Using Information Density to Navigate the Web[1]

Jeremy Ellman[#][*] and John Tait[#]

Abstract

This paper describes a system being developed to identify Internet WWW pages that most closely respond to a users' requirements. The system is designed to enhance, rather than replace existing search engines. It collects pages identified by the search engine, and then uses an external lexical thesaurus to analyse their contents. This provides a secondary ordering metric for the pages based on a Case Based approach to text analysis.

Introduction

Most serious Internet users will recognise that there are significant problems in identifying appropriate information on the Internet. Often these are connected with the size and diversity of Internet Information. Queries to the well known search engines return hundreds of resulting links. However, it is impractical to follow more than a fraction of these, particularly since the value of Web pages is highly variable, and the interaction times are often long.

This paper describes a system that collects the pages offered by a search engine and analyses their to determine which documents are either vacuous, or more likely to be of interest to the user. It uses two techniques to do this, one active the other passive.

The passive technique determines that candidate documents do not contain serious information. This means they are likely to be less interesting to the user. The active technique matches the content profile of candidate pages against that of a pre-selected exemplar. If they contain a similar content structure, they are likely to be of interest. Both techniques are based on Lexical Chains (Morris and Hirst 1991).

A lexical chain is a sequence of words in a text that between them make up one strand of the text's meaning. Links between words that make up the lexical chains are usually identified by using an external knowledge source. Morris and Hirst (1991) introduced the concept basing it on Roget's thesaurus. Words may be linked into chains when they share the same Roget category, contain index entries that point to the other's Roget category, or share several other, more distant, thesaural relationships.

Once the Lexical Chains have been calculated for a document, we can find it's Information Metric. This is a measure of its semantic connectivity, and is found the sum of the link strength of the lexical chains. A low Information Metric implies a page with low argument structure and vice-versa. This can be used as a passive, secondary ordering measure over that determined by the search engine.

[1] In Proc IEE Intelligent World Wide Web Agents 17th March 1997

[#]School of Computing and Information Systems, University of Sunderland, UK

[*]MARI Computer Systems
Wansbeck Business Park
Ashington NE 63 8QZ, UK
Email : Jeremy.Ellman@mari.co.uk.

Active analysis uses the page Document Profile. This is a set of semantic (Roget) categories that anchor key lexical chains. Weights based on chain length and strength are attached to these categories. This profile can be matched against that derived from the examplar in a Case Based Reasoning approach using a nearest neighbour algorithm (Aamodt and Plaza 1994). The program is known as *"Hesperus"*.

This paper proceeds as follows: Firstly we shall look at some problems inherent in using an Information Retrieval search engine such as AltaVista or Excite. Next we shall describe the Generic Document profile as a part solution to some of these problems, and describe how this may be calculated. We will then discuss some related work and present initial conclusions.

The Information Retrieval Task.

This section summarises the nature of the Information Retrieval (IR) task (Salton 1989 and many others) as it is usually understood. Its purpose it to highlight several assumptions that are intrinsic to IR that are not immediately applicable to Internet resource discovery. Once these concepts are identified, they can be replaced with broader, more robust approaches.

In general the traditional approach to IR might be summarised as follows:-

① There is a large collection of documents.
② A searchable index is created from this collection of documents.
③ A user has a fixed requirement for information.
④ By querying the index the document that contains the required information can be found.

There are several assumptions implicit in the above view of IR. Although they are obvious, it is worth raising them, since I shall argue below that they do not always hold when applying IR to the web.

The Index Assumption

The role of the Index is of course key to Information Retrieval. It is built by processing the document collection, and storing key information about each document. Thus, when the user' s query references some more or less particular information, he can be sure that an appropriate index entry will be found. This is a natural consequence of the implicit assumption that the index has referenced the entire document collection.

Vocabulary Problem

The next implicit assumption concerns the vocabulary problem. This problem arises when the user's query does not reference the exact keyword in the index (or something morphologically related to it), but has used any other term (eg a synonym, a more descriptive phrase, or a general rather than technical term). In this case the query will not locate the most appropriate document. Whilst solutions have been proposed for simple variants of this problem (eg Chen 1994), there is an implicit expectation that the user understands the domain well enough to phrase his query using appropriate terminology.

Relevance Feedback

The third implicit assumption concerns relevance feedback. It has been commonly observed that retrieval performance is enhanced when, after a first retrieval attempt, the user indicates the most relevant documents and his query is repeated. The assumption here is that user requirements are stable. Many authors believe that this is rarely the case in practice, and indeed, with respect to IR, Croft (1995a) has observed that users often use relevance feedback to browse a document collection. This could be the result of the user acquiring new terminology as a result of the retrieval process, or better understanding the type of information he is trying to locate.

The Generic Document Profile

The “*Generic Document Profile”*. is simply a set of semantic categories derived from Roget' s thesaurus with associated weights. These weights based on chain length and strength are attached to these categories. This profile can be matched against that derived from an example in a Case Based Reasoning approach using a nearest neighbour algorithm (Aamodt and Plaza 1994).

