Information Extraction from Heterogeneous WWW Resources

Muhammad Suhaizan Sulong, and Farid Meziane

School of Computing, Science & Engineering, University of Salford, Manchester, United Kingdom.

Abstract. The information available on the WWW is growing very fast. However, a fundamental problem with the information on the WWW is its lack of structure making its exploitation very difficult. As a result, the desired information is getting more difficult to retrieve and extract. To overcome this problem many tools and techniques are being developed and used for locating the web pages of interest and extracting the desired information from these pages. In this paper we present the first prototype of an Information Extraction (IE) system that attempts to extract information on different Computer Science related courses offered by British Universities.

Keywords. Information extraction, Semantic web, Unstructured information.

1. Introduction and Motivation

It is estimated that there are more than 200 millions of web pages available on the WWW (Craven *et al.*, 1998; Freitag, 1998) and this number keeps increasing on a daily basis. It is also stated that computers cannot understand any of these pages (Craven *et al.*, 1998) and humans can hardly make use of this wealth of information without the use of tools that guide them towards the desired information. Indeed, most queries nowadays will return thousands of hits making their manual exploitation time consuming. Initially, search engines were used. However, lately research has shifted to IE system. These systems attempt to localise information within a document rather then just finding the document. There is a lot of interest in extracting information from the WWW (Craven *et al.*, 1998; Freitag, 1998; Hammer *et al.*, 1997; Meziane and Kasiran, 2003; Soderland, 1997; Vijjappu *et al.*, 2001). A new vision and structure of the WWW is being looked at and this is known as the Semantic Web. The Semantic Web is an extension of the current web in which information is given a well-defined meaning, better enabling computers and people to work in cooperation (Berners-Lee et al., 2001). It is a mesh of information linked up (or as a globally linked database) being an efficient way of representing data on the WWW to be easily processed by machines.

In this paper we present an IE system that helps prospective students finding information about computer science courses in UK universities. With more then hundred universities and thousands of courses to choose from, finding the right information about a desired course is time consuming. For the purpose of this project, the entry point to our extraction system is the UK's university's website URL (Unified Resource Location), the Higher Education and Research Opportunities (HERO) website (http://www.hero.ac.uk). The information extracted is presented in a tabular way and comprising the following information: the University name, the course name, the course duration, the course entry requirements and the course fees. The extracted courses are Computer Science and related areas such as Computing, Information Systems, Information Technology, Software Engineering, etc.

The remaining of this paper is organised as follows: section 2 describes the overall approach and the system's architecture. Section 3 describes the rules used for extracting the various information items. Section 4 presents the navigation process used for searching the required information. The evaluation and implementation of the system are summarised in section 5 and in section 6 we present future development and the main conclusions.

2. Overall Approach and System Architecture

The overall system architecture is shown in Figure 1. The dotted square represents the core of the system that consists of the extraction rules module, extraction mechanisms that contains extraction engine and the navigation module. Outside the dotted lines are the modules that process user queries, the output of the system which is represented as a set of HTML pages and a database that stores the extracted results. Given the nature of the system, where updates are not very frequent as we expect Universities to update information about their courses once a year, results are stored in the database for similar queries. At its current implementation, the entry to the system is the set of web pages contained in the Higher Education and Research Opportunities (HERO) website (http://www.hero.ac.uk). The system aims to extract one or more information listed in section 1 from a University's website. Once the top level of the website is loaded, extraction rules will be applied for each module. If the system fails to extract the required information then the links of the page are collected and navigation rules are applied for a better selection of the links to be used first. The extracted information is then stored in the database. We adopted a learning approach for the system. We have first used 30 websites to manually define rules for extracting the required information.

