle$

Kapitola 5

Binárne kódovanie charakteristík

Pôvodne som implementoval model s reálnočíselným kódovaním, mal ale nespokojivé výsledky pri zovšeobecňovaní. Urobil som viac modifikácií, ktoré by sa spätne dali použiť aj na pôvodnom modeli. V jednotlivých sekciách tejto kapitoly popíšem zmeny, ktoré som urobil, aj dôvody prečo boli potrebné. Zmeny zahŕňajú zmenu šablóny aj zarovnávanie foném do šablóny, a aj prechod na binárne kódovanie a s tým späté zmeny pri tvorbe konečného reálnočíselného vektora. Aby som znížil množstvo potrebných neurónov zavediem flexibilný počet bitov, pomocou ktorých sa budú fonémy kódovať.

5.1 Fonetický zápis slova

V šablóne ešte priamo nie je vidieť, že sme prešli na binárne kódovanie. Modifikovaná šablóna by sa dala použiť aj pri reálnočíselnom kódovaní. Až prechod zo šablóny na jednoduchý reálnočíselný vektor nám ukáže, aký nárast počtu neurónov priniesol prechod na binárne kódovanie. Popíšem problémy pôvodnej mnou navrhutej šablóny:

- slabikotvorné spoluhlásky a samohlásky používajú tie isté políčka vstup-

ného/výstupného vektora neurónky na úplne rôzne atribúty.

- samohlásky majú jeden atribút zbytočný
- zbytočne na vstupe/výstupe kódujem typ fonémy, hoci je určený pozíciou v šablóne
- problém so zarovnávaním dvoch tvarov jedného slova, nominatív a vyskloňovaný tvar majú zápis posunutý o jednu až dve slabiky¹

Navrhnem nový typ šablóny tvaru *CCCCSVV*. Do šablóny som pridal políčka typu S (slabikotvorná spoluhláska/sonór). Znížil som počet samohlások v jednej slabike šablóny, lebo v aktuálnej testovacej ani trérovacej množine nemám viac ako dve zasebou idúce samohlásky. Slabikotvorné pozície sa budú mapovať na túto pozíciu, týmto spôsobom ušetrím aj jeden nepoužívaný atribút pri samohláskach. Typ fonémy už budem kódovať len pri samohláskach/dvojháskach. Pri spoluhláskach a slabikotvorných spoluhláskach už budem kódovať len prítomnosť fonémy alebo prázdne políčko. Ešte podrobnejšie vysvetlím štvrtý problém:

	3c	c	v	v	v	3c	c	v	v	v	3c	c	v	v	v	3c	c	v	v	v
1.	-	-	-	-	-	-	-	-	-	-	-	d'	-	-	ie	-	t'	-	-	a
2.	-	d'	-	-	ie	-	t'	-	-	a	-	t	-	-	a	-	m	-	-	i

Tabuľka 5.1: Problém pôvodného zarovňovania vpravo

Riešením je zarovnávať tvary jedného podstatného mena doľava. To znamená, že sa stále snažím mať fonémy, čo najviac vpravo, a pritom sa postarám o to, aby tvary jedného slova mali prvú fonému na tej istej pozícii v šablóne.

¹slabikou nazvem sedmicu *CCCCVVV* prípadne osmicu *CCCCSVVV* v novej šablóne

	2c	c	2v	v	3c	c	2v	v	3c	c	2v	v	3c	c	2v	v	3c	c	2v	v
1.	-	-	-	-	-	ď	-	ie	-	ť	-	a	-	-	-	-	-	-	-	-
2.	-	-	-	-	-	ď	-	ie	-	ť	-	a	-	t	-	a	-	m	-	i
3.	-	-	-	-	-	d	-	o	-	m	-	-	-	-	-	-	-	-	-	-
4.	-	-	-	-	-	d	-	o	-	m	-	a	-	m	-	i	-	-	-	-
5.	-	d	-	á	-	t	-	u	-	m	-	-	-	-	-	-	-	-	-	-
6.	-	d	-	á	-	t	-	u	-	m	-	a	-	m	-	i	-	-	-	-

Tabuľka 5.2: zarovnávanie do šablóny, modifikácie pre binárne kódovanie

Nové zarovnávanie bude fungovať tak, že najprv zarovnam štandardne doprava, potom posuniem nominatív doľava o dve slabiky a ostatné tvary zarovnam tak, aby mali prvú fonému na tej istej pozícii ako nominatív. V novej šablóne tam budú ešte pozície typu S. Pri takomto prístupe sa nachádzajú pádové prípony na posledných dvoch slabikách šablóny. Niekedy do prvej z posledných dvoch slabík presahuje koreň, ak náhodou nastalo rozšírenie koreňa slova o jednu slabiku (dieťa \Rightarrow dieťatami).

5.2 Kódovanie foném

Keď si pozrieme tabuľku 10.3, tak vidíme, že samohlásky a dvojhlásky majú 3 relevantné charakteristiky (atribúty), ktoré nám umožňujú medzi nimi rozlišovať (predpokladám, že vieme, či je to samohláska alebo dvojhláska).

Spoluhlásky majú relevantné atribúty 4.

Slabikotvorné spoluhlásky majú 2 atribúty, ktoré nám medzi nimi umožňujú rozlišovať.

Teraz zrátam koľko bitov potrebujem na jednotlivé typy s tým, že musím aj zakódovať, či tam fonéma je, alebo tam je len prázdne miesto.

Samohlásky a dvojhlásky:

- typ: samohláska, dvojhláska, žiadna; 3 hodnoty \implies 2 bity.
- dĺžka: dlhá, krátka; 2 hodnoty, potrebujem 1 bit.
- rezonančné priestory: zadné, stredné, predné; 3 hodnoty \implies 2 bity.
- rezonančné priestory: vysoké, stredové, nízke; 3 hodnoty \implies 2 bity.

Na samohlásku potom potrebujem $7(2+1+2+2)$ bitov.

Spoluhlásky:

- typ: prítomná, žiadna; 2 hodnoty \implies 1 bit.
- miesto artikulácie: pernoperné, pernozubné, predďasnové, zadďasnové, ďasnovopodnebné, tvrdopodnebné, mäkkopodnebné, hrtanové; 7 hodnôt \implies bity.
- podľa účasti hlasu: zvučné, znelé, neznelé; 3 hodnoty \implies 2 bity.
- podľa sluchového dojmu: explozívne, afrikáty, frikatívy, sonóry, likvidy, vibranty; používam iba prvé 4 hodnoty \implies 2 bity.
- pomocný atribút: 3 hodnoty \implies 2 bity.

