Software Practicals
Winter Semester 2017/18

Database Systems Research Group
Heidelberg University
18.10.2017

Organization

Outline

- Overview of topics (today)
  - send application for a topic until Wednesday, October 25, 1pm
- First milestone (mid/end November)
  - prototype/part of software
  - summary of research (literature and related systems/tools)
  - further milestones by agreement with supervisor
- End of practical (mid/end January)
  - code (SVN / Buildscript / comments)
  - report (up to 10 pages)
  - presentation/demo of practical and software (10-15 minutes)

Organizational issues

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

- Application
  - by email to supervisor
  - brief list of relevant courses / prior knowledge
  - schedule and milestones for the practical
  - group work is not possible
  - application is binding (don't apply if you don't want to do the practical)
- Deadlines
  - presentation: planned for Tuesday, January 23, 2018
  - report: February 12, 2018
  - no extension possible
  - not finished = failed (grade 5,0)

Assessment

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

- Credit points (Leistungspunkte)
  - Beginners Practical (BP): 6 (4 FÜK) [for Bachelor students]
    - work load: 180 h (~1 ½ days/week)
  - Advanced Practical (AP): 8 (3 FÜK)
    - work load: 240 h (~2 days/week)
- Grading based on
  - code (readability, structure, functionality)
  - documentation (README, comments)
  - report
  - commitment
  - cool ideas!!
- IMPORTANT
  - talk to / communicate with your supervisor

Topics

Overview of Topics

UNIVERSITÄT
HEIDELBERG
ZUKUNFT
SEIT 1386

1. Information Networks from German Law Texts, **AP** (Gertz)
2. Integration and Exploration of Patient Data, **BP/AP** (Gertz)
3. Management and Exploration System for Clinical Admission Notes, **AP** (Gertz)
4. Network Topic Extraction and Visualization Implicit Networks, **AP** (Spitz)
5. Implicit Network Extraction and Exploration in R, **AP** (Spitz)
6. QA for the open-source ELKI data mining framework, **BP** (Schubert)
7. Density-Based Clustering, **AP** (Schubert)

Slides will be uploaded after the session to our webpage
https://dbs.ifi.uni-heidelberg.de/teaching/

AP: Information Networks from German Law Texts (Gertz)

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Project takes place in the context of the Doctoral Research Group "Digitales Recht", together with Law School at Heidelberg University

Given:

1. Bürgerliche Gesetzbuch (BGB) as JSON and/or XML-Files.
2. Data is structured based on paragraphs etc.

Task: Build a pipeline to extract and visualize word cooccurrence networks from law texts.

Subtasks:

- Familiarize yourself with the LOAD network model and law texts
- Implement and evaluate the network extraction pipeline

Languages / Tools:

- Python/Java/R; MongoDB for data storage

BP/AP: Integration and Exploration of Patient Data (Gertz)

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Project takes place in collaboration with the Institute for Computational Cardiology at Heidelberg University

Given:

1. Patient data and observations as flat files (mostly .csv),
2. Certain workload (query and data exploration tasks)

Task: Develop a database in support of efficient query processing and (visual) data exploration.

Subtasks:

- Design and implement database schema and data import wrappers
- Implement data query and exploration frontend

Languages / Tools:

- Python/Java; PostgreSQL/MongoDB for data storage

AP: Management and Exploration System for Clinical Admission Notes (Gertz)

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Project takes place in collaboration with the Institute for Computational Cardiology at Heidelberg University

Given:

1. Clinical admission notes, mostly as Word documents
2. Notes exhibit a typical document structure

Task: Develop a data management system to store, query and explore admission notes

Subtasks:

- Design and implement data model for admission notes
- Implement data wrappers
- Design and implement data exploration frontend

Languages / Tools:

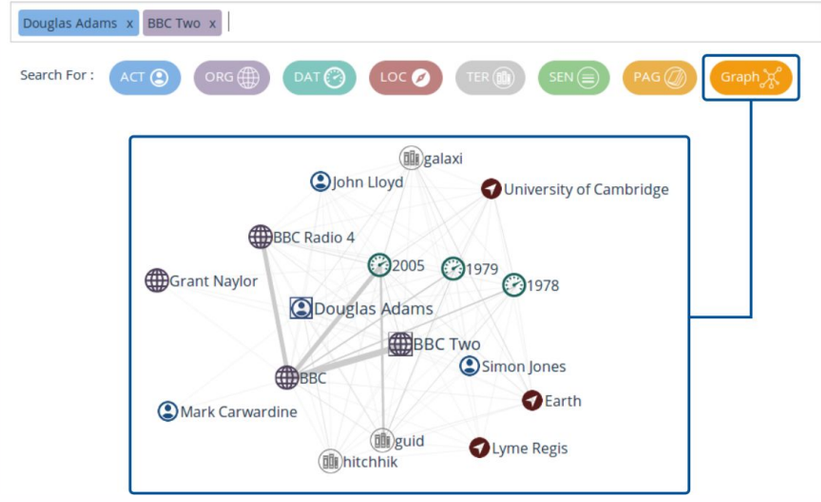
- Python/Java; MongoDB for data storage

AP: Network Topic Extraction and Visualization in Weighted Implicit Networks (Spitz)

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Given:

1. Theoretical background on implicit network extraction / storage [1] [2]
2. Java / JavaScrip / MongoDB implementation of a web query interface.



