

8. Übungsblatt - Abgabe: 06.01.2015

Aufgabe 8.1 - WordNet

Lesen Sie Kapitel 19 aus Jurafsky & Martin von 19.1 bis einschließlich 19.3. (Der Text steht als Kopiervorlage im Vorlesungsordner, wenn Sie direkt aus dem Buch kopieren: benutzen Sie bitte die 2. Auflage.) Beantworten Sie mit Hilfe des Texts die folgenden Fragen:

- (a) Was versteht man unter einem Synset? Erläutern Sie an einem deutschen Beispiel.
- (b) Tabelle 19.2 gibt die WordNet-Relationen für Nomen an. Geben Sie für jede dieser Relationen jeweils ein Beispiel, bei dem der Begriff „Hund“ in seiner „Hauptlesart“ auf der linken Seite der Relation steht oder begründen Sie kurz, warum es für die Relation kein geeignetes Beispiel mit diesem Wort gibt. (Ignorieren Sie den letzten Eintrag „Derivationally Related Form“.)
- (c) Überlegen Sie sich drei Beispiele für unterschiedliche Arten von Relationen (und benennen Sie diese), die in WordNet nicht vorkommen, aber prinzipiell wünschenswert oder denkbar wären und geben Sie jeweils ein deutsches Beispiel für eine Instanz dieser Relation.

Aufgabe 8.2 - Semantische Relationen

Die folgenden Daten stammen aus einem Assoziationsexperiment, das in Saarbrücken durchgeführt wurde: Versuchspersonen wurde ein Wort präsentiert (Triggerwort), und sie sollten semantisch ähnliche Wörter dazu aufzählen. Im folgenden sind drei Triggerwörter, jeweils mit 5 häufig genannten assoziierten Wörtern, angegeben:

- **Fahrrad:** Fahrzeug, Sport, Rad, Pedal, Moped
 - **Erdbeere:** Kuchen, Frucht, Sahne, Garten, süß
 - **Buch:** Seite, Roman, Autor, Bibliothek, lesen
- (a) Beurteilen Sie für alle Paare von Triggerwort und Assoziation, ob die zwischen ihnen bestehende semantische Relation durch eine der WordNet-Relationen (Hyponymie/Hyperonymie, Meronymie/Holonymie, Antonymie) plausibel beschrieben wird.
 - (b) Geben Sie in den Fällen, in denen dies Ihrer Ansicht nach der Fall ist, die Relation an (z.B.: „Fahrrad – Fahrzeug: Hyponymie“) und versuchen Sie die Beziehung in der WordNet-Datenbank aufzufinden. Problem ist, dass es kein deutsches WordNet

mit einer frei zugänglichen Web-Schnittstelle gibt. Nehmen Sie deswegen den Umweg über das Englische: Passende englische Übersetzung wählen und angeben (ggf. mithilfe von Wörterbuch oder LEO) und im englischen WordNet (<http://wordnetweb.princeton.edu/perl/webwn>) recherchieren. Geben Sie die Resultate der Recherche an.

- (c) Versuchen Sie in den Fällen, in denen Ihrer Ansicht nach keine WordNet-Relation vorliegt, die semantische Beziehung mit eigenen Worten zu charakterisieren.

Aufgabe 8.3 WordNet über Kommandozeile

In dieser Aufgabe sollen Sie WordNet über die Kommandozeile benutzen: Auf der Vorlesungsseite ist unter *Nomen.txt* eine Textdatei verlinkt, die in jeder Zeile ein englisches Nomen enthält. Ihre Aufgabe ist es, als output eine Datei zu erzeugen, in der in jeder Zeile zunächst wieder das ursprüngliche Wort steht, dann ein tab und dann eine Liste aller Wordnet-Synonyme dieses Wortes, jeweils mit Kommas getrennt.

Gehen Sie dabei folgendermaßen vor:

- a) Im ersten Schritt sollen Sie auf Wordnet zugreifen und sich Synonyme anzeigen lassen. Mit *wn* sprechen Sie in der Coli die Wordnet-Installation an. *wn house -hypon* liefert Ihnen beispielsweise alle Hyponyme zum Nomen *house*. Geben Sie den Befehl an, den Sie brauchen, um sich für das Wort *house* alle Synonyme anzeigen zu lassen.
- b) Benutzen Sie jetzt die überwiegend schon bekannten Befehle *grep*, *sed* und *tr*. um nur noch alle Synonyme (also alles nach dem =>) von allen Wordsenses, durch Kommas getrennt, übrigzubehalten. Dabei kann ein Begriff mehrfach vorkommen. (Evtl haben Sie *tr* noch nicht benutzt. Schauen Sie sich die Doku an. Mit *tr '\n' ' ' <input_filename* kann man unkompliziert Zeilenumbrüche ersetzen.)
- c) Bis hierhin haben Sie immer nur ein einzelnes Wort bearbeitet. Um Ihre Befehlskette auf mehrere Wörter, den Inhalt von *Nomen.txt*, anzuwenden zu können, müssen Sie die Befehle in eine Datei, ein sogenanntes Shell-Skript, z.B. *wordnet.sh*, schreiben. Sie müssen dem Skript als Argument das Wort übergeben, dass es bearbeiten soll. Im Skript sprechen Sie dieses Argument dann mit *\$1* an. Das Shellskript *wordnet.sh*, dass das obigen Wordnet-Kommando ausführt, könnte also folgenden Code beinhalten: *wn \$1 -hypon*. Sie würden es mit *sh wordnet.sh house* für das Wort *house* aufrufen. Sie können eine neue Datei mit dem Befehl *touch datei.sh* anlegen und dann z.B. mit dem Editor *emacs* öffnen. Schreiben Sie ein entsprechendes Shellskript und testen Sie, dass Sie es für ein einzelnes Wort wie *house* anwenden können. Geben Sie den Inhalt Ihres Skriptes an.
- d) der Befehl *xargs* erlaubt es Ihnen, einen Befehl auf jedes Element einer Liste anzuwenden, z.B. jede Zeile einer mit *cat* geöffneten Datei. Sie werden *xargs* hier mit der Option *-I {}* aufrufen, um ein Argument an das Skript übergeben zu koennen. Benutzen Sie außerdem *paste*, um zwei Dateien zeilenweise nebeneinander zu drucken. Geben Sie die Befehle an, mit denen Sie Ihr Skript für alle Wörter aus *Nomen.txt*

Dr. Vera Demberg

Andrea Horbach

Einführung in die Computerlinguistik

<http://www.coli.uni-saarland.de/courses/I2CL-14/>

WS 2014/2015

anwenden und dann eine Datei erstellen, die je Zeile zuerst das Eingabewort, dann einen Tab und dann alle Synonyme enthält.

Abgabe in Gruppen von bis zu drei Studierenden am **06.01.2015** vor der Vorlesung.