Na spoluhlásku potom potrebujem $10(1+3+2+2+2)$ bitov.

Slabikotvorné:

- typ: prítomná, žiadna; 2 hodnoty \implies 1 bit.
- dĺžka: dlhá, krátka; 2 hodnoty \implies 1 bit.
- pomocný atribút: 2 hodnoty \implies 1 bity.

Na slabikotvornú spoluhlásku potom potrebujem $3(1+1+1)$ bit.

Jedna slabika pôvodne spotrebovala **35** hodnôt. Každá slabika obsahovala štyri spoluhlásky a tri samohlásky, a každá z nich používala 5 reálnočíselných hodnôt.

Slabika pri binárnom kódovaní už spotrebuje **57** hodnôt. Mám štyri spoluhlásky, dve samohlásky a jednu slabikotvornú samohlásku. Koľko bitov každá spotrebuje, je uvedené vyššie.² Keď zachovám počet slabík, tak výpočet neurónky bude môžeme približne odhadnúť na $2.5(= (\frac{57}{35})^2)$ násobok pôvodného času. Pre modifikované zarovnávanie som musel ešte zväčšiť veľkosť šablóny o jednu slabiku, z päť na šesť. Z toho vyplýva ďalšie zvýšenie náročnosti $1.4 \times^3$.

5.3 Pripravenie vstupu a spracovanie výstupu siete

Všetky zmeny, čo som urobil v kóde, súvisia s prechodom na binárne kódovanie a modifikáciou zarovňovania. Prešiel som z bipolárneho na unipolárne kódovanie, aby sa mi lepšie reprezentovali 0 a 1 v Grayovom zápise čísel.

Neuróny výstupnej vrstvy dávajú výstupy v intervale od 0 po 1. Najprv každú hodnotu zaokrúhlim na jednu z hraničných hodnôt tohto intervalu. Potom mám kódovania foném len pomocou hodnôt 0 a 1. Za výstupnú fonému volím fonému, ktorá má najbližší binárny zápis k zápisu, čo mám na výstupe. Bit, ktorý určuje žiadna vs prítomná fonéma, má prioritu.

²Z toho potom máme: $4 \cdot 10 + 1 \cdot 3 + 2 \cdot 7 = 57$

³ $(6/5)^2$

Kapitola 6

Tvorba trénovacej a testovacej množiny

V tejto kapitole chcem popísať ako som generoval dáta potrebné na tvorbu trénovacích a testovacích množín. Potreboval som množinu vyskloňovaných slov zapísaných pomocou foném. Od množiny používaných slov sa ďalej vyžadovalo, aby zodpovedala množine slov, s ktorými sa môže stretnúť malé dieťa.

V tejto časti sa chcem ešte raz poďakovať doc. PaedDr. Jane Kesselovej za množinu slov aj s početnosťami výskytov ako sa dieťa so slovami z tejto množiny môže stretnúť (V prílohách, množina slov je podmnožina týchto slov, ktorú som ja použil). Pri trénovaní siete slová s väčšou početnosťou dávam viac krát na vstup. Pre príliš veľkú dĺžku alebo viacslovnosť som z tejto množiny slová vylučoval. Taktiež som nepoužil žiadne násobné a pomnožné substantíva. Násobné podstatné mená by sa dali testovať na sieťach, ktoré som natrénoval, akurát by nemali tvary množného čísla. Pomnožné podstatné mená by už boli problém, lebo pomnožné podstatné mená nemajú tvar nominatívu jednotného čísla, ktorý ja dávam na vstupy siete.

Z tejto množiny som vygeneroval množinu slov, kde už každé slovo má všetkých dvanásť pádových tvarov. Stále všetko zapisujem pomocou písmen. Toto vyskloňovanie podstatných mien som sa pokúsil zautomatizovať¹. Podľa zadania niekoľkých pádových tvarov slovám priraďujem vzor. Pomocou slova a vzoru už viem slovo kompletne vyskloňovať. Použil som čiastočne modifikované vzory z knihy [3], pozrieť si ich môžete v prílohách. Zmenil som označenie alternácií, a pre nejednoznačnosť alternácie predĺžovanie somhlásky v koreni slova, som pridal ďalšie vzory².

Ďalej som potreboval vygenerované slová písané pomocou grafém prepísať na fonetický zápis. Pomoc som dostal z Ústavu informatiky SAV, spracovanie reči, ktorý vlastní program, ktorý dokáže práve tento prepis vykonať. Takto vygenerované slová už bolo treba iba otagovať vzorom a životnosťou a početnosťou výskytov slova. Takto vygenerované vstupné súbory pre program obsahujú v každom riadku početnosť, rod, životnosť a potom všetkých dvanásť tvarov jedného substantíva.

¹program Vzorovac, GUI pomocou gtk (Pred prípadným použitím treba do systému nainštalovať potrebné knižnice)

²Ešte som aj opravil nejaké chyby

Kapitola 7

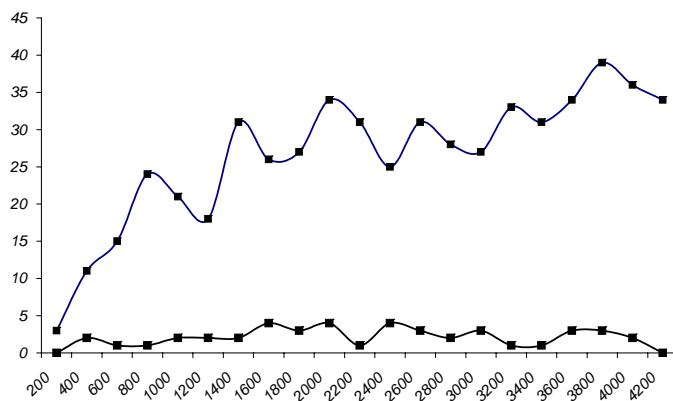
Výsledky pokusov s reálnočíselným kódovaním

V tejto kapitole v krátkosti prezentujem výsledky, ktoré dosiahol tento model. Úspešnosť mal ale dosť malú v porovnaní s modelmi s binárnym kódovaním, preto nebudem dôslednejšie rozoberať chyby, ktorých sa model dopúšťal. Ani nebudem popisovať žiadne pokusy robené bez pomôcok. Grafy ukazujú úspešnosti na tvaroch slov, na x-ovej osi je epocha. Na každom grafe sú dve krivky: úspešnosť na trénovacej a na testovacej množine.