Task: Extend the interface to include network-based topic extraction.

Subtasks:

- Get to know the LOAD network model and EVELIN interface
- Implement a prototype of the topic extraction algorithm
- Implement the web interface in the EVELIN framework

Languages / Tools:

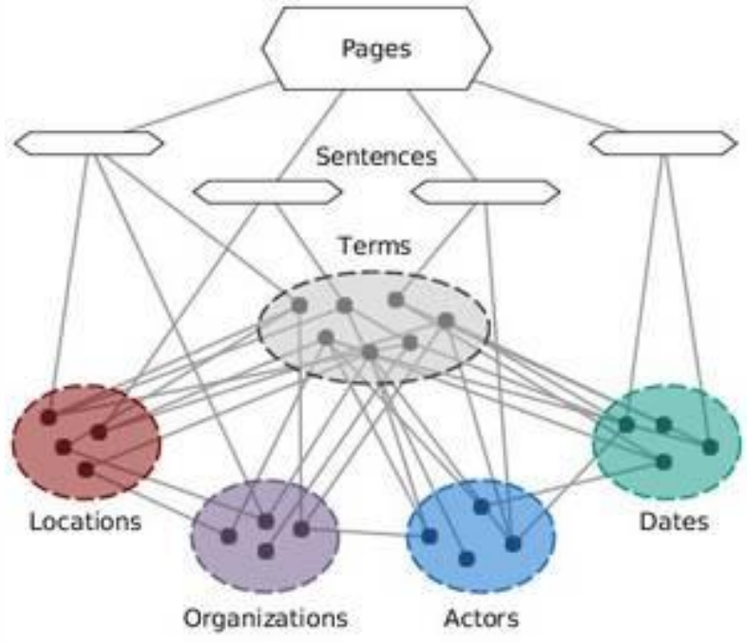
- Java ^ JavaScript ^ HTML/CSS ^ MongoDB

AP: Implicit Network Extraction and Exploration in R (Spitz)

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Given:

1. Theoretical background on latent network extraction / storage [1] [2]
2. Tidy NLP and text mining packages in the R programming language [3]



Task: Create an R package with an end-to-end pipeline (raw text -> network).

Subtasks:

- Get to know the LOAD network model and R
- Design a conceptual prototype pipeline
- Create an end-to-end package in tidy R format

Languages / Tools:

- R / basic C/C++ and NLP knowledge may be helpful

BP: QA for the open-source ELKI data mining framework (Schubert)

UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Given:

1. The ELKI data mining framework is written in Java and contains about 200.000 lines of Java code.
2. Only about 1/3rd of the code is currently covered by unit tests.

Task: Systematically contribute unit tests to this open-source project.

Subtasks:

- Familiarize yourself with JUnit4.
- Identify areas that need testing (e.g. linear algebra package).
- Design and implement unit tests.
- Fix bugs you find when testing.

Languages / Tools:

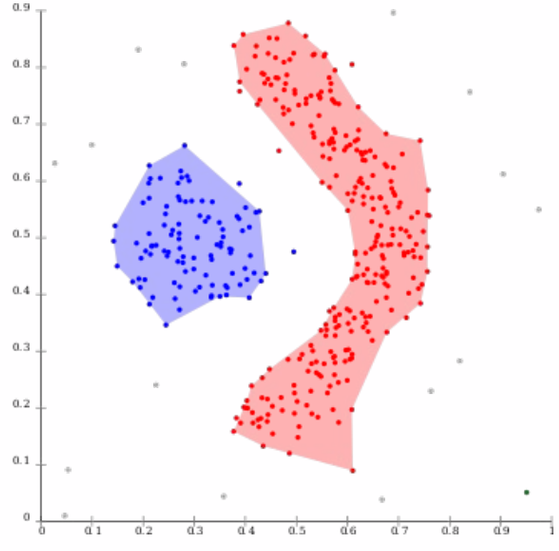
- Java knowledge is required, but you will learn a lot.

AP: Density-Based Clustering (Schubert)

UNIVERSITÄT HEIDELBERG ZUKUNFT SEIT 1386

Task:

1. Implement AnyDBC (DBSCAN variant) **or** CFSFDP in ELKI
2. Evaluate, benchmark, and optimize
3. Implement a new variant of the algorithm (idea provided by us)
4. Evaluate, benchmark, and optimize
5. Prepare a paper draft for submission to a conference or journal



Subtasks:

- Understand how ELKI works
- Use existing index structures of ELKI for accelerating the algorithm

Languages / Tools:

- Java knowledge is required
- Data mining / KDD knowledge is very helpful

Supervisors

- Michael Gertz (MG)

  gertz@informatik.uni-heidelberg.de

- Andreas Spitz (AS)

  spitz@informatik.uni-heidelberg.de

- Erich Schubert (ES)

  schubert@informatik.uni-heidelberg.de