

Übung 3

Aufgabe: Implementierung von IBM Modell 1

EM-Pseudocode für IBM Modell 1:

```
initialize  $t(e|f)$  uniformly
for 5 iterations
  set count(e,f) to 0 for all e,f
  for all sentence pairs (e_s,f_s)
    for all words e in e_s
      total = 0
      for all words f in f_s
        total +=  $t(e|f)$ 
      for all words f in f_s
        count(e,f) +=  $t(e|f)$  / total
  for all f
    total = 0
    for all e
      total += count(e,f)
    for all e
       $t(e|f)$  = count(e,f) / total
```

Welche Alignmentstruktur modelliert IBM Modell 1 in dem Pseudocode oben? In welcher Richtung sind 1:n-Alignierungen möglich? In welcher Sprache sind nicht alignierte Wörter möglich?

Welche Datenstrukturen nehmen Sie am besten für $t(e|f)$ und count(e,f)?

Implementieren Sie den obigen Code in Python und testen Sie ihn mit den drei Phrasen auf Seite 75 der Vorlesungsfolien. Lesen Sie die Phrasen aus zwei Dateien ein, in denen die Phrasen zeilenweise gespeichert sind.

Wie berechnen Sie das Viterbi-Alignment jedes Phrasenpaares? Formulieren Sie Pseudocode dafür.

Erweitern Sie Ihre Implementierung um die Viterbi-Alignierung geben Sie die Alignierung aus und testen Sie das Ergebnis.

Schicken Sie Ihren Code und Ihre Antworten auf die Fragen an `schmid@cis.lmu.de`.