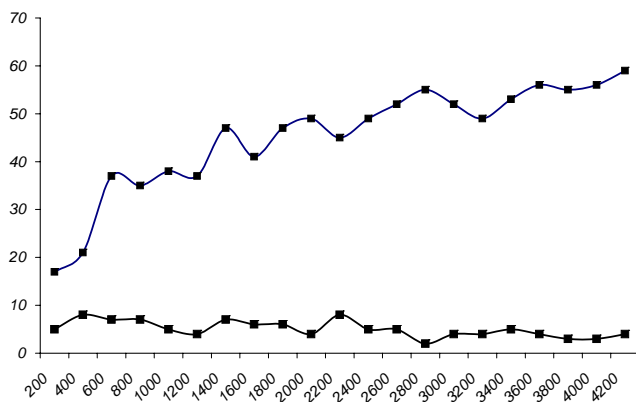
V grafoch 7.1,7.2 vidíme naozaj nízku úspešnosť na testovacej množine, nestabilitu na trénovacej. Taktiež sa zdá, že preučenie nastáva vo veľmi skorých fázach učenia¹.

¹Úspešnosť na testovacej osciluje alebo klesá od veľmi skorých epoch, aj keď úspešnosť na trénovacej množine ešte narastá

KAPITOLA 7. VÝSLEDKY POKUSOV S REÁLNOČÍSELNÝM KÓDOVANÍM



Graf 7.1: Úspěšnost tvarov pri učení pádov množného čísla



Graf 7.2: Úspěšnost tvarov pri učení pádov jednotného čísla

Kapitola 8

Výsledky pokusov s binárnym kódovaním kódovaním

Pri týchto testoch som sa sústredil na výskum, či pomôcky rod a životnosť napomáhajú úspešnosti učenia siete. Pokusy sú robené na neurónovej sieti s počtami 356 vstupných hodnôt, 684 neurónov na skrytej vrstve, 342 neurónov na výstupe. Rýchlosť učenia, momentum a strmosť sigmoidy sa po niekoľkých pokusoch ustálili na hodnotách $\alpha = 0.01$, $momentum = 0.2$, $\lambda = 1$.

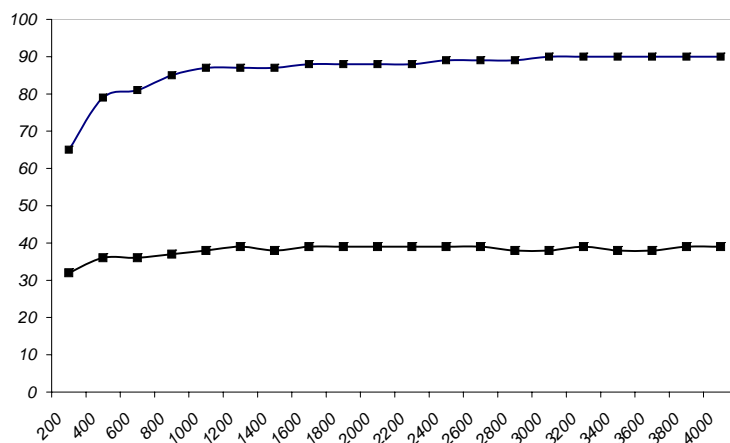
8.1 Sieť učená aj s pomôckami

Grafy úspešnosti na tvaroch (graf 8.1) a na slovách (graf 8.2) .

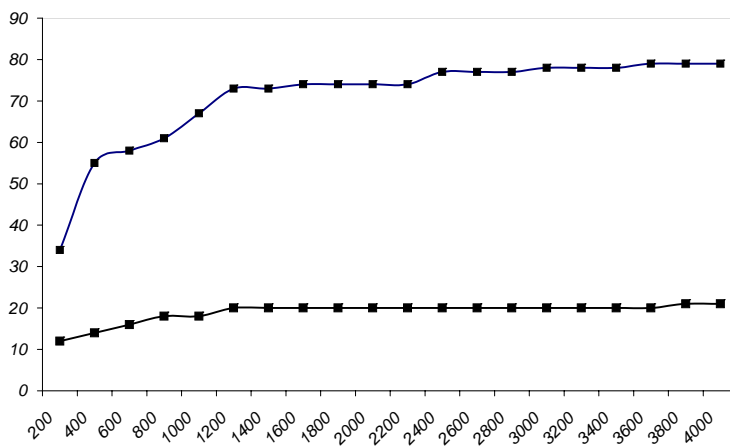
8.2 Sieť učená aj s pomôckami, pri testovaní zadávané chybné pomôcky

Sieť učím so štandardými vstupmi aj s pomôckami. Po skončení učenia vstupy upravím nižšie uvedeným spôsobom a sledujem, čo to spôsobí. Rod

KAPITOLA 8. VÝSLEDKY POKUSOV S BINÁRNÝM KÓDOVANÍM KÓDOVANÍM



Graf 8.1: Úspešnosť tvarov pri pádoch jednotného čísla



Graf 8.2: Úspešnosť slov pri pádoch jednotného čísla

som prerobil tak, že mužský a stredný rod kódujem ako ženský, a ženský ako mužský. U životnosti kódujem osobné a zvieracie ako neživotné, a neživotné ako osobné. Grafy úspešnosti pádov sú v 3000. epoche.

Pri pádoch (graf 8.4) môžeme vidieť pôvodný pokus zobrazený bielymi stĺpcami. Môžeme sa pozrieť aj na prípad, keď je pokazená len životnosť (graf 8.3). Výsledky, čo budú nasledovať hovoria, že ak sa učíme skloňovať s informáciou a rode (životnosti) a naučíme sa to pri skloňovaní používať, tak náhla strata schopnosti určiť rod (životnosť by mala podobné následky). Tiež sa ukáže, ako je morfológia jednotlivých pádov závislá na rode (životnosti). Sieť učená bez pomôcok nám potom ukáže, nakoľko vie sieť sama extrahovať z fonetického zápisu informácie zodpovedajúce určeniu rodu (životnosti).

