Übungsblatt 1

17.10.18

Präsenzaufgaben

Aufgabe 1 Installation NLTK

Für die weiteren Übungen werden wir das Natural Language Toolkit (NLTK) für Python verwenden. Falls es nicht schon installiert ist, installieren Sie es jetzt.

(a) UNIX (Linux oder Mac)

Auf UNIX-Systemen ist Python, sowie der Paketmanager `pip`, in der Regel bereits installiert. Sie können also das NLTK einfach mittels Kommandozeile installieren.

Zunächst sollten Sie sicherstellen, dass Ihre Version von `pip` auf dem neuesten Stand ist:

```
tux@linux:~$ sudo -H pip3 install -U pip
```

Dann führen Sie die eigentliche Installation folgendermaßen aus:

Systemweit:

```
tux@linux:~$ sudo -H pip3 install -U nltk
```

Lokal:

```
tux@linux:~$ pip3 install --user -U nltk
```

Beachten Sie, dass Sie alternativ auch die Python-Distribution *Anaconda* installieren können. Dort ist das NLTK zusammen mit weiteren praktischen Paketen bereits vorinstalliert.

(b) Windows

Falls Sie Windows verwenden, müssen Sie zunächst sicherstellen, dass Sie Python installiert haben und anschließend das NLTK installieren. Am einfachsten geht dies, indem Sie die Python-Distribution *Anaconda* herunterladen und installieren (Link s. u.). Diese enthält Python, das NLTK, sowie weitere nützliche Pakete für den Umgang mit Daten.
Falls Sie es bevorzugen, nur das absolut Nötige zu installieren, empfiehlt sich ein Blick auf *Miniconda*. Diese Distribution enthält nur Python und *Anaconda* selbst. Das NLTK installieren Sie dann mithilfe des folgenden Befehls im *Anaconda Command Prompt*:

```
C:\Users\nutzer\Anaconda>conda install nltk
```

Für weiter Informationen siehe

- Installation des NLTK
  `http://www.nltk.org/install.html`
- Der Paketmanager `pip`
  `https://packaging.python.org/tutorials/installing-packages/`
- Installation von *Anaconda*
  `https://www.anaconda.com/download/`
- Leichtgewichtige Anaconda-Alternative *Miniconda*
  `https://conda.io/miniconda.html`

Aufgabe 2 Test der Installation, Herunterladen von Ressourcen

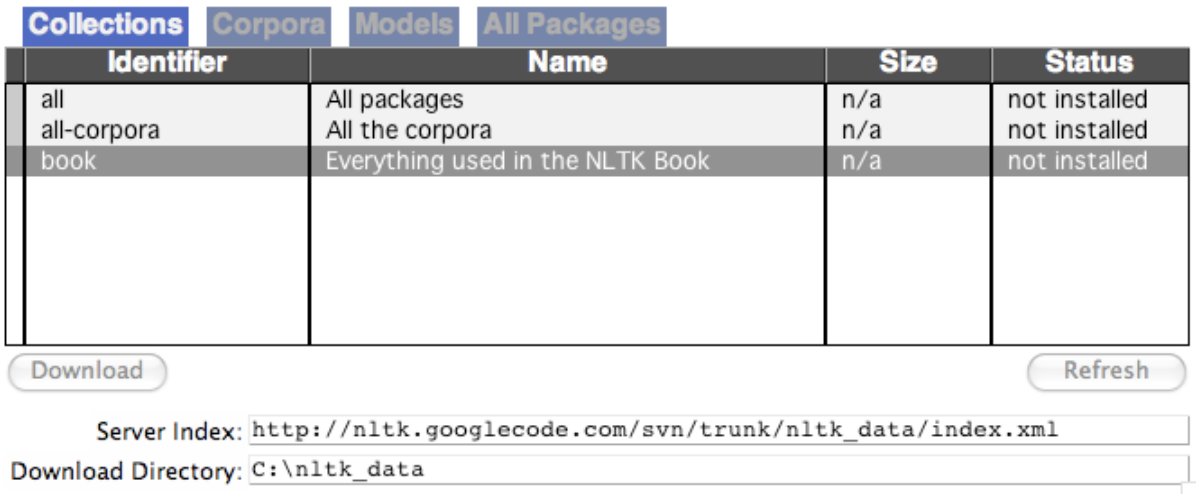
Vergewissern Sie sich, dass die Installation ordnungsgemäß funktioniert hat, indem Sie versuchen das NLTK in einer interaktiven Pythonsession zu importieren:

```
1 import nltk
2
```

Mit dem folgenden Befehl können Sie dann Korpora und andere Ressourcen herunterladen und in NLTK integrieren:

```
3 nltk.download()
4
```

Die Ressource *book* enthält alle Beispiele aus dem NLTK-Buch, d. h. weit mehr als wir benötigen werden. Sie können die Ressource entweder jetzt herunterladen oder im Laufe des Kurses ab und an den Download-Dialog erneut öffnen und nur die benötigten Daten einzeln laden.



