



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

ZRE

ZPRACOVÁNÍ ŘEČOVÝCH SIGNÁLŮ

Projektová dokumentace

Viterbiho dekodér pro rozpoznávání řeči

Autor:

Petr Buchal
Martin Ivančo
Jan Kohút

Login:

xbucha02
xivanc03
xkohut08

3. Květen, 2019

1 Úvod

Projekt předmětu ZRE klade za cíl implementovat Viterbiho dekodér pro rozpoznávání řeči. K řešení tohoto problému jsme použili jazyk Python 2.

2 Rozpoznávací síť

Nejdříve jsme si navrhli rozpoznávací síť, viz obrázek 1. Jedno požadované slovo se může skládat z několika variant posloupností fonémů, v rozpoznávací síti se nachází model pro každou takovou variantu. Slova síť zpracovává paralelně, tedy v každém kroku se přesouvají tokeny do každého možného stavu a to v modelech pro všechna požadovaná slova. Síť poté v každém stavu vybere token, který má nejvyšší hodnotu. Naše síť neuvažuje ticho na začátku a na konci. Takové řešení nám přijde redundantní, protože síť může provést cyklení přes model ticha, který je souběžný s ostatními modely pro slova.

3 Token passing

Při inicializaci nastavíme tokenu v rozhodovacím uzlu hodnotu nula, ostatním tokenům je poté přidělena velká záporná hodnota. Při každém přechodu se do hodnoty tokenu přičítá logaritmus přechodové pravděpodobnosti, což je v našem modelu 0.5. Dále se v cílovém stavu během přechodu přičítá k hodnotě tokenu logaritmus likelihoodu. Token se během přechodu přeposílá do všech stavů, na které je jeho aktuální stav napojen. Pokud se po provedení přechodu objeví ve stavu více tokenů, nadále se počítá pouze s tím, který měl největší hodnotu. V naší síti se nachází tzv. rozhodovací uzel, ten v každém kroku vezme tokeny, které do něj doputovaly, vybere ten s největší hodnotou a přepošle ho dále, ostatní zahodí. O zápis cesty stavů, kterou token prošel se stará rozhodovací uzel.

4 Výstup dekodéru

a30000b1	a30001b1	a30002b1	a30003b1	a30004b1
nula	devět	jedna	šest	čtyři
jedna	pět	čtyři	osm	jedna
tři	dva	devět	jedna	šest
čtyři	šest	nula	tři	tři
devět	sedm	šest	nula	pět
pět	osm	dva	pět	osm
osm	jedna	pět	sedm	devět
sedm	tři	sedm	dva	dva
šest	nula	tři	čtyři	sedm
	čtyři	osm	devět	nula

a30005b1	a30006b1	a30007b1	a30008b1	a30009b1
devět	dvě	sedm	osm	devět
čtyři	nula	nula	šest	nula
šest	tři	šest	devět	čtyři
pět	čtyři	tři	pět	šest
nula	šest	čtyři	čtyři	sedm
dvě	devět	dva	sedm	tři
osm	pět	jedna	dva	pět
tři	jedna	pět	jedna	dva
sedm	sedm	osm	tři	osm
jedna	osm	devět	nula	jedna

