Fortgeschrittene Methoden der statistische maschinelle Übersetzung

Übung 1

Abgabe (pdf, Zip) an `kaeshammer@phil.uni-duesseldorf.de`
bis einschließlich Donnerstag, 06.11.2014

Aufgabe 1

1.1

Wie berechnet man die Wahrscheinlichkeit für den folgenden Satz unter Annahme eines (bereits gelernten) Trigramm-Sprachmodells?

```
ich habe das rote Haus gesehen .
```

1.2

Was ist Smoothing (natürlich im Kontext von Sprachmodellen)? Warum benötigt man Smoothing?

1.3

Für die Schätzung der Parameter (also der Wahrscheinlichkeiten) eines n-gram-Sprachmodells kann man statt purer Maximum-Likelihood-Schätzung (ML estimation) oder *Add-1-Smoothing* auch *Add-α-Smoothing* verwenden:

$$\hat{P}_{Add\alpha}(w_i \mid h_i) = \frac{C(h_i w_i) + \alpha}{C(h_i) + \alpha|V|}$$

mit $0 \leq \alpha \leq 1$. Zeigen Sie, dass dies eine Wahrscheinlichkeitsverteilung ist.

Aufgabe 2

Implementieren Sie ein **Bigramm-Sprachmodell** (in Python 2.x) inklusive *Training* (Schätzen der Parameter/Wahrscheinlichkeiten aus einem Trainingskorpus) und Anwenden des Sprachmodells auf ungesehene Testdaten. Für das Schätzen der Parameter ver-

wenden Sie zunächst pure Maximum-Likelihood-Schätzung, dann auch ML-Schätzung mit Add-1-Smoothing.

Der Rahmen für die Implementierung ist bereits gegeben. Sie finden den Code und ein Mini-Trainingskorpus unter `http://user.phil-fak.uni-duesseldorf.de/~kaeshammer/smt14/material/u1.zip`. Der Aufruf lautet `python lm.py de.txt` und es gibt auch schon eine Ausgabe, die sich vervollständigt, wenn Sie die fehlenden Teile im Code implementieren. Überlegen Sie, ob die Ausgaben für die Bigramme und die Testsätze, die Sie erhalten, dem entsprechen, was Sie, gegeben das Testkorpus, erwarten bzw. wie man sie erklären kann.

Bitte beachten Sie: Über Wörter, die in Testdaten vorkommen, die aber in den Trainingsdaten nicht vorgekommen sind, können wir keine Aussage machen. Ein Ausweg ist es, Wörter, die im Trainingskorpus nur selten (z.B. ein Mal) vorkommen, durch ein spezielles Wort (z.B. `<UNK>`) zu ersetzen und diese nicht zu V hinzuzufügen. Auch in Testdaten werden dann Wörter $w \notin V$ mit `<UNK>` ersetzt.

Allgemeine Kommentare:

- Sie müssen sich nicht unbedingt an die im Code vorgegebenen Strukturen halten. Stellen Sie aber in jedem Fall sicher, dass ein Außenstehender Ihren Code versteht, zum Beispiel durch Kommentare. Falls Sie den ursprüngliche Aufruf des Programms verändern, fügen Sie bitte eine Readme-Datei hinzu, die spezifiziert, wie der Code aufzurufen ist.
- Die Python Standard Library (`https://docs.python.org/2/library/index.html`) dürfen Sie gerne verwenden, zum Beispiel für Mengen, Dictionaries (Assoziative Arrays) usw.
- Code von Programmen, die beim Ausführen Fehler erzeugen, wird nicht akzeptiert.