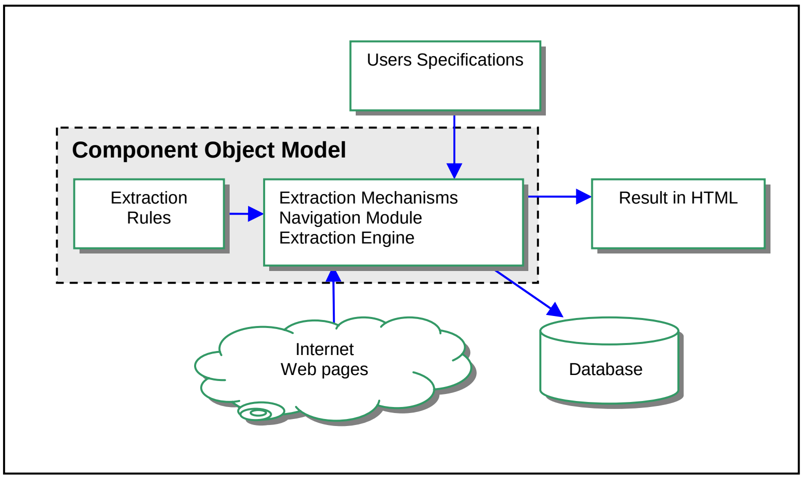


Figure 1. System's architecture

3. Extraction Rules

When trying to extract an information item from a website, we take into account the information that precede the item, the structure of the item and the information that may follow the item. Each extraction rule will be composed by a "precede_expression", a valid "structure" of the item and a "follow_expression". A "precede_ expression" is an expression we expect to find before the item and a "follow_expression" is an expression the system expects to find after the item. An extraction rule will have the format:

extraction_rule = precede_expression;
item_structure;
follow_expression

In the following subsections we define a set of "precede_expression", a set of "item_structures" and a set of "follow_expressions" for the course duration and entry requirements items.

3.1. The course duration rules

From the studied sample, the common expressions found to precede course duration are:

DPList = [full time, full-time Full Time, Full time, full, ft,
part time, part-time, Part Time, Part time, pt]

The course duration may have different formats on a University's web site. It varies from the use of characters only such as twelve months or a mixture of characters and digits such as 12 months. Based on the sample, the rules used for defining the structures of a course duration are:

course_duration = numeric_value, "year"
course_duration = char_value ,"year"
course_duration = numeric_value, "years"
course_duration = char_value, "years"
course_duration = numeric_value, "months"
course_duration = char_value ,"months"
course_duration = numeric_value,"mth"

We do not associate any length to the numeric value. However, when the information is extracted, the course duration is stored as a string of digits only representing the number of months. In some websites, the course duration was followed by the expression given in the list DFList.

DFList = [Duration, duration, on a full time basis, full time,
on a full time basis, full time , full-time Full
Time, Full time, full, ft, part time, part-time,
Part Time, Part time, pt , period]

3.2. The entry requirements rules

From the studied sample, the common expressions found to precede entry requirements are:

EPList = [first class, second class, upper second class,
lower second class, good]

The entry "requirements" is given as a textual description of the entry requirements and was difficult to extract. The approach we have adopted is to extract a sentence that contains one of the words or expressions in the EPList and a word or expression from EFList in this order.

Words that normally follow an entry requirements condition ends with one of the following words.

EFList = [non-computing, any discipline, computer related
Science, mathematica, engineering]

3.3. Rules formulation

The information we want to extract will not be represented by web pages or links between the web pages. They represent small parts of text embedded in pages. We formulate the rules used for the information extraction process using Horn clauses similar to [Craven *et al*., 1998]. We illustrate only the rules for the course duration extraction as the rules for extracting other items are similar.

```
course_duration(String):- before(String, String1),
                          member(String1, DPList),
                          cdstructure(String1),
                          after(String, String2),
                         member(String2, DFList).

course_duration (String):- before(String, String1),
                          member(String1, DPList),
                          cdstructure(String1).

course_duration (String):- cdtructure(String),
                          after(String, String2),
                          member(String2, DFList).

course_duration (String):- cdtructure(String).
```

where

before(String1, String2): means *String2* appears just before *String1* in the text.
after(String1,String2): means *String2* appears just after *String1* in the text.
cdtucture(String): means that *String* has the structure of a *course_duration*.
member(X,List): is the usual membership relation where X is member of the list List.