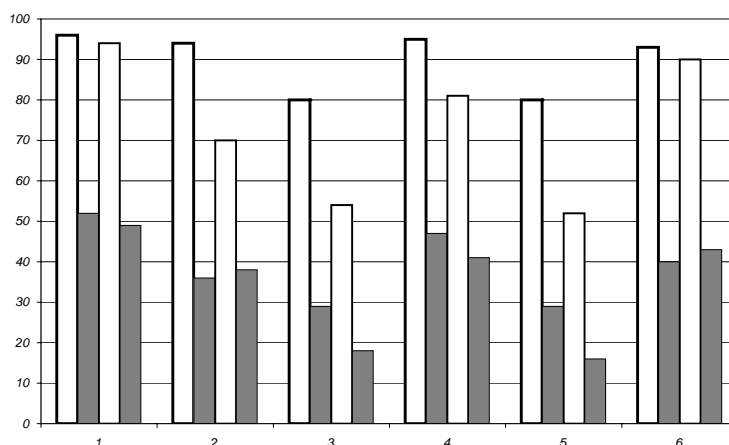
8.2.1 Chybná životnosť

Keď sa pozrieme na graf 8.3, zistíme ako zadávanie nekoreknej životnosti mení úspešnosť siete ¹:

- Nominatív, akuzatív a inštrumentál nie sú pády závislé na životnosti.
- Datív a lokál su na životnosť veľmi citlivé: problém sa dá nájsť napr. v mužskom rode chlapovi, hrdinovi vs dubu, stroju v datíve a pre lokál zase chlapovi, hrdinovi vs dube, stroji², podobne by nám mali vyjsť chyby aj v akuzatíve a naozaj v grafe sú, aj keď nie tak výrazné.

¹Toto je moja interpretácia výsledkov

²keby sme povedali, že dub je napr. osobné podstatné meno, tak by prebral koncovky zo vzoru chlap a z toho by vyšli viditeľné chyby



Graf 8.3: Porovnanie úspešnosti pádov pri zmenej životnosti (z ľava tréno-
vacia, testovacia, zmenená tréno-
vacia, zmenená testovacia)

8.2.2 Chybná životnosť aj rod

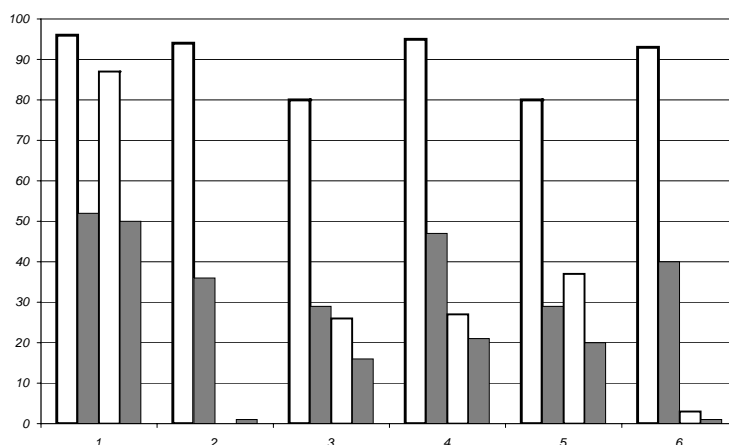
Keď sa pozrieme na graf 8.4, zistíme ako dávanie nekorektnej životnosti a rodu mení úspešnosť siete. Čo z toho môžeme vyvodiť:

- jedine tvorba tvaru pre nominatív nie je závislá na rode³
- pre ostatné pády je informácia o rode dosť podstatná, veľmi neúspešné sú genitív a inštrumentál⁴, aj čitateľ si jednoducho môže prejsť cez vzory jednotlivých rodov a uvedomiť si, aké skloňovanie v týchto pádoch majú⁵.

³je to aj celkom prirodzené, keď ho dávame na vstup

⁴aj testovacia aj tréno-
vacia množina majú úspešnosť blízku 0

⁵Treba si všímať aj alternácie koreňa, nie len koncovky



Graf 8.4: Porovnanie úspešnosti pádov(z ľava trénovacia, testovacia, zmenená trénovacia, zmenená testovacia)

V tabuľke 8.1 som vybral niekoľko príkladov, ako sa zmenilo skloňovanie slov, keď som pokazil rod a životnosť. Treba si uvedomiť, že to je zápis pomocou foném(tabuľka 10.3). Pri inštrumentáli slova med vidíme markantný prechod na skloňovanie ženského rodu. Genitívu tiež vyšiel mimo pádových koncoviek mužského rodu. O ostatných pádoch sa nedá s istotou povedať, že sa skloňujú ako v ženskom rode, ale vyhovujú aj ženskému rodu. Podobné zmeny môžeme nájsť aj pri slove *slovo*. Ono akurát prechádza zo stredného do ženského rodu. Slová *ryba*, *maruška* sú príkladom modifikácie pádových tvarov preto, že som začal dávať na vstup mužský rod namiesto ženský.

KAPITOLA 8. VÝSLEDKY POKUSOV S BINÁRNÝM KÓDOVANÍM
KÓDOVANÍM

Pád	Tvar s korektnými pomôčkami	Tvar so zmenenými pomôčkami
1	m e t	m e t
2	m e d u	m e t i
3	m e d u	m e t i
4	m e t	m e t
5	m e J \ i	m e t i
6	m e d o m	m e t o u _ ^
1	s l o v o	s l o v o
2	w l o v a	s l o v e
3	s l o v u	s l o v e
4	s l o v o	s l o v u :
5	s l o v e	s l o v e
6	w l o v o m	s l o v o u _ ^
1	r i b a	r i b a
2	r i b i	r i b a
3	r i b e	r i b e
4	r i b u	r i b a
5	r i b e	r i b e
6	r i b o u _ ^	r i b o m
1	m a r u s k a	m a r u S k a
2	m a r u s k i	m a r u S k a
3	m a r u s k e	m a r u S k e
4	m a r u s k u	m a r u S k a
5	m a r u s k e	m a r u S k e i
6	m a r u s k o u _ ^	m a r u S k o F

Tabuľka 8.1: Chyby, ktoré vznikli pri premiešaní pomôcok

Pád	Tvar s pomôckami	Tvar bez pomôcok
1	n o s	n o s
2	n o s i	n o s i
3	n o s u	n o s u
4	n o s	n o s
5	n o s e	n o s e
6	n o s o m	n o s o u _ ^

