Information Retrieval

WS 2016 / 2017

Lecture 14, Tuesday February 7th, 2017

(Course Evaluation, Exam, Work at our Chair)

Prof. Dr. Hannah Bast
Chair of Algorithms and Data Structures
Department of Computer Science
University of Freiburg

Overview of this lecture

- Organizational

– Your experiences with ES13	SPARQL, SQL, SQLite
– Official evaluation	results + discussion
– Infos about the exam	when, where, how, tasks
– Work at our chair	how, projects, next courses

Experiences with ES13 1/4

- Summary / excerpts
  - Many of you skipped this exercise sheet, because you already had enough points for the exam admission
  - Those who did it, were pleasantly surprised

    "This exercise sheet was really the best one in my opinion"

    "This sheet was fun and now I am interested in Sparql"

    "Ich fand diese Übung richtig nett"
  - Comment on arXiv-sanity by Andrey Karpathy (ML "guru"), which uses many of the techniques treated in this course

    "... what we are learning in the lecture makes very much sense and is used for real-world problems"

Experiences with ES13 2/4

- Results
  - SPARQL for "female leaders of Fortune 1000 companies"

```
SELECT ?person ?company WHERE {
    ?person "Leader of" ?company .
    ?person Gender "Female" .
    ?company "Appears in ranked list" "Fortune 1000"   }
```

  - Results

Indra Nooyi	PepsiCo
Marissa Mayer	Yahoo!
Meg Whitman	Hewlett-Packard
Meg Whitman	eBay

Very incomplete, the complete list is much longer

Experiences with ES13 3/4

- Seeing through walls with WiFi
  - WiFi signals indeed go through (some) walls

Weak block:	wood, plaster, cinder blocks, glass
Medium block:	water, stone, brick wall
Strong block:	concrete, metal, mirrors, ceramic

  - WiFi signals are partially reflected also by bodies (water!)

    Some of you thought only by metal → that's not true

  - Problem 1: "multi-path" reflection

    For moving objects, focus on what changes over time

  - Problem 2: relatively low resolution (centimeters)

    Still good enough for tracking person movement and even measuring heart rate and breathing frequency (use "phase")

Experiences with ES13 4/4

- Seeing through walls with WiFi
  - Videos

    WiTrack: 3D Motion Tracking Through Walls (YouTube)

    WiFi can indentify people through walls (YouTube)
  - Papers

    3D Tracking via Body Radio Reflections

    Smart Homes that Monitor Breathing and Heart Rate
  - Pictures

    New York Skyline in WiFi Resolution (Quora)

Results Course Evaluation 1/7

- Participants
  - Registered for exam: **77** ... similar last year (87)

    43 x MSc Inf, 17 x BSc Inf, 9 x M.Sc. ESE, 8 x Other
  - Participated in the evaluation: **73** ... **great !**

    That is 95% of the number of registrations, compared to 71% in the last year, where (for once) no points from the exercise sheets were needed for admission to the exam

    Nominations for teaching award: **55** ... **thanks a lot !**
  - In the following, a summary of your feedback
  - You find **all** the details linked on the course Wiki

Results Course Evaluation 2/7

- Style of the course
  - Learned a lot: 66% fully agree, 29% agree, 5% ok
  - Well explained: 73% fully agree, 22% agree, 5% ok
  - Asks own activity: 73% fully agree, 26% agree, 1% ok
  - Level of contents: 56% appropriate, 42% high, 2% other
  - Overall mark: 79% very good, 17% good, 3% ok
  - Most of you liked: the structure and organization, that everything was very practical and well-motivated, the explanations, the exercises (though too much work for some), the video recordings, the fast response times on the forum, the philosophical questions, humorous details
  - Criticism / suggestions: see slide 11

Results Course Evaluation 3/7

- Student's effort
  - Effort relative to ECTS ... 1 = very high, 5 = very low

30% x 1	45% x 2	25% x 3	0% x 4	0% x 5	this course
14% x 1	28% x 2	53% x 3	3% x 4	1% x 5	department average