4. The Navigation Process

Websites contain collections of hypertext documents. A hypertext document is typically composed of nodes and links. The nodes are the documents part and the links are the relationships between documents. A node contains the information and a link allows the navigation of other documents of the hypertext collection. A link (n_1; n_2) therefore represents a connection between the source node n_1 and the destination node n_2 (Frei and Schäuble, 1992). Thus, a hypertext document is better represented as a directed graph. Furthermore, we distinguish two types of links (Frei and Stieger 1991), referential links and semantic links. Referential links are used for a better organisation and easy reading of a document. However, the purpose of semantic links is to provide more details, additional or similar information about a specific topic.

Any content-specific retrieval system should consider both nodes and semantic links. In our current approach, we give a more restricted definition of semantic links. A semantic link is a link that can be index by one or more words from a predefined set of keys. This restricted view of semantic links is used to target primarily those links that have a high probability of containing the information the system is looking after. Hence, improving the overall search time of the extraction process as a single node may contain hundreds of links.

In addition to the source node and destination node already associated with a link, we now associate a list of indices for each link. Furthermore, semantic links are divided into three subtypes depending on how they appear on the node and the nature of the destination node. The link name can appear as simple text or an image. The target node can be a static HTML node or a dynamic one (the result of querying a database for example). Our system does not deal with extracting information from images and needs therefore to understand the URL (Universal Resource Locator) of the target node. The navigation process is summarized in Figure 2.

The indexation process starts with the tokenization of the link name and target URL. Each token is then compared to a predefined list of indices. If the link name is textual we index both link name and target URL however, if the link is an image we just index the target URL. Again using the list of the first 30 websites selected, we extracted the following indices given in lists NKList. We did not differentiate between the different information items as they are normally found in the same place. A link whose NKList Name is not empty will be considered as a semantic link. Otherwise, it will be considered as a referential links. During navigation process, only semantic nodes are used.

NK = [Prospectus, Postgraduates, Undergraduates,
Postgraduate Study, Undergraduate Study,
Course Information, Taught Courses,
Computing]

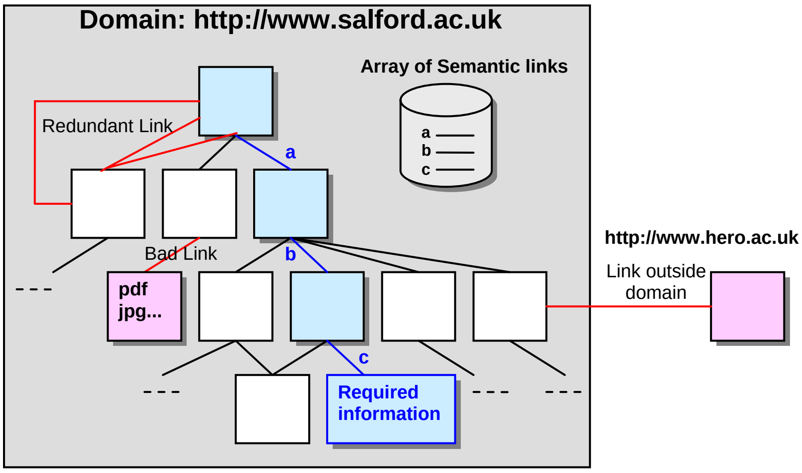


Figure 2. The navigation process

The extraction process can be summarised by the algorithm given in Figure 3 and used in [Meziane and Kasiran, 2003].

```
extract(page, existence-item)
{
      apply extraction rules to page
I     f existence item found
      return item
else
{
      extract all links in the current page
      index all links in current page
      let S be the set semantic links
      for each link ks in S
      extract(ks, existence-item)
}
return "not found"
}
```

Figure 3. The navigation process algorithm

5. System Implementation and evaluation

The system is implemented using a 3-tiers architecture as shown in Figure 4. The first tier representing the system's interface, the second the search and navigation system and the third the system's database. The system is implemented using Object-Oriented Technology with Visual Basic 6.0. The class diagram is given in Figure 5. An example of the results obtained is given in Figure 6.

