

Projekt ZRE

Dekódovanie HMM

Peter Lukáč - xlukac11

Apríl 2020

1 Popis rozpoznávacej štruktúri

Likelyhoody každého fonému vo výstupe majú 3 stavy. Pre jednoduchšie spracovanie sú stavy každého fonému zredukované do jedného stavu sumou pôvodných 3 stavov. Rozpoznávacia sieť má jednoduchú schému. Sieť sa aplikuje na pevný počet rámcov vstupných likelyhoodov. V sieti sa vypočítajú likelyhoody pre každý model slova. Niektoré slova (2 a 8) majú viacero variant a vyberá sa varianta z vyšším likelyhoodom, v experimentoch toto rozšírenie zlepšilo výsledky. Výstupom siete je label modelu slova z najvyšším likelyhoodom po spracovaní daného počtu rámcov. V štruktúre predpokládame, že samotné slova v sebe neobsahujú pauzu.

1.1 Modely slov

Každý model má určitý počet foném. Rámce sa do stavov "nalievajú" v cykloch. Kumulatívna hodnota každého stavu sa berie ako maximum kumulatívnej hodnoty daného alebo predchádzajúceho stavu z predchádzajúceho rámca plus likelyhood súčasného rámca pre daný stav plus prechodová pravdepodobnosť. Výsledný likelyhood modelu je kumulovaný likelyhood posledného stavu.

1.2 Detekcia slov

Pri spracovávaní rámcov musíme určiť hranice slova pre ktoré budeme vyberať model slova s maximálnym likelyhoodom. Ináč povedané musíme rozhodnúť na ktorom rámci je koniec a začiatok každého slova aby sme mohli vybrať slovo s najvyšším likelyhoodom. Pre detekciu hranice rozdelíme vstupné rámce na dve časti a vypočítame likelyhoody celých štruktúr pre obe časti. Suma likelyhoodov oboch štruktúr bude najväčšia v bode kde sa jedno slovo končí a druhé začína, teda tam kde modely slov dosahujú najlepší "fitment".