Aufgabe 3 Installation tk

Wir werden im Laufe der Übung einige Graphen, z. B. Syntaxbäume, visualisieren wollen. Dafür eignet sich sehr gut das `tk`-Paket. Falls es nicht schon installiert ist, installieren Sie es nun:

Anaconda	bereits installiert
Miniconda	`conda install -c anaconda tk`
Debian/Ubuntu	`sudo apt-get install python3-tk`
OpenSUSE	`sudo zypper install python3-tk`

Aufgabe 4 Test: Syntaxbäume zeichnen

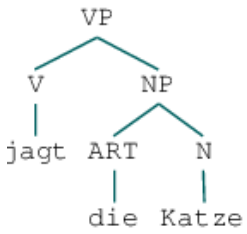
Zum Testen, ob NLTK erfolgreich installiert wurden, führen Sie folgende Befehle aus.

Zunächst konstruieren wir manuell einen Syntaxbaum:

```
1 from nltk.tree import Tree
2
3 np1 = Tree('NP', [Tree('ART', ['der']), Tree('N', ['Hund'])])
4 np2 = Tree('NP', [Tree('ART', ['die']), Tree('N', ['Katze'])])
5 vp = Tree('VP', [Tree('V', ['jagt']), np2])
```

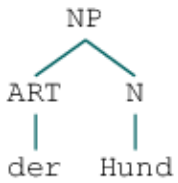
Um ihn anzuzeigen, genügt es, die Variable alleine in die letzte Zeile einer Zelle zu schreiben.

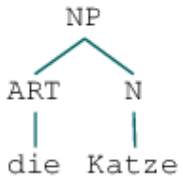
```
1 vp
```



Möchte man mehr als nur einen Baum in derselben Zelle anzeigen, bietet ‘display‘ mehr Flexibilität.

```
1 from IPython.display import display
2
3 display(np1)
4 display(np2)
```



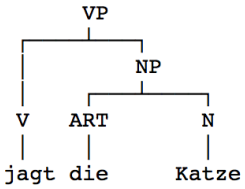


Für eine Darstellung in der Konsole gibt es zwei Möglichkeiten:

```
1 print(vp)
```

(VP (V jagt) (NP (ART die) (N Katze)))

```
1 vp.pretty_print(unicodelines=True)
```



Aufgabe 5 Installation Jupyter Notebooks

Parallel zu den Übungsblättern werden zu jeder Übung *Jupyter Notebooks* gleichen Inhalts angeboten werden. *Jupyter Notebooks* sind interaktive, aus einer Mischung aus Text, Code und Diagrammen bestehende Web-Dokumente, in denen Sie die Präsenzaufgaben lösen, sich Notizen machen und bequem experimentieren können.
Wollen Sie dieses Zusatzangebot nutzen, müssen Sie die Software zur Ausführung von *Jupyter Notebooks* installieren:

Anaconda	bereits installiert
Miniconda	`conda install jupyter`
pip	`pip3 install jupyter`

Auf der Kurs-Website finden Sie ein erstes Notebook zum Testen Ihrer Installation.

Aufgabe 6 Installation Jupyter Widgets

Die Übungsblätter beinhalten interaktiv aufbereitete Aufgabentypen, insbesondere Multiple-Choice-Fragen. Hierfür wird die Erweiterung `ipywidgets` verwendet, die Sie folgendermaßen installieren:

```
pip3 install ipywidgets
jupyter nbextension enable --py --sys-prefix widgetsnbextension
```

Falls Probleme auftreten, muss ggf. `widgetsnbextension` deinstalliert und wie beschreiben erneut installiert werden.

```
pip3 uninstall ipywidgets widgetsnbextension
```

Wenn Sie eine Anaconda-Installation durchgeführt haben, können Sie die Widgets auch mit conda installieren:

```
conda install -c conda-forge ipywidgets
```

Aufgabe 7 Test der Jupyter-Installation

Zum Testen, ob Jupyter und die Widgets erfolgreich installiert wurden, laden Sie das Notebook zur ersten Übung (`01-notebook.ipynb`) herunter, wechseln Sie mit `cd` zu dem Downloadordner und führen Sie folgenden Befehl aus:

```
jupyter notebook
```

Wählen Sie in der sich öffnenden Jupyter-Browsersitzung die `01-notebook.ipynb`-Datei aus und führen Sie im Notebook folgenden Codeblock aus, um die Installation der Widgets zu testen:

```
1 from ipywidgets import widgets
2 widgets.IntSlider()
```

Aufgabe 8 Installation Stanford Parser

In der Veranstaltung werden u. a. die Java-basierten Parser und Modelle der Stanford-NLP-Tools über das entsprechende NLTK-Interface verwendet. Das Stanford-CoreNLP-Paket beinhaltet (neben Modulen wie POS-tagger, NER, IE, coreference resolution) verschiedene Parser (insbesondere Dependenzparsing) und unterstützt viele verschiedene Sprachen. Das englische Modell ist im CoreNLP-Paket enthalten, Modelle für die anderen Sprachen können separat heruntergeladen werden.