The measures regularly used to evaluate Information Extraction (IE) Systems are: precision and recall (Robertson and Willett, 1969; Lehnert and Sundheim, 1991). Precision measures the ratio of the correctly extracted information against all the extracted information. Recall measures the ratio of the correct information extracted from the texts against all the available information.

The effectiveness of the extraction procedure was tested with the entire collection consisting of another 30 different websites of various universities that is selected randomly from the HERO website containing courses for postgraduate, mainly covering various fields of computer science. Also a significant portion of this collection is made up of computing, software engineering, distributed networking, Internet computing and computer-related courses.

Despite the diversity of the collection the system works well and the developed rules achieve medium rates of precision and recall. The results are summarised in table 1.

The first row shows the number of courses for which the correct duration and entry requirements are extracted. The second row those incorrectly extracted. The reasons for not extracting the information are quite interesting to observe and the rules are updated as a consequence. In one case 2 courses were listed in a single web page which means that each link of the course is pointed to the same web page i.e. same URL. The information is presented using bookmark (i.e. the use of hash (#) in the URL) to point to the respective course. These affects both attributes of extraction; the course duration and course requirements. Once the details of one course were extracted, the system did not return to that page. In another case it was a rule that was not developed as that case was not in the initial 30 pages used to manually develop the rules. In the entry requirements the keywords "honours graduates" and "honours BSc degree" were used instead of the ones the system was expecting such as "honours degree", "first degree", "initial degree", and "suitable for graduates".

Out of the 30 universities, the system was unable to extract the course information and only one of the 5 universities extracted partially. Only course requirements were able to be extracted but not the course duration. The keyword used in that web page is "1/3 year" which means 1 year full time course and 3 year part time course. Therefore, it does not match with the set of keywords defined in the extraction rules for course duration.

Frames were used by 2 universities as a layout of their website interface. This means that the system at current implementation is unable to extract and retrieve hyperlinks defined with frames and need more rules and learning process to be included. There was only one case where all the names of the courses offered did not match with the defined extraction rules. The courses are MSc. E-Business Systems, Diploma/MSc. in Business Systems Analysis and Design and MSc. in Software Systems as these were not defined as Computer Science related courses.

Most of the universities used the name of the course as hyperlink to direct the respective web page that contains the course information except one university that does not use the name of the course. Therefore, the system was unable to display the correct course name to the users.

Table 1. The precision and recall of the system

	Duration	Entry Requirements
Correctly Extracted	21	19
Incorrectly Extracted	4	7
Not extracted	5	4
Precision	84%	73%
Recall	70%	63%

6. Future Work and Conclusion

We learned a lot of lessons from this first version of the system. The nature of the information on the WWW which is unstructured in nature makes it difficult to define rules that will work for every website as developers use different naming conventions and organise their websites in different ways. The more the system is used the more rules will be added and will become more efficient. Various websites starts using images to convey their information. This will require some research on how information can be extracted from images. The systems should also be generalised to take more free user queries and work on the results of a search engines hits rather then from a single website.

7. References

Craven, M., DiPasquo, D., Freitag, D., McCallum, A.K. Mitchell, T.M., Nigam, K., & Slattery, S. (1998). Learning to extract symbolic knowledge from the World Wide Web. *In Proceedings of AAAI-98, 15th Conference of the American Association for Artificial Intelligence (*pp. 509–516), Menlo Park, USA: AAAI Press.

Frei, H.P., & Schäuble, P. (1991). Designing a hyper-media information system. In *DEXA'91* (pp. 449–454), Wien: Springer-Verlag.

Frei, H.P., & Stieger, D. (1992). Making use of hypertext links when retrieving information. In *Proceedings of the ACM Conference on Hypertext and Hypermedia* (pp. 102–111), Milan, Italy.

Freitag, D. (1998). Information extraction from HTML: Application of a General Machine Learning approach. In *Proceedings of the 15th Conference on Artificial Intelligence AAAI-98* (pp. 517–523), Madison, Wisconsin, USA.

Hammer, J., Garcia-Molina, H., Cho, J., Crespo, A., & Aranha, R. (1997). Extracting semi structured information from the web. In *Proceedings of the Workshop on Management for Semi Structured Data*, (pp. 18–25), Tucson, Arizona, USA.