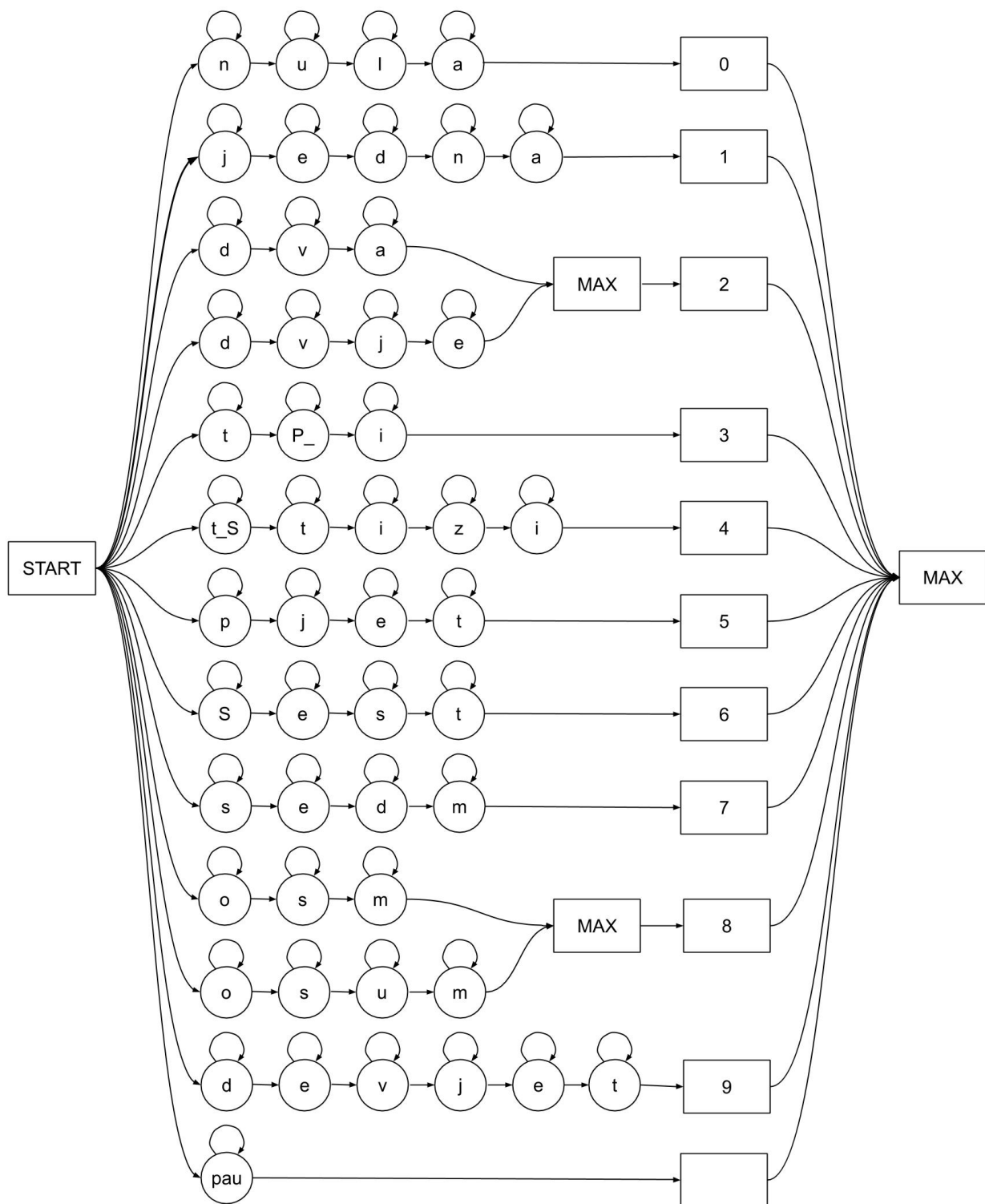
Túto znalosť využijeme tak, že budeme postupne deliť určitú oblasť, ktorá má približne 2 slová (pre dev data to vychádza na 120 rámcov). Oblasť delíme na 2 časti z určitým krokom (v našom prípade sme sa rozhodli pre 3). Vyberieme rozdelenie, ktoré má najväčšiu sumu likelyhoodov dvoch rozpoznávacích sietí. Z prvého rozdelenia vyberieme model slova s najväčším likelyhoodom na od daného bodu pokračujeme ďalej.

2 Implementácia

Dekóder je implementovaný v jazyku `python`. Program prakticky pozostáva z jedného cyklu pre výber oblasti s maximum likelyhoodom a funkcií pre výpočet likelyhoodu slov a likelyhoodu celej siete.

Celý proces delenia oblasti a hľadania maximum likelyhoodu je má značne veľa cyklov a v kombinácii s "pomalým" interpretom `python`-u sa nepriaznivo odrazilo na výkone. Spracovanie celej sekvencie môže trvať až niekoľko desiatok sekúnd. Na merlinovy to trvalo až 2-3 minuty.

Program sa spúšťa ako: `python3 main.py FILE`



Obr. 1: Rozpoznávací sieť

3 Zhodnotenie

Dekóder dosahuje uspokojivú úspešnosť. Nevýhodou je pomalé dekódovanie spôsobené možno až príliš komplikovaným riešením. Tento prístup bol zvolený preto, že využíva podstatu skrytých markových modelov.

4 Výsledky pre dev a eval data

- dev/a30000b1.lik: 0 1 3 4 9 5 8 7 6 2
- dev/a30001b1.lik: 9 5 2 6 7 8 1 3 0 4
- dev/a30002b1.lik: 1 4 9 0 6 2 5 3 8
- dev/a30003b1.lik: 6 8 1 3 2 7 2 4 2 9
- dev/a30004b1.lik: 4 1 6 3 5 8 9 2 7 0 0
- eval/a30005b1.lik: 9 4 6 5 0 9 8 3 7 1
- eval/a30006b1.lik: 9 0 3 4 5 9 5 1 7 8
- eval/a30007b1.lik: 7 0 6 3 3 9 1 5 8 9
- eval/a30008b1.lik: 8 6 9 5 4 7 1 3 0
- eval/a30009b1.lik: 9 0 4 6 7 3 5 2 8 1