

Designing for users: an holistic approach to teaching Information Retrieval

Ian Ruthven, David Elsweller and Emma Nicol
Department of Computer and Information Sciences
University of Strathclyde
Glasgow, G1 1XH
ir@cis.strath.ac.uk

Many Information Retrieval courses take a reductionist approach: reducing the complexity of Information Retrieval systems into a series of decomposed modules for indexing, retrieval, and results presentation. In this short paper we describe a complementary approach for practical coursework that encourages students to take a holistic approach to thinking about Information Retrieval systems. This approach helps students to consider Information Retrieval systems as a whole system, the potential users of a specific Information Retrieval system, and appropriate evaluation approaches.

Keywords: constructivism, information retrieval, problem solving

1. INTRODUCTION

Many Information Retrieval courses take a reductionist approach: reducing the complexity of Information Retrieval systems into a series of decomposed modules for indexing, retrieval, results presentation. Partly this is of necessity; teaching is necessarily linear and decomposed into manageable units (lectures, tutorials, laboratories). One effect of this mode of teaching is that the holism of an Information Retrieval system can be missed. The holism reflects the fact that an Information Retrieval system is a set of complex, interacting modules, the combination of which may have effects (in terms of retrieval results, for example) that cannot be predicted from the simple study of the individual units. Although the field of Information Retrieval usually aims for solutions that are good most of the time for most users [1], Information Retrieval systems are not single units but must be designed with respect to a specific context. That is, the media being searched, the potential users and potential user tasks can imply different retrieval designs.

In this short paper we describe a complementary approach for practical coursework that encourages students to take a holistic approach to thinking about Information Retrieval systems. This approach helps students to consider Information Retrieval systems as a whole system, the potential users of a specific Information Retrieval system, and appropriate evaluation approaches. The challenge is to encourage students to think first about for what problem (what information objects are being searched, by whom and for what purposes) Information Retrieval may be a solution.

In section 2, we describe the context in which we developed this approach to coursework and the previous pedagogic challenges we were trying to overcome. In section 3 we describe the nature of the coursework exercise, with some examples, and in section 4 we summarise with a brief discussion and recommendations.

2. CONTEXT

The CIS Department at The University of Strathclyde teaches Information Retrieval at both undergraduate and postgraduate level. The postgraduate Information Retrieval class is taught to the Masters in Library and Information Studies course; the undergraduate course is taught as part of the Computer Science curriculum. This paper reports on the experiences of teaching Information Retrieval at the undergraduate level. We concentrate on this course due to the special challenges presented which motivated our new thinking about how to instruct and assess Information Retrieval material.

Information Retrieval (IR) is a fourth year course (fourth year is the final year for undergraduates in Scotland) and so is a research-led course. This means that the course should build on students' existing knowledge of Computer Science, gained from their previous three years of study, but should also present material that is informed by the latest research ideas. This implies that the students appreciate the open problems in IR, learn about new approaches and more radical solutions that are still in the laboratory stages of development. In this course we are particularly keen that students learn about the breadth of IR problems and domains as much of their current experience is with Web search engines.

IR has two assessments: a final exam worth 80% of the course marks and a separate coursework assessment to be completed before the end of the course, worth 20% of the course marks. University regulations enforce a final exam for final year student courses and the 80/20 split of exam/coursework marks.

The coursework component of IR faced several challenges:

- A relatively large cohort of students. IR is a large, popular final year option: the class numbers have ranged between 60 and 90 students in previous years. The large size of the class meant that the main instruction method is lecturing supplemented with relatively large sized tutorials and laboratory sessions which gave practical experience of IR packages such as Lucene or Terrier.
- Mixed domain expertise. IR is a course shared by students on nine different degree programs ranging from mathematically based degrees (such as our shared Mathematics with Computing degree), through degrees with a heavy engineering influence (such as Computer and Electronic Systems) to degrees that are primarily non-computing (such as Technology and Business Systems, a business degree in which students take a minor in a technology-based discipline such as Computing). The material of the course must, therefore be both be high-level, to justify its research-led nature and also accessible and interesting to a wide range of academic interests.
- Different skill bases. As students come from different academic backgrounds, assessments must also be appropriate to students with different skills from their previous years of study.