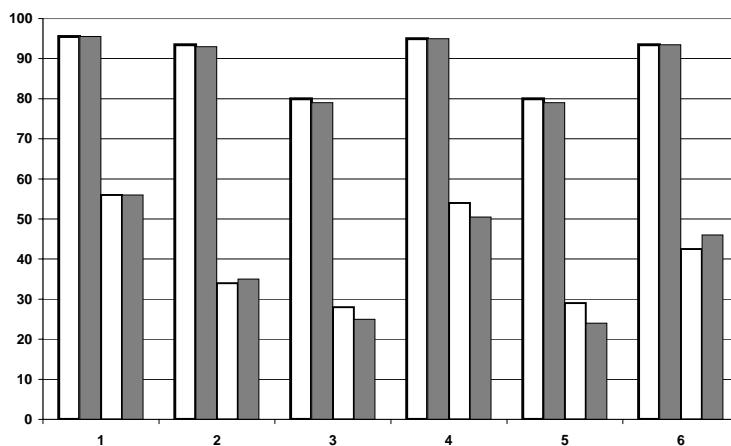
Tabuľka 8.2: Učenie bez pomôcok

8.3 Sieť učená bez pomôcok

Výsledky tohto pokusu nám ukážu nakoľko vie sieť extrahovať informácie porovnateľné s informáciou o rode (životnosti) len z fonetického zápisu slova. Môže sa jej celkom úspešne dariť uhádnuť tieto gramatické kategórie. Takže sa pozrime, ako sa darí neurónke bez pomôcok. Na grafe 8.5 vidíme, že sú v porovnateľne úspešné. Znamená to, že pomôcky sú nepotrebné? Pokusy vyššie v texte nám ukázali, že rod aj životnosť na vstupe sieť využije. Vysvetlenie sa dá nájsť, keď sa pozrieme, akých chýb sa dopúšťa sieť pri trénovacej a hlavne pri testovacej množine. V princípe sa to ukazuje už aj tým, ako je sieť úspešná pri nominatíve. Sieť pri generovaní tvarov robí veľa chýb nesúvisiacich so skloňovaním. Dost' veľa chýb je v koreňoch slov, kde sa ešte skloňovanie neprejavuje alternáciami. Sieť vo veľkej miere nevie len okopírovať fonémy zo začiatku slov. Kebyže sa nám podarí odstrániť tieto chyby, potom by sme možno uvideli nejaký viditeľnejší rozdiel medzi neurónovou sieťou učenou s a bez pomôcok.

Môžeme si všimnúť zaujímavý efekt v tabuľke 8.3, kde vidíme v prvých piatich pádoch skloňovanie slova nos podľa mužského rodu a potom zrazu v

inštrumentáli podľa ženského. Podobný efekt som si všimol aj pri počiatočných fázach pri učení s pomôckami. V jednom prípade sa napr. stalo, že v dvestej epoche som mal v akuzatíve pri slove *barón* tvar *barónu* a štyristej epoche už tvar *baróna* ⁶.



Graf 8.5: Porovnanie úspešnosti pádov, na striedačku pád siete učenej s pomôckami a bez

⁶s tým, že nominatív, genitív a inštrumenál boli v poriadku v oboch prípadoch a v zvyšných dvoch pádoch boli porovnateľné patvary

Kapitola 9

Záver

V práci sme sa pokúsili simulovať učenie sa skloňovania podstatných mien neurónovou sieťou. Použili sme dva podstatne odlišné prístupy kódovania vstupov a výstupov s tým, že reálnočíselné kódovanie foném sa ukázalo ako nevhodné. Pri binárnom kódovaní sme sa pozreli na niekoľko aspektov skloňovania substantív. Pri pokusoch bez a so zadaním rodu a životnosti sme videli, ako sa to prejavuje na výstupovh siete. Všimli sme si, že sieť ktorá nedostáva na vstupe rod, má tendenciu miešať pri jednom slove skloňovania, ktoré sa ukazujú ináč v rozdielnych rodoch. Pokusy so zle zadaným rodom a životnosťou nám ukázali, ako sloňovanie niektorých pádov úzko súvisí práve s týmito gramatickými kategóriami. Pomerne nízka úspešnosť pri zovšeobecňovaní na testovacej množine by sa dala zdôvodniť obrovskými rozmermi siete. V porovnaní s učením sa skloňovania u ľudských jedincov by mal obstať aj pomer v úspešnostiach v jednotlivých pádov, tiež by bolo zaujímavé pozrieť sa na pomer rýchlosť učenia ľudského jedinca a siete aspoň v počiatočných fázach učenia.

Kapitola 10

Prílohy

10.1 Vzory

chlap 0,a,ovi,a,ovi,om,i,ov,om,ov,och,mi;

peržan 0,a,ovi,a,ovi,om,ia,ov,om,ov,och,mi;

dedo o,a,ovi,a,ovi,om,ovia,ov,om,ov,och,ami;

feláh 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi;

botto o,u,ovi,u,ovi,om,ovia,ov,om,ov,och,ami;

host' 0,a,ovi,a,ovi,om,ia,í,om,í,och,ami;

boh 0,a,u,a,u,om,ovia,ov,om,ov,och,mi;

mím 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,ami;

lovec 0,a,ovi,a,ovi,om,i,ov,om,ov,och,ami >-1sg;

vodník 0,a,ovi,a,ovi,om,i,ov,om,ov,och,mi -k+1pl +c+1pl;

černoch 0,a,ovi,a,ovi,om,i,ov,om,ov,och,mi -ch+1pl +s+1pl;

hrdina a,u,ovi,u,ovi,om,ovia,ov,om,ov,och,ami;

otec 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,ami >-1sg;

turek 0,a,ovi,a,ovi,om,i,ov,om,ov,och,ami -k+1pl +c+1pl >-1sg;

paris 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi -s-1sg +d-1sg;

aias 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi -s-1sg +n-1sg;
 nero 0,a,ovi,a,ovi,om,i,ov,om,ov,och,mi +n-1sg ó-1sg;
 anakreón 0,a,ovi,a,ovi,om,i,ov,om,ov,och,mi +t-1sg -1sg;
 achilles 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi -es-1sg;
 syzifos 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi -os-1sg;
 anaxagoras 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi -as-1sg;
 génius 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,ami -us-1sg;
 pontifex 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,mi -ex-1sg +ik-1sg;
 noe 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,ami +m-1sg;
 futbolista a,u,ovi,u,ovi,om,i,ov,om,ov,och,ami;
 bandita a,u,ovi,u,ovi,om,i,ov,om,ov,och,mi;
 hume 0,a,ovi,a,ovi,om,ovia,ov,om,ov,och,ami;
 vrchný ý,ého,ému,ého,om,ým,í,ých,ým,ých,ých,ými;
 cestujúci i,e,ho,emu,e,ho,om,im,i,ich,im,ich,ich,imi;
 dub 0,a,u,0,e,om,y,ov,om,y,och,mi;
 mak 0,u,u,0,u,om,y,ov,om,y,och,mi;
 med 0,u,u,0,e,om,y,ov,om,y,och,mi;
 svietnik 0,a,u,0,u,om,y,ov,om,y,och,mi;
 papier 0,a,u,0,i,om,e,ov,om,e,och,mi;
 model 0,u,u,0,i,om,y,ov,om,y,och,mi;
 strom 0,u,u,0,e,om,y,ov,om,y,och,ami;
 meter 0,a,u,0,i,om,e,ov,om,e,och,ami >-14sg;
 hotel 0,a,u,0,i,om,y,ov,om,y,och,mi;
 rytmus 0,u,u,0,e,om,y,ov,om,y,och,ami -us-14sg;
 nónius 0,a,u,0,u,om,y,ov,om,y,och,ami -us-14sg;
 most 0,a,u,0,e,om,y,ov,om,y,och,ami;
 zlomok 0,u,u,0,u,om,y,ov,om,y,och,ami >-14sg;