In Case Based Reasoning (CBR) a query and examples are usually represented as attribute value pairs. Thus to apply CBR to document comparison, both the text acting as a query, and the documents to be compared against need to be represented in equivalent terms. If this representation is based on simple terms, the problem becomes hugely complex, since there are many words in a language. This would also be a fragile approach, since semantically equivalent words would not count as equal. However, if we represent a document as Roget categories, the problem becomes tractable, since there are only 1024 main thesaurus categories. Document strength in these categories is derived from the lexical chains extracted. I call this representation the “*Generic Document Profile”*, since it is not word specific, and is derived from the whole text. Now all that remains is to describe how a text may be analysed so that values in each particular category can be determined. This is done as a post processing step after defining the lexical chains the document contains.

What is a lexical chain?

A lexical chain is a set of words in a text that are related by both proximity, and by lexical linking relations between the words. Since Morris and Hirst (1991) lexical chains have been applied to several different areas of language processing such as word sense disambiguation and text segmentation, (Okumura and Honda 1994), malapropism detection (StOnge and Hirst 1995), detection of HyperText links in newspaper articles (Green 1996), and lexical cohesion and Information Retrieval (Stairmand and Black 1996, Stairmand 1996b)

Lexical chains are composed of links. That is, a relationship found between two words using an external thesaurus such as Roget' s, or WordNet (Miller 1991). Morris and Hirst (1991) suggested several possible ways that words can be linked using Roget’s Thesaurus. These include the identical word, or what we shall call the ID relation. We extend this to include lemmas (or inflected forms) of that word.

Words that are members of the same thesaural category may also be linked. We call this the CAT relation.

The entries in Roget' s thesaurus are ordered into groups of categoriesiȩ there is a semantic hierarchy). Thus words within neighbouring thesaural categories may be linked. This is the GRP relationship.

Words in Roget categories often refer to other categories. Words may therefore be related where one word' s entry refers to an entry that contains the other word. We call this the ONE relation, since it involves one level of indirection.

Finally, Morris and Hirst (1991) also propose a relation where both words have thesaural entries that contain pointers to a third category. In practice however this resulted in so many spurious connections it could not be used. This was due to lexical ambiguity.

The creation of the Generic Document profile can be time-consuming so it is split into several stages. These are:-

1. Index the Roget, and create "pointer table" (once only)
2. Lemmatise and Tag input text
3. Eliminate Stopwords
4. Lookup word in Roget
5. Create Chains from the text
6. Create Generic Document Profile

Creating Chains from Text

The algorithm used to create chains from the pre-processed text was as follows:-

```
1.  get a new word from the input
2.  For all chains in the queue
3.          for all the links in the chain
4.                  try to link the input word to a word in that chain using ID
5.                  if link is made add word to the front of the chain
6.                  if a link is made using one (of several) word senses
7.                          then DELETE other word senses
8.                          calculate new chain weight, and re-sort the queue
9.                          goto STEP 1
10.                 if input word greater than the MAXDIST (ie 500 words) break
11. Repeat step 2-10 using CAT check
12. Repeat step 2-10 using ONE check
13. Repeat step 2-10 using GRP check
14. if word can not be linked form a new chain
```

The Generic Document Profile is created from the lexical chains. The strength of every link in every chain is taken, and summed into the appropriate profile category. This gives the required attribute value representation.

Searching the Web using Hesperus.

At this stage of the work document analysis and matching are complete, but assisted Web searching is primitive. A simple Java program takes a first set of key words and searches AltaVista. Each of the links returned are then retrieved and saved to files for which the generic profile is then calculated. The user then starts to view the files. When the user selects one file as being "most interesting", the order in which the others are presented is automatically changed to present the most similar next. Future work will allow users to indicate a "best document". Additional web searches could be based on key words defined from this in a variation of relevance feedback.

Related Work

Web Information Retrieval Agents

Web based Information Retrieval is a highly active research area, and there are several approaches. Some are knowledge rich, (as in the Softbot, WebWatcher and Letizia) where the agent is intelligent, and some knowledge poor approaches. Others capitalise on the huge resources existing search engines to improve their utility. MetaCrawler for example passes queries on to full search engines, and collates the results. Smeaton and Crimmins (1996) also describe a similar system based on data fusion that is similar in operation

SoftBot

The Softbot (Software robot, Oren and Etzioni 1994) is an AI based intelligent agent to the Internet. It directs a variety of Internet tools (FTP, TELNET, ARCHIE, GOPHER etc) to achieve a user' s goals. The Softbot uses AI planning techniques to select the most appropriate tools, and constructs an plan, subgoals to achieve it. These would include referencing remote servers and different tool applications where appropriate. The long term stated aim of the Softbot is to enable naive users to effectively locate, monitor and transmit information across the net.

WebWatcher

WebWatcher (Armstrong et al 1995) is an experimental information seeking assistant for Web. It is a modification of the public domain Mosaic web viewer that advises users which hyperlinks to follow next. Webwatcher incorporates machine learning methods to acquire knowledge for automatic hyperlink selection.

