Introduction

Sprachverarbeitung (VL + Ü)

Nils Reiter

April 4, 2023

INSTITUT FÜR
DIGITAL HUMANITIES
UNIVERSITÄT ZU KÖLN

About Me

Figure: Nils (right)

Nils Reiter

- Master (»Diplom«) in Computational Linguistics (Saarland University)
- PhD in Computational Linguistics (Heidelberg University, 2007-2013)
- Postdoc at the IMS (Stuttgart University, 2014-2019)
- Professor for Computational Linguistic / Digital Humanities (Cologne University, since October 2019)

- `https://nilsreiter.de`
  `nils.reiter@uni-koeln.de`

About Me

Research Interests

- Artistic/non-standard use of language (e.g., humor, art, metaphors, literature), why do we express things in a certain (individual!) way?
- Operationalization of complex research questions and tasks
- Integration of quantitative/statistical research methods/results into hermeneutic research (e.g., interpretable machine learning)

→ ›Digital Humanities‹

About Me

Research Interests

- Artistic/non-standard use of language (e.g., humor, art, metaphors, literature), why do we express things in a certain (individual!) way?
- Operationalization of complex research questions and tasks
- Integration of quantitative/statistical research methods/results into hermeneutic research (e.g., interpretable machine learning)

→ ›Digital Humanities‹

- …also, I just like programming stuff

Section 2

This Class

Das Basismodul »Grundlagen der Computerlinguistik«

- Wintersemester
  - Seminar: Computerlinguistische Grundlagen (Jürgen Hermes)
    - Annotation, linguistische Grundlagen

Das Basismodul »Grundlagen der Computerlinguistik«

- Wintersemester
  - Seminar: Computerlinguistische Grundlagen (Jürgen Hermes)
    - Annotation, linguistische Grundlagen
- Sommersemester
  - Vorlesung: Sprachverarbeitung (Nils Reiter)
  - Übung: Computerlinguistik (Nils Reiter)
- Modulprüfung
  - Klausur: 13.07., 10:00–11:30

Das Aufbaumodul »Anwendungen der Computerlinguistik«

(Erst im nächsten Semester)

- Wintersemester
  - Übung: Deep Learning (Judith Nester)
  - Hauptseminar: Experimentelles Arbeiten in der Sprachverarbeitung
- Modulprüfung
  - Hausarbeit mit computerlinguistischem Experiment

Learning Goals

After this class (and the module), you will practically and conceptually

- be able to handle corpora
- have an overview over multiple machine learning algorithms
- be able to train your own models, know how to interpret and evaluate them
- know about pros and cons of various representations of language
- have an insight into corpus statistics

Weekly Flow

- Time slots
  - Tuesdays, 16:00–17:30: Exercise
  - Thursdays, 10:00–11:30: Lecture
- Course information
  - General information will be found here: `https://lehre.idh.uni-koeln.de/lehrveranstaltungen/sommersemester-2023/sprachverarbeitung/`
    - Slides will also be uploaded there
  - Literature will be accessible in Ilias (if not publicly available)

Studienleistung

- There will be an exercise every week
- We will start with the exercise together on Tuesdays
- You should finish the exercises at home
- You need to upload for final results three times in the semester (via Ilias)
  - ...but today doesn't count!

Section 3

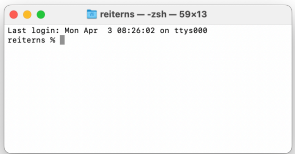
Command Line

Why?

- Powerful: Many »small tasks« can be done directly on the command line
  - Without writing a full-fledged program for it
- Available: Every computer offers a command line as the most basic way of accessing it
- Economic: No overhead compared to GUIs
  - You can get the full machine performance
  - This also makes it networkable
- Simple: Developing GUIs is hard and takes a lot of time
  - Research software cannot afford this
  - User interface on the command line is easy to do
    - In fact: We have done this already in Java 1

What?

- Text-oriented interface to computers
- Roots in teleprinter times
- Basic principles
  - Command prompt allows to enter commands
  - One process running at a time (unless ...)
  - Process output either printed directly or written in files
  - Most commands only give answers in case something is wrong



The Command Prompt

```
USERNAME @ HOSTNAME : WORKINGDIRECTORY $
```

The command prompt shows status information

- `USERNAME`: What is our user name?
- `HOSTNAME`: What's the computers name?
- `WORKINGDIRECTORY`: In which directory are we currently?
- `@ : $`: Separators

On the following slides, the prompt will be represented by $

Commands

```
$ COMMAND [OPTIONS] [ARGUMENTS]
```

- Options change the behavior of the command
  - Typically marked with a dash (-) or double dash (--)
- Arguments change on which the command is applied
- No clear boundary

Commands

Looking Up Things

Three important ways of looking up options, arguments etc.

- Shortest: Use the option `-h`, `--help` or `-help`
  - Shows options and arguments directly in the command line

Commands

Looking Up Things

Three important ways of looking up options, arguments etc.

- Shortest: Use the option `-h`, `--help` or `-help`
  - Shows options and arguments directly in the command line
- Slightly longer: Check out the man page: `$ man COMMAND`
  - Navigate with ↑, ↓; Q to quit.

Commands

Looking Up Things

Three important ways of looking up options, arguments etc.

- Shortest: Use the option `-h`, `--help` or `-help`
  - Shows options and arguments directly in the command line
- Slightly longer: Check out the man page: `$ man COMMAND`
  - Navigate with ↑, ↓; Q to quit.
- Fast, but with pitfall: `stackoverflow.com` / `google.com`
  - Pitfall: Many commands come in many different versions

Commands

Navigating the File System

- `ls`: List the content of the current directory
- `cd`: Change the directory

Commands

Manipulating the File System

- `mkdir`: Create a new sub directory
- `rmdir`: Delete an empty directory
  - Can only delete empty directories
- `rm`: Remove a file (or directory)
  - Potentially dangerous!
- `cp`: Copy a file (or directory)
- `mv`: Move a file (or directory)
  - I.e., copy and delete at the source
  - Also used for renaming

demo

Remote Computers

- `ssh` (secure shell) allows to connect to a remote computer / server
- You need to have a user account on this computer

Remote Computers

- `ssh` (secure shell) allows to connect to a remote computer / server
- You need to have a user account on this computer
- In this class, we'll be using `compute.spinfo.uni-koeln.de`
  - Reachable with the university network
  - You need to get an account via Ilias
    `https://www.ilias.uni-koeln.de/ilias/goto_uk_book_5164722.html`

```
1 $ ssh studentXY@compute.spinfo.uni-koeln.de
```

Section 4

Exercise

Exercise

- If not done yet: Set up a VPN connection on your computer. Follow these pages: `https://rrzk.uni-koeln.de/internetzugang-web/netzzugang/vpn`
- Make sure that you have an SSH client available on your computer.
- Connect to `compute.spinfo.uni-koeln.de` (by entering `ssh USERNAME@compute.spinfo.uni-koeln.de`).
- Create a new directory called `sprachverarbeitung` (to store everything related to this class).
  - `mkdir sprachverarbeitung`
- Change into that directory.
  - `cd sprachverarbeitung`
- Copy the file `/resources/gutenberg/2/1/4/2149/2149-8.txt` into that directory.
  - `cp /resources/gutenberg/2/1/4/2149/2149-8.txt ./`
- Have a look into the file.
  - `less 2149-8.txt` (You can exit this interface by pressing Q).
- Rename the file to, e.g., `poe.txt`
  - `mv 2149-8.txt poe.txt`
- Close the ssh connection by entering `exit`