šperk 0,u,u,0,u,om,y,ov,om,y,och,ami;
 ohníček 0,a,u,0,u,om,y,ov,om,y,och,ami >-14sg;
 pojem 0,u,u,0,e,om,y,ov,om,y,och,ami >-14sg;
 papier a,u,u,0,i,om,e,ov,om,e,och,mi;
 chrbát 0,a,u,0,e,om,y,ov,om,y,och,ami >-14sg;
 chlieb 0,a,u,0,e,om,y,ov,om,y,och,mi }-14sg;
 mráz 0,u,u,0,e,om,y,ov,om,y,och,mi }-14sg;
 stôl 0,a,u,0,e,om,y,ov,om,y,och,mi }-14sg;
 vietor 0,a,u,0,e,om,y,ov,om,y,och,ami >-14sg }-14sg;
 stroj 0,a,u,0,i,om,e,ov,om,e,och,mi;
 jeleň 0,a,ovi,a,ovi,om,e,ov,om,e,och,mi;
 mravec 0,a,ovi,a,ovi,om,e,ov,om,e,och,ami >-1sg;
 čaj 0,u,u,0,i,om,e,ov,om,e,och,mi;
 veniec 0,a,u,0,i,om,e,ov,om,e,och,ami >-14sg;
 dážď 0,a,u,0,i,om,e,ov,om,e,och,ami }-14sg;
 had 0,a,ovi,a,ovi,om,y,ov,om,y,och,mi;
 orol 0,a,ovi,a,ovi,om,y,ov,om,y,och,ami >-1sg;
 peniaz 0,u,u,0,i,om,e,í,om,e,och,mi }+2pl;
 plášť 0,a,u,0,i,om,e,ov,om,e,och,ami;
 deň 0,a,u,0,i,om,i,í,om,i,och,ami >-14sg;
 kôň 0,a,ovi,a,ovi,om,e,í,om,e,och,mi }-1sg;
 kuli 0,ho,mu,ho,m,m,ovia,ov,om,ov,och,ami;
 atašé 0,0,0,0,0,0,0,0,0,0,0;
 žena a,y,e,u,e,ou,y,0,ám,y,ách,ami {ie+2pl;
 zore a,y,e,u,e,ou,e,0,ám,y,ách,ami {ô+2pl;
 matka a,y,e,u,e,ou,y,0,ám,y,ách,ami <ie+2pl;
 vojna a,y,e,u,e,ou,y,0,ám,y,ách,ami <e+2pl;

jamka a,y,e,u,e,ou,y,0,ám,y,ách,ami <ô+2pl;
perla a,y,e,u,e,ou,y,0,ám,y,ách,ami <á+2pl;
kráska a,y,e,u,e,ou,y,0,am,y,ach,ami <o+2pl;
čarodejka a,y,e,u,e,ou,y,0,ám,y,ách,ami <o+2pl;
idea a,y,e,u,i,ou,y,í,ám,y,ách,ami;
dáma a,y,e,u,e,ou,y,0,am,y,ach,ami;
úloha a,y,e,u,e,ou,y,0,ám,y,ách,ami;
izis 0,y,e,u,e,ou,y,0,ám,y,ách,ami -s-1sg +d-1sg;
ceres 0,y,e,u,e,ou,y,0,ám,y,ách,ami -s-1sg +r-1sg;
juno o,y,e,u,e,ou,y,0,am,y,ach,ami +n-1sg;
demeter 0,y,e,u,e,ou,y,0,ám,y,ách,ami >-1sg-2pl {ie+2pl;
víchrica a,e,i,u,i,ou,e,0,iam,e,iach,ami;
ulica a,e,i,u,i,ou,e,0,iam,e,iach,ami {í+2pl;
funkcia a,e,i,u,i,ou,e,í,ám,e,ách,ami;
sudkyňa a,e,i,u,i,ou,e,0,ám,e,iach,ami {ý+2pl;
vládkyňa a,e,i,u,i,ou,e,0,ám,e,iach,ami;
čerešňa a,e,i,u,i,ou,e,0,iam,e,iach,ami <ie+2pl;
ruža a,e,i,u,i,ou,e,í,iam,e,iach,ami;
vôňa a,e,i,u,i,ou,e,í,am,e,ach,ami;
svieca a,e,i,u,i,ou,e,0,am,e,ach,ami;
dlaň 0,e,i,0,i,ou,e,í,iam,e,iach,ami;
mysel' 0,e,i,0,i,ou,e,í,iam,e,iach,ami >-14sg;
báseň 0,e,i,0,i,ou,e,í,am,e,ach,ami >-14sg;
koľaj 0,e,i,0,i,ou,e,í,am,e,ach,ami;
pani i,ej,ej,iu,ej,ou,ie,í,iam,ie,iach,iami;
kráľovná á,ej,ej,ú,ej,ou,é,0,ám,é,ách,ami <ie+2pl;
gazdiná á,ej,ej,ú,ej,ou,é,0,ám,é,ách,ami í+2pl;

