Module 11
PoCoTo: Practice

Florian Fink

Centrum für Informations- und Sprachverarbeitung (CIS)
Ludwig-Maximilians-Universität München (LMU)

2015-09-15

Preparations

Downloading and installing PoCoTo

- download the binary distribution of PoCoTo: ocrcorrection.zip
- this will download a zipped archive file ocrcorrection.zip.
- extract (unzip) this archive to a convenient place somewhere in your user directory
- this will create a folder ocrcorrection
- in the folder ocrcorrection/bin, identify the appropriate executable for your operating system:
  - MS Windows: either ocrcorrection.exe or ocrcorrection64.exe
  - otherwise it is the file ocrcorrection
- if you like, you can create a link to the executable of the correction tool on your desktop:
  - drag and drop the executable from your file explorer onto your desktop
  - use the operation link from here if asked what to do with the file

Download the practice data

- download the following zip files: `1668-Hobbes-Leviathan.zip` and `1841-DieGrenzboten.zip`
- extract (unzip) the archives
- identify the contents of the archives:
  - `abbyy-xml` contains Abbyy XML formatted OCR output
  - `tess-hocr` contains Tesseract hOCR formatted OCR output
  - `tif` contains the image files
  - `gt` contains ground truth
  - `abbyy-txt, tess-txt` contain OCR output in pure text format (no annotations)

Starting PoCoTo

- before you start the application, make sure that the Java Runtime Environment (jre) is installed on your system
- to start the application, just double click on either the executable or on the link on your desktop
- PoCoTo's splash screen should open and after a while, PoCoTo's main GUI will be seen

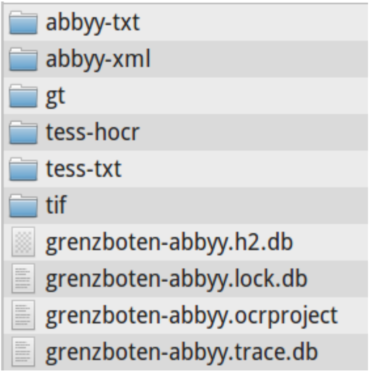
Working with PoCoTo

Creating a new project

- `PoCoTo` is described in detail in the manual which may be consulted for any questions
- First we will create a project using Abbyy XML data
- Click to `file -> new project` to open the project wizard
  - project name field: e.g. `grenzboten-abbyy`
  - project path: navigate to the directory of your XML data
  - note the 3 created files and look up their meaning in the manual
- Click `next`:
  - `OCR Input`: choose the directory containing the abbyy-xml files
  - `Input type`: in this case, ABBYY XML
  - `Encoding`: this should normally be set to UTF-8
- Click `next`:
  - `Img Dir`: choose the directory containing the tif files

Creating a new project (cont'd)

- Click `finish`
- now your directory structure should look like the image below!



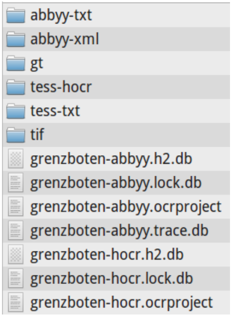
abbyy-txt
abbyy-xml
gt
tess-hocr
tess-txt
tif
grenzboten-abbyy.h2.db
grenzboten-abbyy.lock.db
grenzboten-abbyy.ocrproject
grenzboten-abbyy.trace.db

Page navigation and error correction

- Try to find an OCR error on any page but the first.
- Try to correct the OCR error.
- What is the most common error in the project?
- How can you examine the most common errors?
- How can you fix all those errors at once?
- After you have fixed all these errors, how do you save your work?

Creating another project using hOCR data

- Let's create another project using the hOCR files.
- Which directory should you choose for `OCR Input`?
- What kind of `Input type` should you select?
- What directory should you choose for `Img Dir`?
- Your directory structure should look like the image.



abbyy-txt
abbyy-xml
gt
tess-hocr
tess-txt
tif
grenzboten-abbyy.h2.db
grenzboten-abbyy.lock.db
grenzboten-abbyy.ocrproject
grenzboten-abbyy.trace.db
grenzboten-hocr.h2.db
grenzboten-hocr.lock.db
grenzboten-hocr.ocrproject

Comparing two projects

- What is the striking difference between the Abbyy Project and the hOCR Project?
- Try to find an OCR error on any page but the first.
- Try to correct the OCR error.
- Ok, now lets switch back to the Abbyy Project:
    - goto `file -> recent project`
    - select the project name of your Abbyy project

Ordering a document profile

- Try to order a document profile for the document's language:
    - Click to `profiler -> order document profile`.
    - In the window that opens up select the language of your document.
    - Click `Ok` and start the profiling.
- After the profiler has finished, what is the most common *error pattern* (e.g., `n -> u`)
- How do you correct this error pattern?

Measuring your correction effort

- let’s compare the error rate of your pages before and after correction
- before correction (with provided text files):

```
for i in gt/*.gt.txt; do j=‘basename $i‘;
echo $j; ocrevalutf8 accuracy $i tess-txt/”${j/.gt.txt/.tess.txt}”
> tess-txt/”${j/.gt.txt/.acc}”; done
```

- build a summary report:

```
cd tess-txt
accsum *.acc > accum
more accsum
```

- now export your project (`File -> Export -> as plain text, one file per page`) to a new directory (e.g., `tess-corr`) and apply the same procedure as above
- how many errors have been fixed?
- exporting your project as text files *before* correction, you can get the baseline (this is already provided as separate OCR output)

Books and sources

- for further experimentation, some book excerpts (ca. 100 pages each) can be found in the data directory of this workshop
    - preprocessed page images
    - ground truth (incomplete for Hobbes and Zonaras)
    - OCR output from ABBYY and Tesseract
- sources for the scans of complete volumes are given below
- many thanks to Kay Würzner (Grenzboten), Federico Boschetti (Zonaras) and Haide Friedrich-Salgado (Hobbes) for providing us with ground truth
- Goethe: Wahlverwandtschaften, vol. 1, 1809
    - Text in Deutsches Textarchiv
- Die Grenzboten, 1841
- Hobbes: Leviathan, Latin edition, 1668
- Zonaras: Epitome historiarum, 1870

Thanks for your attention!