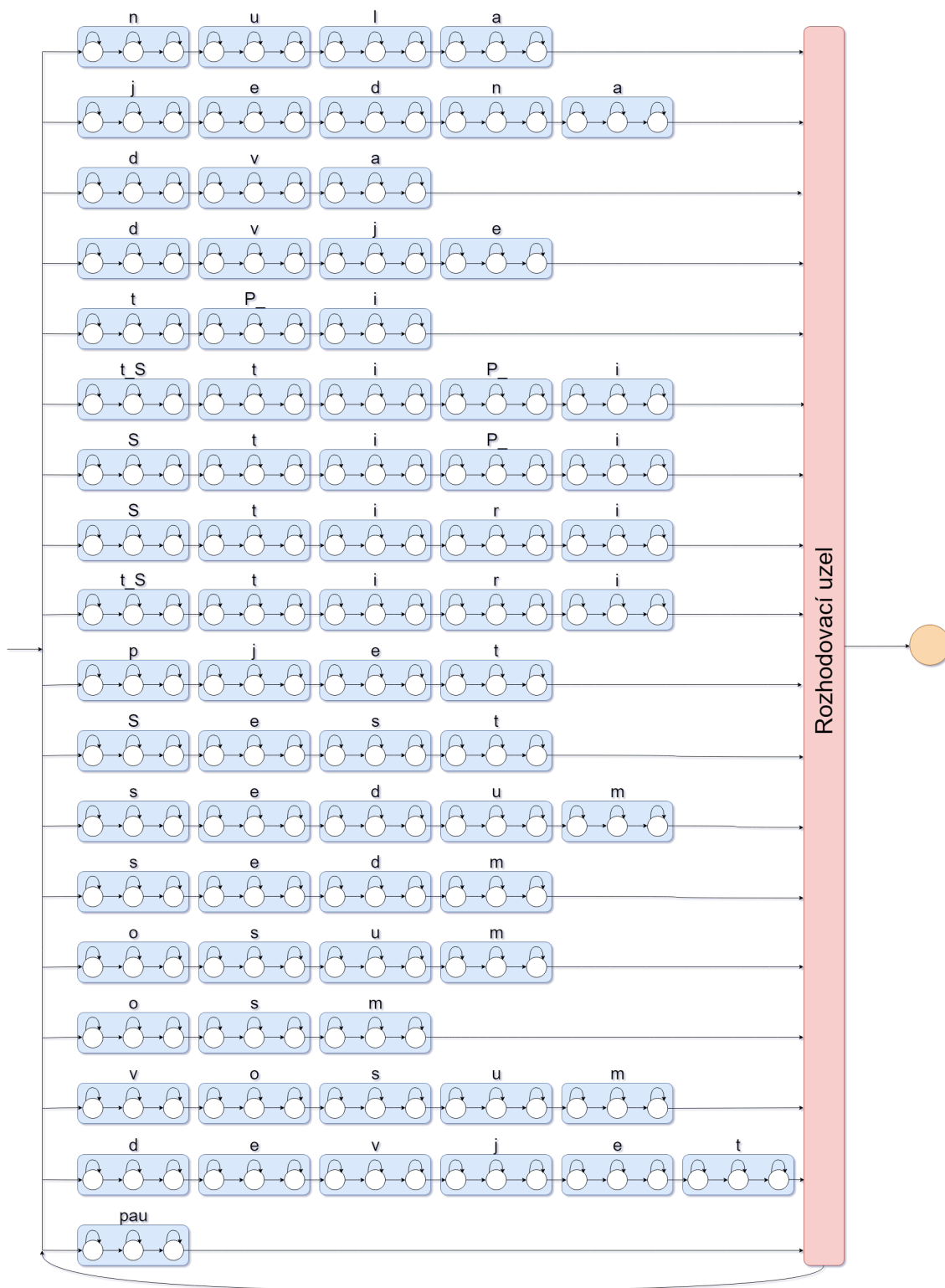
5 Argumenty a příklad spuštění

--likelihood-matrix LIKELIHOOD-MATRIX	slouží pro výběr souboru s likelihoody
--phonemes PHONEMES	slouží pro výběr souboru s fonémy
--zre-dict ZRE-DICT	slouží pro výběr souboru se slovníkem
--verbose	slouží pro zapnutí debugovacích výpisů

- `python run_model.py --likelihood-matrix ./eval/a30009b1.lik --phonemes ./dicos/phonemes --zre-dict ./dicos/zre.dict`

6 Závěr

S výsledkem našeho projektu jsme spokojení, necht' na development datech byla dosažena úspěšnost 100 %.



Obrázek 1: Schéma rozpoznávací sítě, modře jsou označeny fonémy, červeně rozhodovací uzel a oranžově koncový uzel. 4