Lehnert, W., & Sundheim, B. (1991). A performance evaluation of text analysis technologies, *AI Magazine*, pp. 81-94.

Meziane, F., & Kasiran, M.K. (2003). Extracting unstructured information from the WWW to support merchant existence in e-commerce, *Proceedings of the 8th International Conference on Application of Natural Language to Information Systems* (NLDB03), Burg, Germany.

Robertson, A.M., & Willett, P. (1996). An upperbound to the performance of ranked-output searching: optimal weighting of query terms using a genetic algorithm, *Journal of Documentation, 52*: 405-420.

Soderland, S. (1997). Learning to extract text-based information from the WorldWide Web. In *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining* (pp. 251–254), California, USA.

Vijjappu, L., Tan, A.H., & Tan, C.L. (2001). Web structure analysis for information mining. In *Proceeding of the 1st International Workshop on Web Document Analysis* (pp. 15–18), Seattle, Washington, USA.

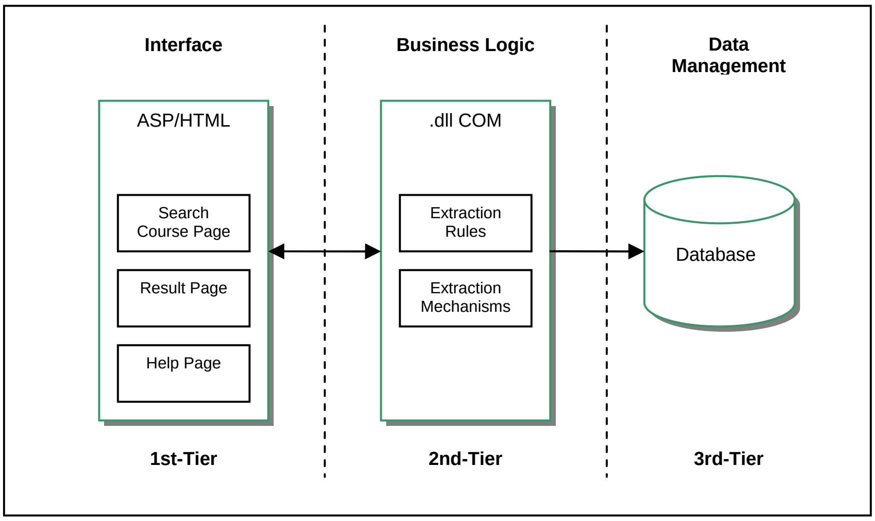


Figure 4. The 3-tiered system architecture

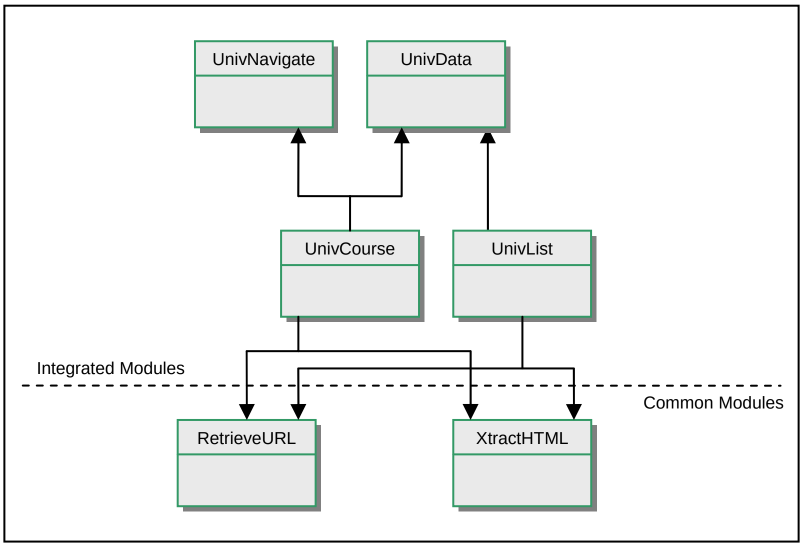
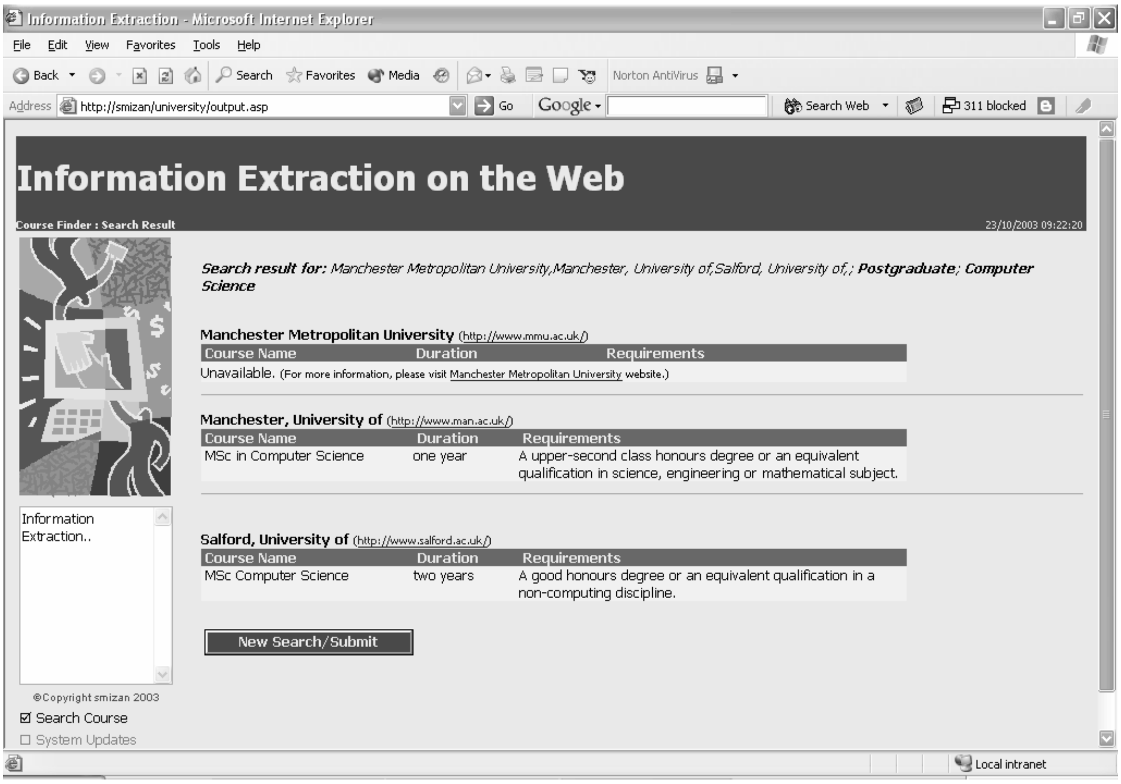


Figure 5. The system’s class diagram


Information Extraction - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media Norton AntiVirus
Address http://smizan/university/output.asp Go Google Search Web 311 blocked
Local intranet

Information Extraction on the Web

Course Finder : Search Result 23/10/2003 09:22:20

Search result for: *Manchester Metropolitan University,Manchester, University of,Salford, University of,;* ***Postgraduate; Computer Science***

Manchester Metropolitan University (http://www.mmu.ac.uk/)

Course Name	Duration	Requirements
Unavailable. (For more information, please visit Manchester Metropolitan University website.)		

Manchester, University of (http://www.man.ac.uk/)

Course Name	Duration	Requirements
MSc in Computer Science	one year	A upper-second class honours degree or an equivalent qualification in science, engineering or mathematical subject.

Salford, University of (http://www.salford.ac.uk/)

Course Name	Duration	Requirements
MSc Computer Science	two years	A good honours degree or an equivalent qualification in a non-computing discipline.

New Search/Submit

Information Extraction..

©Copyright smizan 2003

☑ Search Course

☐ System Updates

Figure 6. A sample of the system’s results