

GuUCLe - The UCL Search Engine

Emmet Cassidy, Jason Cheung, David Kelly & Nicholas Read



Aims

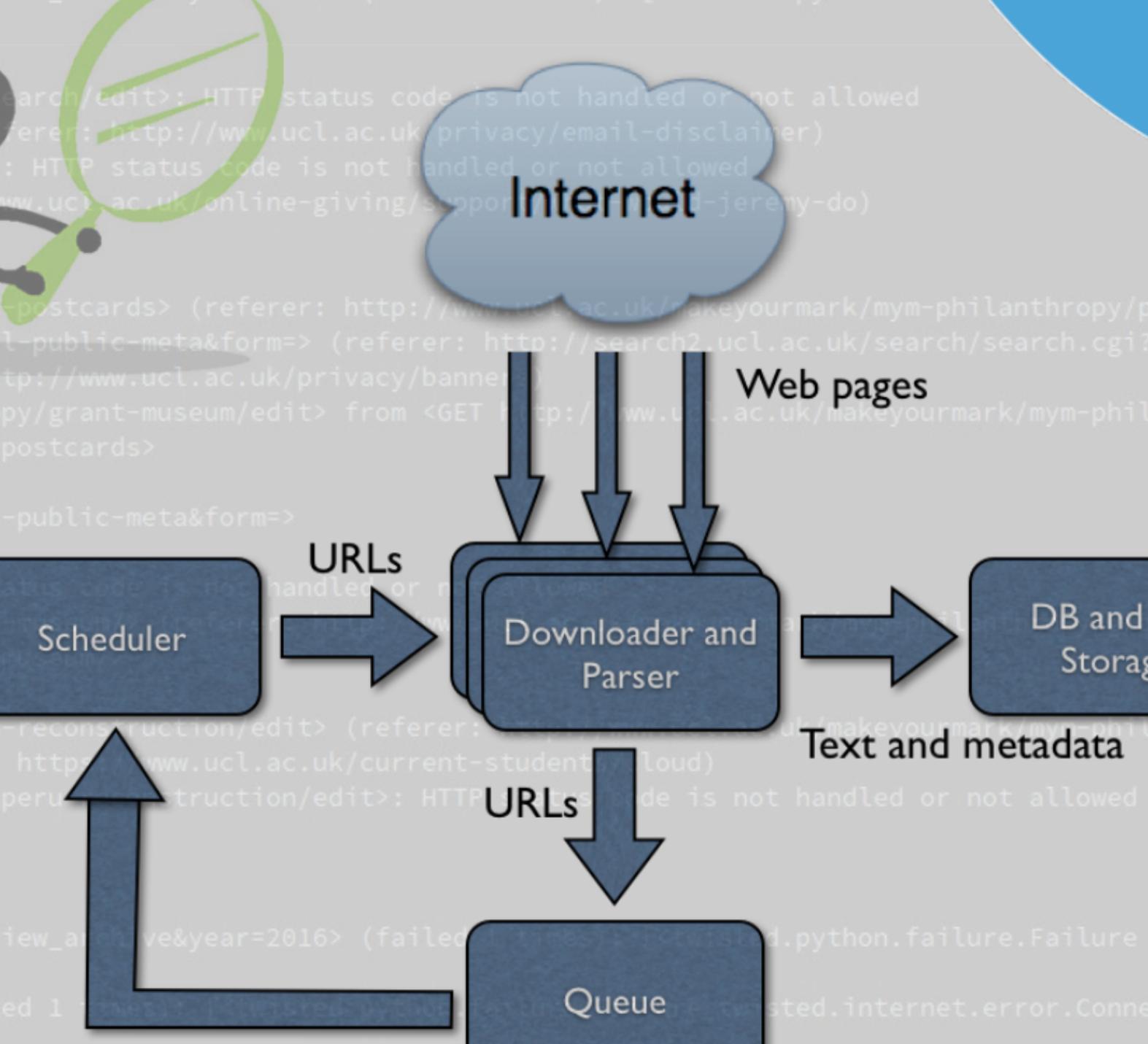
In this project we aim to implement a live search engine that is capable of indexing and searching the ucl.ac.uk website, including all sub-domains.

Using open source information retrieval and parsing packages we will implement a PageRank algorithm and evaluate the efficacy of our solution with respect to the existing UCL search engine.

