



Grundlagen der automatischen Spracherkennung

Aufgabe 10 – Fertigstellung des Erkenners und Bestimmung der Wortfehlerrate

31.01.2024

Wentao Yu



Bestimmung des Erkennungsergebnisses in `hmm.py`

Mithilfe der Funktionen `viterbi()` und `getTranscription()` kann aus einer Matrix von posteriors (durch das DNN geschätzte Zustandswahrscheinlichkeiten) die Transkription berechnet und zurückgegeben werden:

```
def posteriors_to_transcription(posterior):  
    ...
```



Test der `posteriors_to_transcription()` Funktion in `uebung10.py`

1. In der Zip-Datei `Aufgabe10.zip` liegt die Datei `./data/TEST-WOMAN-BF-7O17O49A.npy`, die die Zustandswahrscheinlichkeiten für die Datei `./TIDIGITS-ASE/TEST/wav/TEST-WOMAN-BF-7O17O49A.wav` enthält. Bitte kopieren Sie diese in Ihren `data`-Ordner.
2. Wenden Sie Ihre Funktion `posteriors_to_transcription()` auf diese Zustandswahrscheinlichkeiten an.
3. Nutzen Sie Ihr trainiertes DNN, um die Posterior-Matrix aus der Datei `./TIDIGITS-ASE/TEST/wav/TEST-WOMAN-BF-7O17O49A.wav` mithilfe der Funktion `wav_to_posteriors()` zu berechnen. Anschließend wenden Sie die Funktion `posteriors_to_transcription()` auch auf Ihre selbst erstellten Zustandswahrscheinlichkeiten an.

Das korrekte Ergebnis ist für beide Fälle:

```
['SEVEN', 'OH', 'ONE', 'SEVEN', 'OH', 'FOUR', 'NINE']
```



Bestimmung der Wortfehlerrate

Berechnen Sie in

```
def test_model(datadir, hmm, model, parameters):  
    ...
```

in `uebung10.py` nun die Wortfehlerrate, zunächst für einige Dateien aus dem Testdatensatz (siehe `dataset/test.json`), und schließlich, wenn Ihre Funktionen getestet sind, für den gesamten Testdatensatz. Dazu benötigen Sie folgende Schritte:

1. Die WAV-Dateien liegen im Ordner `./TIDIGITS-ASE/TEST/wav`. Wenden Sie die Funktionen `wav_to_posteriors()` und `posteriors_to_transcription()` an, um die Transkriptionen (`word_seq`) für alle WAV-Dateien zu berechnen.



Bestimmung der Wortfehlerrate

2. Die Referenztranskriptionen liegen im Ordner `./TIDIGITS-ASE/TEST/lab`. Laden Sie zu jeder Datei die entsprechende Referenztranskription (`ref_seq`) mithilfe von `open()` und lesen Sie den Inhalt mit `read()` und `split()` ein.
3. Berechnen Sie die Anzahl der Deletions D , Insertions I , Substitutions S und die Gesamtzahl der Worte N für die jeweilige Datei mithilfe der Funktion

```
N, D, I, S = tools.needleman_wunsch(ref_seq, word_seq)
```



Bestimmung der Wortfehlerrate

4. Berechnen Sie schließlich die Wortfehlerrate für den gesamten Testdatensatz mit

$$WER = 100 \cdot \left(\frac{D_{ges} + I_{ges} + S_{ges}}{N_{ges}} \right)$$

indem Sie die Ausgabe von `tools.needleman_wunsch()` jeweils in D_{ges} , I_{ges} , S_{ges} , und N_{ges} akkumulieren.



Melden Sie sich gerne bei Fragen