ženská á,ej,ej,ú,ej,ou,é,ých,ým,é,ých,ými;
 mať 0,e,i,0,i,ou,e,í,iam,e,iach,ami +er-1sg;
 kosť 0,i,i,0,i,ou,i,í,iam,i,iach,ami;
 cirkev 0,i,i,0,i,ou,i,í,iam,i,ách,ami >-14sg;
 kader 0,e,i,0,i,ou,e,í,ám,e,ách,ami;
 labuť 0,i,i,0,i,ou,e,í,iam,e,iach,ami;
 sapfo 0,0,0,0,0,0,0,0,0,0,0;
 mesto o,a,u,o,e,om,á,0,ám,á,ách,ami ie+2pl;
 mestečko o,a,u,o,u,om,á,0,ám,á,ách,ami <ie+2pl;
 rádio o,a,u,o,u,om,á,í,ám,á,ách,ami;
 jedlo o,a,u,o,u,om,á,0,ám,á,ách,ami <ie+2pl;
 sklenárstvo o,a,u,o,e,om,a,0,am,a,ach,ami <ie+2pl;
 slnko o,a,u,o,u,om,á,0,ám,á,ách,ami Í+2pl;
 krídlo o,a,u,o,e,om,a,0,am,a,ach,ami <e+2pl;
 rúcho o,a,u,o,u,om,a,0,am,a,ach,ami;
 miesto o,a,u,o,e,om,a,0,am,a,ach,ami;
 lýtko o,a,u,o,u,om,a,0,am,a,ach,ami <o+2pl;
 gesto o,a,u,o,e,om,á,0,ám,á,ách,ami;
 capriccio io,a,u,o,u,om,á,í,ám,á,ách,ami;
 fórum 0,a,u,0,e,om,a,0,am,a,ach,ami -um-14sg;
 múzeum 0,a,u,0,u,om,á,í,ám,á,ách,ami -um-14sg;
 srdce e,a,u,e,i,om,ia,0,iam,ia,iach,ami ř+2pl;
 riečište e,a,u,e,i,om,ia,0,iam,ia,iach,ami;
 drieve e,a,u,e,i,om,ia,0,iam,ia,iach,ami <e+2pl;
 líce e,a,u,e,i,om,a,0,am,a,ach,ami;
 citoslovce e,a,u,e,i,om,ia,0,iam,ia,iach,ami <ie+2pl;
 more e,a,u,e,i,om,ia,í,iam,ia,iach,iami;

oje e,a,u,e,i,om,e,í,am,e,ach,ami;

vysvedčenie ie,ia,iu,ie,í,ím,ia,í,iam,ia,iach,iami;

dievča a,a,u,a,i,om,á,0,ám,á,ách,ami +ať+2356sg +at+123456pl;

mača a,a,u,a,i,om,á,0,ám,á,ách,ami +ať+2356sg +at+123456pl ia+2pl;

kura a,a,u,a,i,om,e,0,om,e,och,ami +ať+2356sg +enc+123456pl;

teľa a,a,u,a,i,om,e,0,om,e,och,ami +ať+2356sg +c+123456pl;

dieťa a,a,u,a,i,om,i,í,om,i,och,mi +ať+2356sg +123456pl;

cestovné é,ého,ému,é,om,ým,é,ých,ým,é,ých,ými;

10.2 Množina slov

144 x voda

141 x babka

138 x deň

120 x mamka

119 x mama, škola

116 x kráľ

99 x mamička, pani

93 x auto

88 x rozprávka

84 x chlapci

81 x rok

75 x detičky, Petra Lešková, ujo

72 x deti, les, ocko

66 x brat, noha, vlk

63 x koruna

60 x Červená Čiapočka, návšteva

58 x pódiun, princezná, zvieratká

51 x vec

50 x pes

49 x autobus, dievča, kamarát, korytnačka, oriešky,

42 x 12 slov

človek, dcéra, domček, kopec, lienka, Mišo, okno, otecko, princ, ruka, sobota,

39 x lístoček, prázdniny, stoličky

36 x chata, koláčik, mená, otázka, televízor, teta,

33 x 13 slov

dedko, dievčatko, film, hodina, hotel, Jana Libová, medvedík, miesto 1., Ňufko, pampúšik, panáčik, sestra, strom,

30 x 22 slov

cesta, čaj, dedo, dieťa, divadlo, dom, hosť, kamarátka, kocka, mesto, Nemeč, obed, oco, oko, otec, program, puding, raketa, Rus, stará, stôl, týždeň,

27 x 15 slov

decká, erotika, Gabika, kocúr, kola-kaio, Lenka, ovocie, postieľka, Prešov, sneh, Snehulienka, svadba, taška, trpaslíci, Zuzka,

24 x 26 slov

Amerika, číslo, dvere, finalistka, guma, chalúpka, chvíľa, Janko a Marienka, kniha, kolotoč, košieľka, koza, lietadlo, logopédia, Marienka, minúta, Mirko, pán, piatok, šaty, torta, tráva, trieda, ucho, ústa, vajíčka,

21 x 26 slov

bomba, bratranec, fixa, futbal, hlava, hokej, hračka, hrad, huby, koleso, kráľovstvo, list, noc, obchod, pavúk, pesnička, sesternica, schod, starček, sudička, svetlo, tanec, včela, zmrzlina, zrkadielko, zviera,

18 x 37 slov

bicykel, divák, indián, izba, jablká, jahoda, Janičko, knižka, koláč, koniček,

kostol, kráľovná, krv, lego, lyžička, macocha, morča, pero, pivo, pohárik, polícia, prasiatka, prechádzka, prst, Ruženka, samopal, skriňa, slnko, srdce, šašo, tank, telo, veža, Viedeň, zem, žena, ženích,

15 x 55 slov

angličtina, babička, bod, Branko, brucho, čarodejník, časť, čiernokňažník, dvor, hora, horár, kameň, Karolína, koľajnica, košeľa, košík, kúpaisko, lop-ta, lyža, mami, matka, mikrofón, miska, mrkva, narodeniny, oheň, papier, parkovisko, peniaze, persil, piesok, práca, ráno, riad, robota, rozhovor, stráž, stupeň, šíp, škôlka, tábor, Táňa, taštička, Tatry, trezor, večierok, Vinetu, vlas, vývrtka, záhrada, zajačik, značka, zub, žabka, žiaci,

12 x 94 slov

alergia, baba, barón, boby, Bratislava, bunker, čas, darček, diktát, duch, džíp, farba, farbička, fixka, gól, guľomet, hra, chlapček, chodníček, injekcia, jazero, Kamil, keks, krabička, kulturistka, kvet, lanovka, leto, lístok, Lysá, macík, Mária, Martin, Maruška, maska, med, medicimbal, meteorit, metla, Michal Jackson, miss, more, mostík, muž, nemocnica, nos, obecnstvo, ob-luda, obrázok, okienko, olympiáda, otvor, páska, plachta, riaditeľ, ružička, ryba, satelit, séria, skala, skrývačka, slovo, sranda, srdiečka, striekačka, svet, svetielko, šabl'a, škrečok, šmýkalka, špilka, štyrka, tanier, telefón, topánka, učenie, učiteľka, ujček, úlohy, veverička, víno, vitamín, Vlado, vychoška, vy-učovanie, zajac, zámok, zbojník, zemiak, zima, zlato, zošit, život, žuvačka,

10.3 Tabuľky

Kódovanie	1.reálnočíselné	2.reálnočíselné
Podľa prítomnosti tónovej zložky		
samohláska	0	1
dvojhláska	0.3	0.4
slabikotvorná	0.6	0.2
spoluhláska	0.9	-0.2
žiadna	-0.5	-1
Podľa dĺžky		
krátka	0	-1
dlhá	1	1
Podľa rezonančných priestorov		
predná	0	-1
stredná	0.5	0
zadná	1	1
vysoká	0	-1
stredová	0.5	0
nízka	1	1
Podľa miesta artikulácie		
pernoprené	0	-1
pernozubné	0.15	-0.7
pernoďasnové	0.3	-0.4
zadoďasnové	0.45	-0.1
ďasnovopodnebné	0.6	+0.1
tvrdopodnebné	0.75	0.4
mäkkopodnebné	0.9	0.7
hrtanové	1	1

Tabuľka 10.1: Charakteristiky foném, časť 1.