Previous attempts at coursework assessment had aimed at constructive alignment of the taught material [2] essentially aligning student practical activities to fit learning outcomes. Understanding the process of stemming, for example, would be aligned with using a stemming algorithm to process text or designing a stemming algorithm. This was unsatisfactory for several reasons, in particular:

- Students generally did not appreciate the importance of concepts such as relevance within IR. Although they recognised definitions of relevance they did not recognise the practical impact of varying definitions of relevance, for example in the use of relevance feedback, or the complex and situational nature of relevance.
- Students did not fully appreciate design decisions involved in IR processes such as indexing or retrieval. Although students gained practical experience in how these processes operated and understood the theoretical background they didn't realise the effect of the choice of using these algorithms within an overall system. The distribution of learning activities to individual lectures, laboratories and tutorials also encouraged students to see different IR processes as discrete units rather than part of a whole.
- Students did not always appreciate that different types of information objects require different retrieval solutions.

3. COURSEWORK

Our new approach to assessment was to build on constructivist principles of learning where knowledge is constructed based on mental activity. In such a view learners are seeking understanding based on active engagement with material [3]. We are particularly concerned with social constructivism in which groups of students construct their own understanding of Information Retrieval. Social constructivism does not necessarily give rise to correct understandings (in terms of the teacher's aims) but the process of social constructivism gives a way to understand how students are using material delivered in lectures to build their own mental model of Information Retrieval.

The new assessment model was to turn the assessment into a group exercise in which each group was given a set of information objects, asked to consider which user group may use the information objects, what type of search tasks the users would run on the objects and asked to design, build and evaluate an interactive Information Retrieval system for that group of users. Each group was given the Java-based Lucene toolkit to construct their system. The use of Lucene meant that the group did not have to invest time in implementing low-level retrieval and indexing code but could concentrate on appropriate design decisions for their documents, e.g. whether to use stemming, to use index fields or whole texts, etc.

Students were allocated to groups to balance different abilities between groups; each group had at least one student majoring in Computer Science, and who could be expected to participate in the technical construction of the system. Each group had 5 members. The marking scheme allowed each group to prioritise some activities (e.g. evaluation or GUI design) so each team could operate to strengths of the group.

The marking scheme made it clear to students that this was not a programming exercise; although these were primarily Computer Science students this was not a programming class. Instead, the marking scheme emphasised understanding of the course concepts, appropriate interface design and advanced search functionality, and a mixture of evaluation approaches (system-based and user-based). Novelty and creativity was rewarded. Each developed system was demonstrated the week before submission and submission was a final report documenting the system and design rationale. Anonymous peer assessment forms were used where individual group members were asked to rate their own performance within the group, as well as the performance of the other members. This data was used to alter the group mark to obtain individual marks for the coursework.

The information objects included Shakespeare's plays, religious texts, newspapers, and electronic books. The newspapers included samples of TREC collections but we deliberately chose ones that were recognisably different from each other, e.g. Federal Register which has a number of long, linked documents versus the Wall Street Journal which has short self-contained articles. We also included dynamic collections; one group were given a program to download cached URLs from a Web browser cache (to create a collection of Web URLs), a second group were given a program to read a Thunderbird email inbox (to create a collection of emails). The latter two collections would, therefore, vary depending on which user's email/browser was being analysed.

Laboratory sessions were no longer used as guided instructional sessions and were instead used as spaces where students could work together as groups. Laboratories were also used as clinics (2 hrs per week), to cope with practical questions on Lucene, to answer groups' queries about the exercise in general and to discuss the

solutions being developed by the groups. Students were expected to work outside of the labs as well and each student was expected to spend about 30 hours on the exercise.

Figures 1-3 show screen shots for three of last year's solutions. Figure 1 shows the interface for email retrieval with range restrictions (time, date, presence of attachments), Figure 2 shows an advanced search interface for electronic book searching (with field searching, Boolean or free-text searching, drag-and-drop relevance feedback and user ratings), Figure 3 shows two interfaces for searching television news transcripts (left hand side contains search-by-speaker, user history and query expansion, right hand side is an alternative visualisation interface in which best-match stories are indicated by size).