Query Methodology

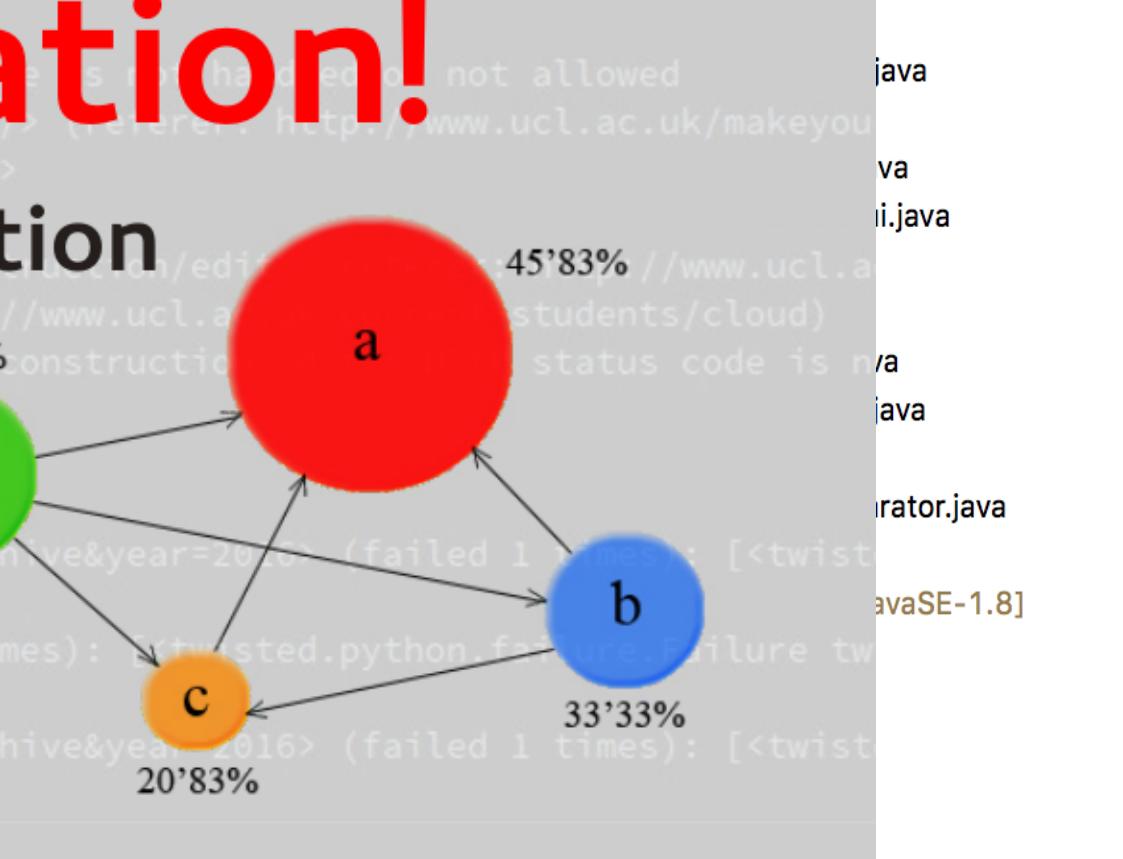
We intend to implement the extended Boolean search paradigm, adjusting the ranking of each result according to the occurrence frequency of search terms. To improve upon this we may combine term frequency with PageRank in a novel normalized metric to augment ranking of more authoritative results.

Custom-built webcrawler!



PageRank implementation!

We will write our own efficient implementation of the PageRank algorithm



```
2
3 import java.util.ArrayList;
4 import java.util.Collections;
5 import java.util.HashMap;
6 import java.util.HashSet;
7
8 /**
9  * Class for representing a collection of webpages
10 * as a graph,
11 * this can be used for link analysis ie pagerank.
12 *
13 * adjacency matrix represents a graph on a list of its connections
14 */
15
16 public class LinksGraph {
17     private ArrayList<Webpage> pages;
18
19     public void addPage(Webpage page) {
20         if (!pages.contains(page)) {
21             pages.add(page);
22         }
23     }
24
25     public void removePage(Webpage page) {
26         if (pages.contains(page)) {
27             pages.remove(page);
28         }
29     }
30
31     public void printPages() {
32         System.out.println("pages added");
33         for (Webpage page : pages) {
34             System.out.println(page);
35         }
36     }
37
38     public void calculatePageRank() {
39         // calculate pagerank here
40     }
41
42     public void printPageRank() {
43         // print pagerank here
44     }
45
46     public static void main(String[] args) {
47         LinksGraph graph = new LinksGraph();
48         graph.addPage(new Webpage("http://www.ucl.ac.uk/disclaimer", 0.004016338488204643));
49         graph.addPage(new Webpage("http://www.ucl.ac.uk/foi", 0.004016338488204643));
50         graph.addPage(new Webpage("http://www.ucl.ac.uk/accessibility", 0.004016338488204643));
51         graph.addPage(new Webpage("http://www.ucl.ac.uk/privacy", 0.004016338488204643));
52         graph.addPage(new Webpage("http://www.ucl.ac.uk/cookies", 0.004016338488204643));
53         graph.addPage(new Webpage("http://www.ucl.ac.uk/", 0.003533970644766658));
54         graph.addPage(new Webpage("http://www.ucl.ac.uk", 0.003490070866597949));
55         graph.addPage(new Webpage("http://www.ucl.ac.uk/prospective-students", 0.0034371459660996044));
56         graph.addPage(new Webpage("http://www.ucl.ac.uk/students", 0.0034371459660996044));
57         graph.addPage(new Webpage("http://www.ucl.ac.uk/staff", 0.0034371459660996044));
58         graph.addPage(new Webpage("http://www.ucl.ac.uk/prospective-students/", 0.003387959636202925));
59         graph.addPage(new Webpage("http://www.ucl.ac.uk/contact-list/", 0.0033006382915422494));
60         graph.addPage(new Webpage("http://www.ucl.ac.uk/media", 0.003160223363268917));
61         graph.addPage(new Webpage("http://www.ucl.ac.uk/contact-list", 0.003124503918339192));
62         graph.addPage(new Webpage("http://www.ucl.ac.uk/departments/faculties", 0.0030714122599279463));
63         graph.addPage(new Webpage("http://www.ucl.ac.uk/maps", 0.0030714122599279463));
64         graph.addPage(new Webpage("http://www.ucl.ac.uk/alumni", 0.003056908368700509));
65         graph.addPage(new Webpage("http://www.ucl.ac.uk/library/", 0.003056908368700509));
66         graph.addPage(new Webpage("http://www.ucl.ac.uk/museums", 0.003056908368700509));
67         graph.addPage(new Webpage("http://www.ucl.ac.uk/qatar", 0.003056908368700509));
68
69         graph.printPages();
70     }
71 }
```