  - So apparently more work than other CS lectures

    This was a frequent comment in your feedback

    Some of you said it was worth it, because you learned a lot

    But note: for about 25% it wasn't too much work at all

    But some also said, it was too much ... see the 30% above

Results Course Evaluation 4/7

- Materials / Online Support / Tutors
  - Materials helpful: 85% fully agree, 11% agree, 3% ok
  - Consumed lecture by presence or by video recordings:

    23% presence, 52% recordings, 21% both, 4% none this course

    35% presence, 15% recordings, 20% both, 30% none dep average
  - Video recordings (Frank, Alexander): great quality, **thanks!**
  - Assistant (Patrick): great support behind the scenes, in particular for the exercise sheets and in the forum, **thanks!**
  - Tutors (Björn, Claudius, Raghu, Natalie, Axel, Johanna): very friendly and helpful, fast and detailed feedback (for most), competent, fair, always available ... **thanks!**

    Many of you explicitly thanked the whole team of the course

Results Course Evaluation 5/7

- Criticism / Suggestions
  - Some exercises take too much time; maybe alternate between more difficult and easier sheets
  - Switching between languages is hard (Python ↔ Java/C++)
  - A lot of the time goes into mere "programming"

    Ok, but also shows that your programming skills need improving
  - More help for the sheets requiring Python (numpy/scipy)
  - Re-introduce "Implementation Advice" slides
  - Tests helpful but too strict **and** not extensive enough ... hmm
  - Git instead of SVN (two people) … Git has **no** access control
  - More emphasis on the theoretical part (one person)

Results Course Evaluation 6/7

- Planned improvements **from last year**
  - Try to get HS 026 (if there are again so many people) ✓
  - Provide DIY tutorial on SVN + Jenkins in English ✓
  - Provide Python crash course and/or cheat sheet ✗
  - Provide live tutorial every four weeks or so ✓
  - More mathematical background for those who want it ✓
  - Even better error checking of exercise sheets ✓
  - Further optimize time management ✓
  - Consider the extensive notes I took for each lecture ✓

Results Course Evaluation 7/7

- Planned improvements **for next year**
  - Clarify that test cases are mandatory in Lecture 1, but also clarify that they do not have to be implemented 1-1
  - Give high priority (again) to error and consistency checking of exercise sheets and TIP files ... continue awarding cookies for mistakes in sheets / TIP files
  - More theoretical sheets, in alternation with the (more time-extensive) implementation sheets
  - Finish Python cheat sheet and make it available early
  - Offer live tutorials again, until nobody comes anymore
  - I again took extensive notes on how each lecture went

    Many changes compared to slides from previous years

Exam 1/7

no RED

- Where, when, what to bring
  - Oral exams (B.Sc. Computer Science students only):

    Wednesday – Friday, February 22 – 24, 30min in 51-02-28
  - Written exam (all other participants):

    Tuesday, February 21, 14:00 – 16:30 in HS 026+036
  - Please bring: **student id**, **colored pens**, **brain**

    student id: make sure you look like your photo or vice versa

    colors: greatly improves readability for examples / drawings

    brain: greatly improves quality of answers in general

Exam 2/7

- Type of exam
  - The written exam is **open book**

    That is, you can bring books, paper, etc ... but please be ecological when printing out slides + good for your karma

    Electronic devices of any kind are obviously not allowed
  - In the oral exam, ask us if you are missing some detail

    But no time to start understanding things for the first time, during neither the oral nor the written exam
  - There will be a sub-forum for questions about the exam

    Answer speed will be slightly slower in the semester break

    Don't ask all your questions in the night before the exam

Exam 3/7

- Types of questions
  - **Type 1:** Do algorithm (or variant thereof) by example, like we often did in the lecture ... see colored pens
  - **Type 2:** Implement a small program (in Python, Java or C++) ... small indeed, max 10 lines per task !
  - **Type 3:** Small calculations or proofs ... see brain
  - In general: the emphasis is on (basic) understanding, not on learning things by heart
  - Important: all the contents + insights from the **exercises** is (very) relevant for the exam