Kódovanie	1.reálnočíselné	2.reálnočíselné
Podľa účasti hlasu		
zvučná	0	-1
znelá	0.5	0
neznelá	1	1
Podľa sluchového vnemu		
explozív	0	-1
afrikát	0.2	-0.6
frikatív	0.4	-0.2
sonór	0.6	0.2
likvid	0.8	0.6
vibrant	1	1
Pomocný atribút		
0	0	0
1	0.5	0.5
2	1	1

Tabuľka 10.2: Charakteristiky foném, časť 2.

fonéma	typ	atribút 0	atribút 1	atribút 2	atribút 3
e	samohlaska	kratka	predna	stredova	0
i	samohlaska	kratka	predna	vysoka	0
a	samohlaska	kratka	stredna	nizka	0
o	samohlaska	kratka	zadna	stredova	0
u	samohlaska	kratka	zadna	vysoka	0
e:	samohlaska	dlha	predna	stredova	0
i:	samohlaska	dlha	predna	vysoka	0
a:	samohlaska	dlha	stredna	nizka	0
o:	samohlaska	dlha	zadna	stredova	0
u:	samohlaska	dlha	zadna	vysoka	0
i_ˆe	dvojhlasika	dlha	predna	stredova	0
i_ˆa	dvojhlasika	dlha	stredna	nizka	0
u_ˆo	dvojhlasika	dlha	zadna	stredova	0
i_ˆu	dvojhlasika	dlha	zadna	vysoka	0
r=	slabikotvorna	kratka	zvucna	sonor	0
l=	slabikotvorna	kratka	zvucna	sonor	1
r=:	slabikotvorna	dlha	zvucna	sonor	0
l=:	slabikotvorna	dlha	zvucna	sonor	1

Tabuľka 10.3: Rozdelenie foném časť 1.

fonéma	typ	atribút 0	atribút 1	atribút 2	atribút 3
m	spoluhlaska	pp	zvucna	sonor	0
b	spoluhlaska	pp	znela	exploziv	0
F	spoluhlaska	pp	znela	sonor	0
p	spoluhlaska	pp	neznela	exploziv	0
v	spoluhlaska	pz	znela	frikativ	0
w	spoluhlaska	pz	znela	frikativ	1
u_ ^	spoluhlaska	pz	znela	frikativ	2
f	spoluhlaska	pz	neznela	frikativ	0
n	spoluhlaska	pd	zvucna	sonor	0
N\	spoluhlaska	pd	zvucna	sonor	1
N	spoluhlaska	pd	zvucna	sonor	2
d	spoluhlaska	pd	znela	exploziv	0
z	spoluhlaska	pd	znela	frikativ	0
dz	spoluhlaska	pd	znela	afrikat	0
t	spoluhlaska	pd	neznela	exploziv	0
s	spoluhlaska	pd	neznela	frikativ	0
ts	spoluhlaska	pd	neznela	afrikat	0
r	spoluhlaska	zd	zvucna	sonor	0
l	spoluhlaska	zd	zvucna	sonor	1
Z	spoluhlaska	zd	znela	frikativ	0
dZ	spoluhlaska	zd	znela	afrikat	0
S	spoluhlaska	zd	neznela	frikativ	0
tS	spoluhlaska	zd	neznela	afrikat	0
J	spoluhlaska	dp	zvucna	sonor	0
L	spoluhlaska	dp	zvucna	sonor	1
J\	spoluhlaska	dp	znela	exploziv	0
c	spoluhlaska	dp	neznela	exploziv	0
j	spoluhlaska	tp	zvucna	sonor	0
i_ ^	spoluhlaska	tp	zvucna	sonor	1
k	spoluhlaska	mp	neznela	exploziv	0
g	spoluhlaska	mp	znela	exploziv	0
x	spoluhlaska	mp	neznela	frikativ	0
G	spoluhlaska	mp	neznela	frikativ	1
h	spoluhlaska	h	znela	frikativ	0

Tabuľka 10.4: Rozdelenie foném časť 2.

Fonéma	ortografický tvar	fonologický tvar
u:	múr	mu:r
i_ ^a	piatok	pi_ ^atok
i_ ^e	mier	mi_ ^er
u_ ^o	kôň	ku_ ^oJ
r=	vrch	vr=x
r=:	vŕba	vr=:ba
l=	vlk	vl=k
l=:	vĺča	vl=:tSa
L	ľad	Lad
F	amfiteáter	aFfitea:ter
n	rana	rana
N	Slovensko	sloveNško
N	banka	baNka
J	vaňa	vaJa
u_ ^	kov	kou_ ^
i_ ^	kraj	krai_ ^
c	Maťo	maco
J\	háďa	ha:J\ a
w	vdova	wdova
S	šek	Sek
Z	veža	veZa
x	chata	xata
ts	cena	tsena
tS	oči	otSi
dz	medza	medza
dZ	džungľa	dZuNgLa

Tabuľka 10.5: Fonémy s iným zápisom ako je zápis zodpovedajúcej grafémy

Literatúra

- [1] B. MacWhinney and J. Leinbach. Implementations are not conceptualizations:revising the verb learning model. *Cognition*, 29:121–157, 1991.
- [2] R. Miikkulainen. Dyslexic and category-specific aphasic impairments in a self-organizing feature map model of the lexicon. *Brain and Language*, 59:334–366, 1997.
- [3] Emil Páleš. *Sapfo - parafrázovač slovenčiny*. VEDA, 1994.
- [4] K. Plunkett and Marchman V. U-shaped learning and frequency effects in a multilayered perceptron:implicaions for child language acquisition. *Cognition*, 38:43–102, 1991.
- [5] F. Rosenblatt. The perceptron: a probabilistic model for information storage. *Psychological Review*, 65:386–408, 1958.
- [6] D. Rumelhart, G. Hinton, and R. Williams. *Learning internal representations by error propagation*. M.I.T.Press, MA, 1986.