Choose your search options:

From: Subject: Date: Sent: Size: Between:

To: Attachments: Range?: And:

Body contains: Any of the terms:

Date	From	Subject	Body
Mon, 18/04/07 14:41:41	linda.hunter@strath.ac.uk	Chris Taylor	For: ideas and on: uncollected research items
Mon, 18/04/07 15:19:42	linda.hunter@strath.ac.uk	Chris Taylor	270.000000
Wed, 18/04/07 12:18:46	linda.hunter@strath.ac.uk	Chris Taylor	For: 00
Wed, 18/04/07 12:38:55	linda.hunter@strath.ac.uk	Sofia Tassi	CS304 Operating Systems: Shell Interface Demos/Animations
Wed, 18/04/07 13:17:40	linda.hunter@strath.ac.uk	Chris Taylor	CS
Wed, 18/04/07 13:43:03	linda.hunter@strath.ac.uk	CS Coursework Submission System	CS304.Eg2
Thu, 19/04/07 14:54:32	linda.hunter@strath.ac.uk	Christopher Hildes	For: Great Value Holiday Essentials at 6C World
Fri, 20/04/07 12:45:55	linda.hunter@strath.ac.uk	Chris Taylor	2000.000000
Tue, 24/04/07 19:04:12	all.students@strath.ac.uk	Mark Roper	April lab availability Wednesday 25th April
Wed, 25/04/07 10:02:30	all.students@strath.ac.uk	Linda Hunter	Office Closure

OFFICE CLOSURE

To: all.students@strath.ac.uk From: Linda Hunter, linda.hunter@strath.ac.uk Sent: Wed, 25/04/07 10:02:30

Hi

Please note the Office will now close from 3.30pm to 4.30pm and not 3pm - 4pm as previously advised.

Linda

**

Linda M Hunter
Departmental Secretary
University of Strathclyde
Graduate School of Informatics
Department of Computer & Information Sciences

Figure 1: Email Interface

the best interface ever

Search Query

search

Advanced Options

Book Contents: Any terms: ☐ All terms: ☐

Heading Contents: Any terms: ☐ All terms: ☐

Title Contents: Any terms: ☐ All terms: ☐

In Book: CGI Programming on the World Wide Web

Fuzzy Search: ☒

Results

The query "search" returned 69 documents. Showing 1 to 25 of 69 results

Page 1 of 2

[Chapter 9] 9.4 Search/Index Gateway

Filename: web/cgi/ch09_04.htm Relevance Score 44.5% Similar Matches

71 Rating: 4.0/5 (1 vote cast)

Figure 2: e-book interface



Figure 3: news transcript interfaces

4. OUTCOMES

The coursework was intended to help students think about Information Retrieval systems as a whole, as distinct types of systems (not simply Web search engines) and to think not just about the system but also about the potential users and information tasks.

Performing this exercise was not trivial for the students involved. Each group were forced to reconsider and assemble their initial understandings of the course material to engineer a group response to the coursework exercise. A particular challenge, expressed by the students, was in understanding what interactive support might be useful – there are many possible interactive support tools but which were useful for individual situations?

All groups were successful in understanding that different documents may imply both different searchers and different search tasks; users of Shakespeare plays could be drama students, actors or teachers, for example, and may want to retrieve specific quotes, character actions or action within the play. These understandings of search tasks fed into the search functionality: students constructing systems for long documents (such as FDR documents or books) usually employed field search, students working on email recognised that time of emails or senders (document authors) were important search keys.

All groups showed a greater understanding of the difficulties of evaluating IR systems. All groups chose to construct interactive IR systems and attempted user-evaluations, mostly usability evaluations, and test collection evaluations. The test collection component raised a particular challenge for the students, namely the question of relevance; what does it mean to evaluate relevance with respect to a search tasks (where individuals may have different ideas of relevance) and how to estimate recall without total knowledge of the relevant documents (raising issues of sampling – some groups even tried pooling).

This exercise was not initially popular when the exercise was announced. Students immediately understood the open challenges and lack of a single solution. However, after the exercise, the majority reported a deeper understanding of the challenges of Information Retrieval and more confidence in applying this knowledge. The students' answers to exam questions also showed a greater depth of understanding of IR issues compared to previous years.

REFERENCES

- [1] James Allan: HARD Track Overview in TREC 2003: High Accuracy Retrieval from Documents. TREC 2003: 24-37. 2003.
- [2] John Biggs. Teaching for Quality Learning at University, (SRHE and Open University Press, Buckingham. 1999.
- [3] Ernst von Glaserfeld. Cognition, Construction of Knowledge and Teaching. Synthese 80(1), 121–, 1989.