Stanford CoreNLP	`https://stanfordnlp.github.io/CoreNLP/download.html`
Stanford german model	`https://stanfordnlp.github.io/CoreNLP/download.html`

Downloaden Sie dort `stanford-corenlp-full-2018-10-05.zip` sowie `stanford-german-corenlp-2018-10-05-models.jar` (Version 3.9.2).
Entpacken Sie die zip-Datei und merken Sie sich gut den Pfad des extrahierten Ordners sowie des deutschen Modells.

Java 1.8 installieren/Version überprüfen: `java -version`

Installieren Sie die aktuellste Java-Version (1.8 ist Voraussetzung für den Stanford Parser; Mac: Java 8 JDK).

Sie müssen evtl. die Umgebungsvariable `JAVAHOME` anpassen.

Für weitere Informationen siehe:

- `https://nlp.stanford.edu/software/lex-parser.shtml`
- `http://www.nltk.org/_modules/nltk/parse/stanford.html`

Den Stanford Parser gibt es auch als Webservice (inkl. Visualisierung):
`http://nlp.stanford.edu:8080/parser/`
`http://nlp.stanford.edu:8080/corenlp/process`

Aufgabe 9 Test: Stanford Parser

Zum Testen, ob der Stanford-Parser erfolgreich installiert wurden, führen Sie folgende Befehle aus.

(a) Deutsche Konstituentenbäume

Ersetzen Sie `PATH_TO_CORE` mit dem Pfad zum CoreNLP und `PATH_TO_GER_MODEL` mit dem Pfad zum deutschen Modell:[1]

```
tux@linux:~$ python3
```

[1] Zur Sicherheit sollten Sie absolute Pfade verwenden.

```python
from nltk.parse.stanford import StanfordParser
import os

PATH_TO_CORE = os.getenv("HOME") + "/stanford-corenlp-full-2018-10-05"
PATH_TO_GER_MODEL = os.getenv("HOME") + "/stanford-models-2018-10-05"

jar = os.path.join(
    PATH_TO_CORE,
    "stanford-corenlp-3.9.2.jar"
)
ger_model = os.path.join(
    PATH_TO_GER_MODEL,
    "stanford-german-corenlp-2018-10-05-models.jar"
)

parser = StanfordParser(
    jar, ger_model,
    model_path="edu/stanford/nlp/models/lexparser/" +
        "germanPCFG.ser.gz"
)

tree_list = list(parser.raw_parse('Dies ist mein erster Satz.'))
tree_list[0].pretty_print()
```

Sie sollten folgenden Output erhalten:

```
             ROOT
              |
              S
  ____________|_________________
 |      |             NP         |
 |      |       ______|_____     |
PDS   VAFIN PPOSAT   ADJA    NN   $.
 |      |     |       |      |    |
Dies   ist   mein   erster Satz   .
```

(b) Englische Dependenzbäume

Ein weiterer Test für das Dependenzparsing (mit dem englischen Modell, in `PATH_TO_CORE`):

```python
from nltk.parse.stanford import StanfordDependencyParser
import os

jar = os.path.join(
    PATH_TO_CORE,
    "stanford-corenlp-3.9.2.jar"
)
model = os.path.join(
    PATH_TO_CORE,
    "stanford-corenlp-3.9.2-models.jar"
)
```

```
12
13 dep_parser = StanfordDependencyParser(
14     jar, model,
15     model_path="edu/stanford/nlp/models/lexparser/" +
16     "englishPCFG.ser.gz"
17 )
18
19 result = dep_parser.raw_parse('I saw an elephant')
20 for parse in result:
21     print(parse.to_dot())
```

Sie sollten in etwa folgenden Output erhalten:

```
digraph G{
edge [dir=forward]
node [shape=plaintext]

0 [label="0 (None)"]
0 -> 2 [label="root"]
1 [label="1 (I)"]
2 [label="2 (saw)"]
2 -> 1 [label="nsubj"]
2 -> 4 [label="dobj"]
3 [label="3 (an)"]
4 [label="4 (elephant)"]
4 -> 3 [label="det"]
}
```

Hausaufgaben

Aufgabe 10 Troubleshooting

Vergewissern Sie sich, dass Sie für das nächste Mal eine funktionierende Installation zur Verfügung haben, und betreiben Sie ggf. Troubleshooting.

Informieren Sie sich auch im NLTK-Kapitel 1.2 ('Getting Started with NLTK'):
`http://www.nltk.org/book/ch01.html`, sowie im Vorwort:
`http://www.nltk.org/book/ch00.html` ('Software Requirements', etc.).

Aufgabe 11 Verwendung git

Die Inhalte von Vorlesung und Übung werden über Github zur Verfügung gestellt und können über die Weboberfläche eingesehen und heruntergeladen werden. Ein lokales Klonen und regelmäßiges Updaten mit git erleichtert aber das Beziehen der Dateien, insbesondere bzgl. der interaktiven Aufgaben, da diese zusätzliche Konfigurationsdateien benötigen, die so automatisch heruntergeladen werden.

Machen Sie sich mit der Verwendung von git vertraut und installieren Sie einen git-Client.