data analytics lab ETH zürich

assignment 1 - web crawling

introduction

In this first assignment, you will write your own Scala Web crawler that autonomously traverses (a small portion of) the Internet, following the pages' hyperlink structure. Every time the crawler encounters an unseen page, that page, in turn, is crawled and analyzed.

Start from this URL and try to discover all links:
http://idvm-infk-hofmann03.inf.ethz.ch/eth/www.ethz.ch/en.html

In real-world Web crawling scenarios you want to discover content from many different sources, potentially exploring the entire Web. To make our toy crawling process somewhat more finite in terms of run times, please ensure that your crawler only follows in-domain links, without leaving the confines of our teaching Web server.

material

No external Web crawling libraries should be used. If you are unsure whether certain libraries are permissible, please contact us.

output

Your crawler should accurately compute the following statistics:

- How many distinct urls can be found?
- How many of those are exact duplicates of other pages?
- How many unique pages are mostly written in English?

- How many are near duplicates?
- What is the (case insensitive!) frequency of the term “student” in all non-duplicate, English pages?

deliverables

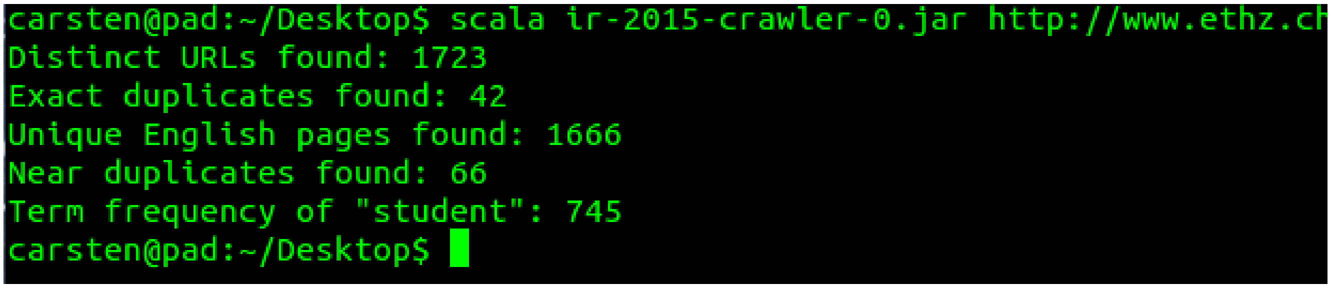
We ask you to send us three files per group:

- ir-2015-code1-[groupid].zip containing the entire code of your crawler.
- ir-2015-crawler-[groupid].jar your executable crawler. It should be a command line script responding to the following call: scala ir-2015-crawler-[groupid].jar url
- ir-2015-report1-[groupid].pdf simply listing the five statistics produced by your crawler.

For each of these files, replace [groupid] with your individual group id.

example

This is what your script should produce:



```
carsten@pad:~/Desktop$ scala ir-2015-crawler-0.jar http://www.ethz.ch
Distinct URLs found: 1723
Exact duplicates found: 42
Unique English pages found: 1666
Near duplicates found: 66
Term frequency of "student": 745
carsten@pad:~/Desktop$
```

grading

To obtain a perfect grade, your report should contain the correct dataset statistics, your executable crawler should produce correct stats when we re-run it on unseen data (in the same format) and, finally, your crawler should terminate in reasonable time.

submission format

Please submit all three deliverables via email to carsten.eickhoff@inf.ethz.ch by the end of October 14th. If you want me to find your submission and notify you upon receipt, please use the following subject line: ir-practical-2015-1-[groupid] Since some mail servers refuse sending/receiving large attachments (the typical threshold being in the 15MB area), please make sure in advance that you do not include any unreasonably large amounts of data to ensure receipt of your submission.

code

TinyIR code base

faq

Q *I get OutOfMemoryError exceptions.*

A You can increase the heap size. But before, think for a second if there is a good reason why the standard Scala heap space (often 256MB) is not sufficient. To increase it, add something like -Xmx2g to the scala command line or in Eclipse at Run→Run Configurations → "your filename" → Arguments → VM arguments.

Q *Which links should I follow?*

A Please follow only links to *.html resources and disregard any anchors (#) or get parameters (?).

Q *What to do with URLs for which the Web server generates directory listings?*

A Do not go there. Exclusively visit *.html URLs.

Q *What to do with relative-path URLs?*

A These need attention! Without treatment, the number of distinct URLs explodes but ignoring these URLs altogether might make you miss content.

Q *Pages A, B, C are all identical but have different URLs. How many exact duplicates do I count?*

A 2.

Q *The order in which I parse near-duplicates may influence counts. Do I have to account for this?*

A As long as you start from the given URL and do not randomize your frontier, all should be well.

Q *What exact term frequency are we supposed to count?*

A Count the frequency of the (case-insensitive) string ``student'' in all non-duplicate, English pages.

Q *How to export an executable *.jar?*

A There are many ways to success. Eclipse allows you to export via: Export -> Java -> Runnable JAR file. SBT and FatJar
are helpful alternatives.