

Übungsblatt 1

20.10.21

Präsenzaufgaben

Aufgabe 1 Installation Python3

Wir werden in den Übungen die Programmiersprache *Python* verwenden. Bitte stellen Sie sicher, dass auf Ihrem System die Version 3 des Pythoninterpreters installiert ist.

(a) Windows

In aller Regel ist Python noch nicht installiert. Wir empfehlen die Python-Distribution *Anaconda* herunterzuladen und zu installieren (Link s. u.). Falls Sie es bevorzugen, nur das absolut Nötige zu installieren, empfiehlt sich ein Blick auf *Miniconda*. Diese Distribution enthält nur Python und *Anaconda* selbst.

(b) UNIX (Linux oder Mac)

Auf UNIX-Systemen ist Python, sowie der Paketmanager `pip`, in der Regel bereits installiert. Um zu überprüfen, ob auch die richtige Version installiert ist, können Sie folgenden Befehl ausführen, der zu keiner Fehlermeldung führen sollte:

```
$ python3
```

Sollte Python 3 nicht installiert sein, ist es empfehlenswert die Python-Distribution *Anaconda* zu installieren (Link s. u.). Dort sind die meisten von uns benötigten Pakete bereits vorinstalliert.

Miniconda ist eine schlankere Alternative ohne vorinstallierte Pakete.

Installationslinks:

- *Anaconda*
<https://www.anaconda.com/download/>
- Leichtgewichtige Anaconda-Alternative *Miniconda*
<https://conda.io/miniconda.html>

Aufgabe 2 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.

Wichtig: Folgen Sie den Hinweisen entweder zu `conda` oder zu `pip`, nicht zu beiden.

(a) **Anaconda/Miniconda**

Falls Sie *Anaconda* verwenden, ist das NLTK bereits installiert.

Falls Sie sich für *Miniconda* entschieden haben, installieren Sie das NLTK mithilfe des folgenden Befehls im *Anaconda Command Prompt*:

```
$ conda install nltk
```

(b) **Paketmanager pip**

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

```
$ sudo -H pip3 install -U pip
```

Dann führen Sie die eigentliche Installation folgendermaßen aus:

Lokal:

```
$ pip3 install --user -U nltk
```

Systemweit:

```
$ sudo -H pip3 install -U nltk
```

Für weitere Informationen siehe

- Installation des NLTK
<http://www.nltk.org/install.html>
- Der Paketmanager `pip`
<https://packaging.python.org/tutorials/installing-packages/>

Aufgabe 3 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
```

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

```
2| nltk.download()
```

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.

Collections	Corpora	Models	All Packages
Identifier	Name	Size	Status
all	All packages	n/a	not installed
all-corpora	All the corpora	n/a	not installed
book	Everything used in the NLTK Book	n/a	not installed

Download Refresh

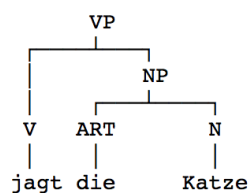
Server Index: http://nltk.googlecode.com/svn/trunk/nltk_data/index.xml

Download Directory: C:\nltk_data

Aufgabe 4 Test: Syntaxbäume zeichnen

Um zu testen, ob das NLTK erfolgreich installiert wurde, führen Sie folgendes Pythonskript aus:

```
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])
6
6 print(vp)
(VP (V jagt) (NP (ART die) (N Katze)))
7
7 vp.pretty_print(unicodelines=True)
```



Aufgabe 5 Installation Jupyter Notebooks

Die Übungsblätter der Veranstaltung werden in Form von *Jupyter Notebooks* angeboten. *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.

Zur Verwendung müssen Sie die Software zur Ausführung von *Jupyter Notebooks* installieren:

<i>Anaconda</i>	bereits installiert
<i>Miniconda</i>	<code>conda install jupyter</code>
<i>pip</i>	<code>pip3 install --user jupyter</code> (lokal) oder <code>sudo -H pip3 install jupyter</code> (systemweit)

*Aufgabe 6 Installation Jupyter Widgets (*optional*)

Begleitend zur Vorlesung werden zusätzliche Notebooks mit interaktiv aufbereitete Aufgabentypen (insbesondere Multiple-Choice-Fragen) zu primär linguistischen Aufgaben bereitgestellt.

Beachten Sie, dass die Verwendung dieser interaktiven Notebooks nur ein Zusatzangebot zum Selbstlernen darstellt. Eine erfolgreiche Installation ist also für die Teilnahme an der Veranstaltung nicht unbedingt notwendig.

Hierfür wird die Erweiterung `ipywidgets` verwendet, die Sie folgendermaßen installieren:

(a) **Anaconda/Miniconda**

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

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

(b) **Paketmanager pip**

Lokal:

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

Systemweit:

```
$ sudo -H 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
```

*Aufgabe 7 Installation tk (*optional*)

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:

<i>Anaconda</i>	bereits installiert
<i>Miniconda</i>	<code>conda install -c anaconda tk</code>
<i>Debian/Ubuntu</i>	<code>sudo apt-get install python3-tk</code>
<i>OpenSUSE</i>	<code>sudo zypper install python3-tk</code>
...	

Aufgabe 8 Test der Jupyter-Installation

Um zu Testen, ob Jupyter und die Widgets erfolgreich installiert wurden, laden Sie das Notebook zur ersten Übung (`01-notebook.ipynb`) von der Kurs-Website 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.

optional: Für einen Test der Installation der **Jupyter Widgets** (nur für interaktive Zusatznotebooks notwendig) führen Sie im Notebook folgenden Codeblock aus:

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

Nach dem Ausführen der Zelle, sollten Sie einen Schieberegler sehen, auf dem Sie Werte von 0 bis 100 einstellen können.

Aufgabe 9 Installation spaCy

Wir werden im Laufe der Veranstaltung *spaCy* als Beispiel eines modernen NLP-Systems verwenden. Die spaCy-Bibliothek beinhaltet u. a. Module zu Tokenisierung, POS-Tagging, Named Entity Recognition, Dependenzparsing und unterstützt viele verschiedene Sprachen. Mit den folgenden Befehlen laden wir das deutsche und das englische Modell herunter.

(a) Anaconda/Miniconda

```
$ conda install -c conda-forge spacy
$ conda install -c conda-forge spacy-lookups-data
$ python3 -m spacy download en_core_web_sm
$ python3 -m spacy download de_core_news_sm
```

(b) Paketmanager pip

Lokal:

```
$ pip3 install --user -U spacy
$ pip3 install --user -U spacy-lookups-data
$ python3 -m spacy download --user en_core_web_sm
$ python3 -m spacy download --user de_core_news_sm
```

Systemweit:

```
$ sudo -H pip3 install -U spacy
$ sudo -H pip3 install -U spacy-lookups-data
$ sudo -H python3 -m spacy download en_core_web_sm
$ sudo -H python3 -m spacy download de_core_news_sm
```

Für weitere Informationen siehe:

- <https://spacy.io/usage>
- <https://spacy.io/models>

Aufgabe 10 Test: spaCy

Testen Sie, ob spaCy und die Modelle erfolgreich installiert wurden, indem Sie folgende Befehle (in Python) ausführen:

1. Laden der Bibliothek und des deutschen Modells

```
1 | import spacy
2 | nlp = spacy.load('de_core_news_sm')
```

2. Analyse eines Satzes

```
3 | doc = nlp('Dies ist mein erster Satz.')
4 | for t in doc:
5 |     print(t.text, t.lemma_, t.pos_, t.tag_, t.dep_,
6 |           t.shape_, t.is_alpha, t.is_stop, sep='\t')
```

3. Visualisierung

```
7 | from spacy import displacy
8 | displacy.render(doc, style='dep')
```

Beachten Sie, dass Sie unverständliche Tags jederzeit wie folgt nachschlagen können:

```
1 | spacy.explain('nk')
```

Hausaufgaben

Aufgabe 11 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 12 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.

*Aufgabe 13 Jupyter Notebook Interface Tour

Die Weboberfläche der Jupyter Notebooks bieten vielfältige komfortable Möglichkeiten, Code zu schreiben, auszuführen und sich Notizen zu machen. Falls Sie daran interessiert sind, diese Möglichkeiten effektiv zu nutzen, können Sie rechts oben im Hauptmenü unter *Help* eine *User Interface Tour* machen (erster Menüeintrag).

*Aufgabe 14 Grundlagen der Programmierung in Python

Solide Vorkenntnisse in der Programmiersprache *Python* sind eine Grundvoraussetzung für die kommenden Übungen. Falls Sie bei sich hier Lücken vermuten, nutzen Sie die ersten Wochen des Semesters für einen Auffrischkurs: <https://www.learnpython.org>.