Exam 4/7

- Preparation
  - To check whether you understood something:

    Put away the material from the lecture and try to write it down / prove it **in your own words / formalism**

    There is no point in learning the individual steps of a proof or argument by heart ... it doesn't work that way
  - Once the basic stuff from the lecture is understood, the best preparation is to do all the exercise sheets

    If you work your way through all the exercise sheets (yourself), there is no way you can fail the exam
  - Also: work through the old exams (linked on the Wiki)

Exam 5/7

c(1) = 1
c(2) = 01
c(3) = 001
c(4) = 0001

- Task 2 from WS 15/16 exam

2.1 1|1|0 0 0 0 1|0 0 1|1|1|0 0 1| proof → see Wiki

1 1 5 3 1 1 3

2.2

```
def decode(bits):
    x = 1
    result = []
    for bit in bits:
        if bit == 0:   x += 1
        if bit == 1:
            result.append(x)
            x = 1
    return result if x == 1 else []
```

Exam 6/7

- Task 2 from WS 13/14 exam

Prefix which minimizes the ED

2.1 $x = \text{angela}$

$y = \text{angelina}$

$PED(x,y) = PED(\text{angela}, \underline{\text{angelin}}\text{a}) = 1$

$PED(y,x) = PED(\text{angelina}, \underline{\text{angela}}) = 2$

2.2. $\text{2-grams}(x) = \{\underline{an}, \underline{ng}, \underline{ge}, \underline{el}, la\}$

$\text{2-gram}(y) = \{\underline{an}, \underline{ng}, \underline{ge}, \underline{el}, li, in, na\}$

$\text{comm}(x,y) = 4$

2-grams of x

2.3. $x = y \Rightarrow \text{comm}_2(x,y) = |x| - 1$

$PED(x,y) = 1 \Rightarrow 1$ ED operation

$\Rightarrow$ "destroys" at most 2 2-grams

$\Rightarrow \text{comm}_2(x,y) \geq |x| - 3$

Exam 7/7

- Task 4 from WS 12/13 exam

Work at our Chair 1/5

- How we work
  - We solve practically relevant (usually hard) problems

    Route Planning, Transit Maps, Search As You Type, Semantic Search, Question Answering, PDF extraction, ...
  - We use theory as a (vital) tool

    All our work has a solid theoretical understanding, otherwise solving complex problems remains hacking and guesswork
  - We make our software + results available to the public

    This requires an effort to write good software, good documentation, nice user interfaces, and so on ...

Work at our Chair 2/5

- Supervision
  - Similarly as in the lecture:

    Very good infrastructure + support, but apart from that you are supposed to work independently

    Great for enthusiastic people who care about practical stuff and who want to get things done

Work at our Chair 3/5

- Machine Learning
  - We are building more and more on **machine learning** to solve our problems

    Not because it's fashionable ... but because it's practical

    It's quite obvious that learning is the future for problems like natural language understanding
  - You have seen a few learning algorithms in this lecture

    The complexity lies not so much in the algorithms, but in understanding how and why they work how well

Work at our Chair 4/5

- Current projects and demos

– Route planning (part of Google Maps)	demo
– World-wide public transit visualization (Travic)	demo
– Automatic drawing of nice transit maps	demo
– Large-scale SPARQL+Text search (QLever)	demo
– Search-as-you-type semantic search (Broccoli)	demo
– Question Answering (Aqqu)	demo
– Question Completion	demo
– Text extraction from PDF (Icecite)	paper

Work at our Chair 5/5

- Upcoming courses
  - **Algorithms and Data Structures** ... in the SS 2017

    Basic course for 2nd semester B.Sc. Informatik students
  - **Information Retrieval** ... in the WS 2017/2018

    You know it ... become a tutor if you do a great exam !
  - **Programming in C++** ... in the SS 2018

    2nd semester B.Sc. Info + 4th semester B.Sc. ESE
  - **B.Sc. / M.Sc. projects or theses**

    Offered all the time, just ask (if no response, ask again please)

    It helps if you attended one of our lectures, with a good mark