These include TFIDF with cosine similarity measure (Salton and McGill 1993), Wordstat, a measure based on the statistics of individual words, and Winnow (Littlestone 1988) that learns a Boolean concept represented as a single linear threshold function of the instance features. A random control was also used. Whilst Winnow and Wordstat were the most effective measures, it is difficult to infer much from this since the system trials were restricted to retrieval of pages in the machine learning area.

Letizia

Letizia (Liebermann 1995) is a user interface agent that assists a user browsing the World Wide Web. As the user operates a conventional Web browser such as Netscape, the agent tracks user behaviour and attempts to anticipate items of interest by doing concurrent, autonomous exploration of links from the user' s current position. The agent automates a browsing strategy consisting of a best-first search augmented by heuristics inferring user interest from browsing behaviour.

Combined Search Interfaces

Metacrawler

Metacrawler (Selberg and Etzioni 1995) permits parallel searches of the Web by executing the same query using several of the search interfaces mentioned above: LYCOS, YAHOO, InfoSeek, Inktomi, and OpenText. Search results are cross correlated and returned as one page to the user. This is inefficient in terms of Web resources, but gives wider coverage than using any single search service.

Smeaton and Crimmins (1996)

Smeaton and Crimmins (1996) describe a system that is similar in style to Metacrawler. That system, which is based on Data Fusion, is distinguished by a client server architecture that is implemented in Java. The system launches a Java applet on the client machine, and then communicates requests to a fusion server[2] that queries several search engines (AltaVista, Excite, OpenText etc) in parallel. The system is aware of the different ranking schemes these search engines use, and collates and eliminates duplicates from the results. Smeaton and Crimmins (1996) report a low level of duplication between the search engines, implying that a meta search agent like this needs to retrieve more than the top ten query results.

Conclusion and Future Work

We have described a program that seeks to improve Internet page location by using a robust Natural Language Analysis method to leverage the results of current search engines. The program assists the user by collecting possible pages in parallel, and then creates Generic Document profiles for them. Once a user has indicated a preference for one page found, other papers that contain similar concepts (not just words) may be indicated. Initial results are promising, and current work is looking to enhance this with a java based browser implementation.

References:

Aamodt A and Plaza E 1994 "Case Based Reasoning: Foundational Issues, Methodological Variations and System Approaches" AI Communication Vol 7, 1 March 1994

Armstrong Robert, Freitag Dayne, Joachims Thorsten, Mitchell Tom "WebWatcher: A Learning Apprentice for the World Wide Web" AAAI Spring Symposium on Information Gathering In Heterogeneous, Distributed Environments. March 1995

Chen Hsinchun "Collaborative Systems: Solving the Vocabulary Problem" IEEE Computer May 1994

DeBra P.M.E. and Post R.D.J (1994) "Information Retrieval in the World Wide Web: Making Client Based searching feasible" in proc WWW 1994

[2] This is required since Java applets may not initiate net connections to other than their point of origin

Dorr B and Jones D 1996 "The role of Word Sense Disambiguation in Lexical Acquisition: Predicting Semantics from Syntactic Cues". Proc COLING 1996

Felbaum 1995 “Wordnet” The MIT Press Cambridge Mass

Green S 1996 "Using Lexical Chains to build Hypertext links in Newspaper Articles" in proc AAAI Symposium

Hirst and St Onge 1995 ‘Lexical Chains as representations of context for the detection and correction of malapropisms” to appear in Felbaum 1995

Miller G., Beckwith R., Felbaum C., Gross D., and Miller K. 1990 ‘Introduction to WordNet: An on-line lexical database” J. Lexicography 3(4) pp235-244

Morris J. and Hirst G. 1991 "Lexical Cohesion computed by thesaural relations as an indicator of the structure of text" Computational Linguistics Vol 17 (1) pp 21-48

Okumura M. and Honda T "Word Sense Disambiguation and text segmentation based on lexical cohesion" Proc COLING 1994 vol 2 pp 755-761

Salton G. and McGill MJ "Introduction to Modern Information Retrieval" McGraw-Hill 1983

Smeaton and Crimmins (1996) "Using a Data Fusion Agent for Searching the WWW" in Proceedings World Wide Web Conference 1996 (available on the internet)

St-Onge David 1995 "Detecting and Correcting Malapropisms with Lexical Chains" MSc Thesis University of Toronto. (available via WWW)

Stairmand M 1996b "A Computational Analysis of Lexical Cohesion with Applications in Information Retrieval" Unpublished PhD Thesis. UMIST Computational Linguistics Laboratory

Stairmand M and Black W J 1996 ‘Conceptual and Contextual Indexing using WordNet-derived Lexical Chains” in proc BCS IRSG

Watson I. and Marir F.1994 "Case Based Reasoning: A review". The Knowledge Engineering Review Vol 9, 4 1994 pp 327-354